State-dependent biasing method for importance sampling in the weighted stochastic simulation algorithm

Min K. Roh,[1,a)] Dan T. Gillespie,[2,b)] and Linda R. Petzold[1,c)]
[1]*Department of Computer Science, University of California Santa Barbara, Santa Barbara, California 93106, USA*
[2]*Dan T. Gillespie Consulting, 30504 Cordoba Pl., Castaic, California 91384, USA*

(Received 7 July 2010; accepted 6 September 2010; published online 1 November 2010)

The weighted stochastic simulation algorithm (wSSA) was developed by Kuwahara and Mura [J. Chem. Phys. **129**, 165101 (2008)] to efficiently estimate the probabilities of rare events in discrete stochastic systems. The wSSA uses importance sampling to enhance the statistical accuracy in the estimation of the probability of the rare event. The original algorithm biases the reaction selection step with a fixed importance sampling parameter. In this paper, we introduce a novel method where the biasing parameter is state-dependent. The new method features improved accuracy, efficiency, and robustness. © *2010 American Institute of Physics*. [doi:10.1063/1.3493460]

I. INTRODUCTION

The stochastic simulation algorithm (SSA) is widely used for the discrete stochastic simulation of chemically reacting systems. Although ensemble simulation by SSA and its variants has been successful in the computation of probability density functions in many chemically reacting systems, the ensemble size needed to compute the probabilities of rare events can be prohibitive.

The weighted SSA (wSSA) was developed by Kuwahara and Mura[1] to efficiently estimate the probabilities of rare events in stochastic chemical systems. The wSSA was developed to estimate $p(\mathbf{x}_0,\varepsilon;t)$, which is the probability that the system, starting at $\mathbf{x}_0$, will reach any state in the set of states ε before time t. The estimation procedure is a carefully biased version of the SSA, which in theory can be used to estimate any expectation of the system. However, it is important to note that, in contrast to SSA trajectories, wSSA trajectories should not be regarded as valid representations of the actual system behavior.

The key element in the wSSA is importance sampling (IS), which is used to bias the reaction selection procedure. The wSSA introduced in Ref. 1 uses a fixed constant as the IS parameter to multiply the original propensities. In this paper, we introduce a state-dependent IS method that has several advantages over the original fixed parameter IS method. The new method features improved accuracy, efficiency, and robustness.

In Sec. II we describe the current status of the wSSA. In Sec. III we present the new state-dependent biasing method. We apply the new biasing method to several examples in Sec. IV and compare its performance with that of the original wSSA. In Sec. V we summarize the results and discuss areas for future work.

II. CURRENT STATUS OF THE WSSA

In this section, we briefly describe the weighted stochastic simulation algorithm. A more detailed explanation of the algorithm can be found in Refs. 1 and 2.

To begin, consider a well-stirred system of molecules of N species $(S_1,\ldots,S_N)$ which interact through M reaction channels $(R_1,\ldots,R_M)$. We specify the state of the system at current time t by the vector $\mathbf{x}=(x_1,\ldots,x_N)$, where x_i is the number of molecules of species S_i. The propensity function a_j of reaction R_j is defined so that $a_j(\mathbf{x})dt$ is the probability that one R_j reaction will occur in the next infinitesimal time interval $[t,t+dt)$, given that its current state is $X(t)=\mathbf{x}$. The propensity sum $a_0(\mathbf{x})$ is defined as $a_0(\mathbf{x})=\Sigma_{j=1}^{M}a_j(\mathbf{x})$. The SSA is based on the fact that the probability that the next reaction will carry the system to $\mathbf{x}+\boldsymbol{\nu}_j$, where $\boldsymbol{\nu}_j$ is the state change vector for reaction j, between times $t+\tau$ and $t+\tau+\partial\tau$ is

$$\text{Prob}\{\mathbf{x}\rightarrow\mathbf{x}+\nu_j \text{ in } (t+\tau,t+\tau+d\tau)\} = a_0(\mathbf{x})e^{-a_0(\mathbf{x})\tau}d\tau\times\frac{a_j(\mathbf{x})}{a_0(\mathbf{x})}. \quad (1)$$

In the direct method implementation of the SSA, we choose τ, the time to the next reaction, by sampling an exponential random variable with mean $1/a_0(\mathbf{x})$. The next reaction index j is chosen with probability $a_j(\mathbf{x})/a_0(\mathbf{x})$.

In the wSSA the time increment τ is chosen as we would in the SSA, but we bias the selection of reaction index j: for that we use, instead of the true propensities $a_j(\mathbf{x})$, an alternate set of propensities $b_j(\mathbf{x})$. We then correct the resulting bias with appropriate weights $w_j(\mathbf{x})$. In the original wSSA, $b_j(\mathbf{x})=\gamma_j a_j(\mathbf{x})$, where $\gamma_j>0$ is called the *importance sam-*

[a)]Electronic mail: min.roh@gmail.com.
[b)]Electronic mail: gillespiedt@mailaps.org.
[c)]Electronic mail: petzold@cs.ucsb.edu.

pling parameter for R_j. Choosing $\gamma_j > 1$ will make R_j more likely to be selected while choosing $\gamma_j < 1$ will have the opposite effect. When $\gamma_j = 1$ for all j, i.e., we do not bias the reaction selection process, the wSSA simply turns into the SSA. Thus, in the wSSA, the right side of Eq. (1) becomes

$$a_0(\mathbf{x})e^{-a_0(\mathbf{x})\tau}d\tau \times \frac{b_j(\mathbf{x})}{b_0(\mathbf{x})}, \tag{2}$$

where $b_0(\mathbf{x}) = \Sigma_{j=1}^{M} b_j(\mathbf{x})$. This biased probability can be restored by multiplying Eq. (2) by the weighting factor,

$$w_j(\mathbf{x}) = \frac{a_j(\mathbf{x})/a_0(\mathbf{x})}{b_j(\mathbf{x})/b_0(\mathbf{x})}. \tag{3}$$

Together, we have

$$\begin{aligned}\text{Prob}\{\mathbf{x} \to \mathbf{x} + v_j \quad \text{in} \quad (t+\tau, t+\tau+d\tau)\} \\ = a_0(\mathbf{x})e^{-a_0(\mathbf{x})\tau}d\tau \times \frac{b_j(\mathbf{x})}{b_0(\mathbf{x})} \times \frac{a_j(\mathbf{x})/a_0(\mathbf{x})}{b_j(\mathbf{x})/b_0(\mathbf{x})}.\end{aligned} \tag{4}$$

We can extend this statistical weighting of a single-reaction jump to an entire trajectory by using the memoryless Markovian property—each jump depends on its starting state but not on the history of the state; therefore, the probability of a single trajectory is the product of all the individual jumps that make up the trajectory. Since each jump in the wSSA requires a correction factor of $w_j(\mathbf{x})$ in Eq. (3), the entire trajectory needs to be weighted by $w = \Pi_j w_j(\mathbf{x})$.

The aim of the wSSA is to estimate

$$\begin{aligned}p(\mathbf{x}_0, \varepsilon; t) \equiv &\text{ the probability that the system,} \\ &\text{starting at time 0 in state } \mathbf{x}_0, \\ &\text{will first reach any state in the set } \varepsilon \\ &\text{before } t.\end{aligned} \tag{5}$$

It is important to note that $p(\mathbf{x}_0, \varepsilon; t)$ is not the probability that the system will reach some state in ε *at* time t, but will have reached that set at least once *before* time t.

Estimating $p(\mathbf{x}_0, \varepsilon; t)$ with the SSA is straightforward. After running n simulations of SSA, we record m_n, the number of trajectories that reached any state in ε before time t. Since each trajectory in the ensemble is equally statistically significant, we can estimate $p(\mathbf{x}_0, \varepsilon; t)$ as m_n/n, which approaches the true probability as $n \to \infty$. While estimating $p(\mathbf{x}_0, \varepsilon; t)$ with the SSA is a simple procedure, an extremely large n is required to obtain an estimate with low uncertainty when $p(\mathbf{x}_0, \varepsilon; t) \ll 1$.

Knowing the uncertainty of an estimate is crucial because it provides quantitative information about the accuracy of the estimate. The one standard deviation uncertainty is given by $\sigma/\sqrt{n}$, where σ is the square root of the sample variance. For sufficiently large n, the true value is 68% likely to fall within one-standard-deviation of the estimate (within the range of $+/-\sigma/\sqrt{n}$). Increasing the uncertainty interval by a factor of 2 raises the confidence level to 95%; increasing it by a factor of 3 gives us a confidence level of 99.7%. Therefore, it is desirable to obtain an estimate with a small uncertainty (i.e., small sample variance) because it signifies that the estimate is close to the true probability.

For unweighted SSA trajectories, the relative uncertainty is (see Ref. 2)

$$\text{relative uncertainty} \equiv \frac{\text{uncertainty}}{m_n/n} = \pm\sqrt{\frac{1-(m_n/n)}{m_n}}. \tag{6}$$

When $p(\mathbf{x}_0, \varepsilon; t) \ll 1$ (and $m_n/n \ll 1$), Eq. (6) reduces to

$$\text{relative uncertainty} \approx \pm\sqrt{\frac{1}{m_n}}. \tag{7}$$

This shows that to we need to observe 10 000 successful trajectories to achieve 1% relative accuracy. Since the average rate of observing a state in ε using SSA is $1/p(\mathbf{x}_0, \varepsilon; t)$, we need about $1/p(\mathbf{x}_0, \varepsilon; t) \times 10\,000$ SSA simulations, which quickly becomes computationally infeasible as $p(\mathbf{x}_0, \varepsilon; t)$ decreases, especially for large systems.

The wSSA resolves the inefficiency of the SSA by assigning a different weight to every trajectory. In estimating the same probability with a given n, we can observe many more successful trajectories using the wSSA than the SSA. Each successful trajectory is likely to have a very small weight, which results from using an alternate set of propensities $b(\mathbf{x})$, instead of the original propensities $a(\mathbf{x})$. Since each trajectory in the wSSA has a different weight, we redefine $m_n/n \equiv m_n^{(1)}/n \equiv (1/n)\Sigma_{k=1}^{n} w_k$, where $w_k = 0$ if the kth trajectory did not reach ε before time t. We also keep track of the second moment of the trajectory weights, $m_n^{(2)}/n \equiv (1/n)\Sigma_{k=1}^{n} w_k^2$ in order to calculate the sample variance, given by $\sigma^2 = (m_n^{(2)}/n) - (m_n^{(1)}/n)^2$. Note that different algorithms, such as described in Ref. 3, can be used to compute a running variance to avoid cancellation error.

The current algorithm can be seen in Fig. 1.

III. THE STATE-DEPENDENT BIASING METHOD

The key element of the wSSA is importance sampling, which is a general technique often used with a Monte Carlo method to reduce the variance of an estimate of interest. The rationale for using this technique is that not all regions of the sample space have the same importance in simulation. When we have some knowledge about which sampling values are more *important* than others, we can use important sampling to improve efficiency as well as accuracy. The technique involves choosing an alternative distribution from which to sample the random numbers. This alternative distribution is chosen such that the important samples are chosen more frequently than they are in the original distribution. After using the alternative distribution to sample, a correction is applied to ensure that the new estimate is unbiased. The wSSA employs this technique in the reaction selection procedure. The next reaction is chosen using $b(\mathbf{x})$ (12° in Algorithm 1), and the bias is corrected with an appropriate weight, $(a_j(\mathbf{x})/b_j(\mathbf{x})) \times (b_0(\mathbf{x})/a_0(\mathbf{x}))$ (13°). Mathematical details of importance sampling and Monte Carlo averaging can be found in Appendix of Ref. 2.

The current method for selecting b functions is to simply multiply the original propensities by a positive scalar γ_j, i.e., $b_j(\mathbf{x}) = \gamma_j a_j(\mathbf{x})$. The fixed multiplier facilitates the wSSA

Algorithm 1 wSSA procedure using constant IS parameters

1° $m_n^{(1)} \leftarrow 0, m_n^{(2)} \leftarrow 0$
2° **for** k=1 to n **do**
3° $s \leftarrow 0, \mathbf{x} \leftarrow \mathbf{x}_0, w \leftarrow 1$
4° evaluate all $a_i(\mathbf{x})$ and $b_i(\mathbf{x})$; calculate $a_0(\mathbf{x})$ and $b_0(\mathbf{x})$
5° **while** $s \leq t$ **do**
6° **if** $\mathbf{x} \in \epsilon$ **then**
7° $m_n^{(1)} \leftarrow m_n^{(1)} + w,\ m_n^{(2)} \leftarrow m_n^{(2)} + w^2$
8° break out of the while loop
9° **end if**
10° generate two unit-interval uniform random numbers r_1 and r_2
11° $\tau \leftarrow a_0^{-1}(\mathbf{x}) \ln(1/r_1)$
12° $j \leftarrow$ smallest integer satisfying $\sum_{i=0}^{j} b_i(\mathbf{x}) \geq r_2 b_0(\mathbf{x})$.
13° $w \leftarrow w \times (a_j(\mathbf{x})/b_j(\mathbf{x})) \times (b_0(\mathbf{x})/a_0(\mathbf{x}))$
14° $s \leftarrow s + \tau,\ \mathbf{x} \leftarrow \mathbf{x} + \nu_j$
15° update $a_i(\mathbf{x})$ and $b_i(\mathbf{x})$; recalculate $a_0(\mathbf{x})$ and $b_0(\mathbf{x})$
16° **end while**
17° **end for**
18° $\sigma^2 = \left(m_n^{(2)}/n\right) - \left(m_n^{(1)}/n\right)^2$
19° repeat from 1 using different b functions to minimize σ^2
20° estimate $p(\mathbf{x}_0, \epsilon; t) = m_n^{(1)}/n$ with a 68% uncertainty of $\pm\sigma/\sqrt{n}$

FIG. 1. The weighted stochastic simulation algorithm using constant importance sampling parameters.

implementation, but there are several associated drawbacks. The most obvious drawback is increased variance due to over- and underperturbations. The changes in species population during the simulation cause the relative propensity of reaction j, $a_j(\mathbf{x})/a_0(\mathbf{x})$, to fluctuate as well, changing the probability of selecting the jth reaction at each time step. If γ_j is constrained to be a constant for all possible values of $a_j(\mathbf{x})/a_0(\mathbf{x})$, then the optimal value for γ_j would be the one that perturbed a_j "just right" for most values of $a_j(\mathbf{x})$. However, if $a_j(\mathbf{x})/a_0(\mathbf{x})$ varied widely, then γ_j would over- or underperturb $a_j(\mathbf{x})$ near the extreme values of $a_j(\mathbf{x})/a_0(\mathbf{x})$. A related drawback concerns a narrow parameter space of γ_j that produces an accurate estimate for $p(\mathbf{x}_0, \varepsilon; t)$. If $a_j(\mathbf{x})/a_0(\mathbf{x})$ reaches values near 0 as well as 1 during a single simulation, then it is impossible to avoid both over- and underperturbations because the jth importance sampling parameter can have only a single value which can avoid only one of the two cases. Thus, there will be very few values for γ_j that produce an accurate estimate. Unless the initial value for γ_j is set near these few values, the resulting estimate is guaranteed to have high variance. This unreliable estimate is not useful in deciding how to perturb γ_j to obtain an estimate with lower variance (step 19° of the Algorithm 1). Consequently, one may waste a lot of computational power searching for a value of γ_j that produces an accurate estimate.

Since the above problems are due to fluctuations in the propensity, which is caused by fluctuations in population size, an intuitive solution would be to vary γ_j according to the current state $\mathbf{x}$. Although an arbitrary function may be used to achieve such goal, we proceed as follows.

First, we partition reactions into three groups: G_E, G_D, and G_N. We define G_E as the set of reactions that are to be *encouraged*. This set may include reactions that directly or indirectly increase the likelihood of rare event observation. The IS parameters for this group have values greater than 1 to increase the reaction firing frequency. Similarly we define G_D as the set of reactions that are to be *discouraged*. The IS parameters for G_D will be between 0 and 1, to decrease the likelihood of firing a reaction in G_D. All reactions that do *not* influence the rare event observation are grouped into G_N. Since the reactions in G_N do not need to be perturbed, the IS parameters for these reactions are set to 1. We note that in practice, optimal partitioning of the reactions requires knowledge of the system, which may not always be available.

Second, we define the relative propensity of reaction j as $\rho_j(\mathbf{x}) \equiv a_j(\mathbf{x})/a_0(\mathbf{x})$. $\rho_j(\mathbf{x})$ is a fractional propensity between 0 and 1 that roughly indicates the likelihood of choosing R_j as the next reaction from the current state $\mathbf{x}$. Since the value of ρ_j at each time step gives a qualitative indication of the amount of perturbation needed by R_j, the new biasing method will define γ_j to be a function of ρ_j.

When $\rho_j \rightarrow 0$, the probability of choosing R_j as the next reaction decreases. Thus, for $R_j \in G_E$, more encouragement is needed as its relative propensity decreases. However, when $\rho_j \approx 1$, R_j is likely to be selected as the next reaction without any additional encouragement. In this case, taking $b_j(\mathbf{x}) > a_j(\mathbf{x})$ will overperturb the system and thus increase the variance in the estimate. Therefore, for $R_j \in G_E$, there must be a value of ρ_j at which no further encouragement is applied. Similarly for $R_j \in G_D$, no further discouragement is

necessary when $\rho_j \approx 0$. The value of ρ_j for which we stop the perturbation is defined as ρ_j^0, i.e., $\gamma_j = 1$ for $\rho_j \geq \rho_j^0 (R_j \in G_E)$ and for $\rho_j \leq \rho_j^0 (R_j \in G_D)$. Lastly for $R_j \in G_D$, more discouragement is necessary as ρ_j increases. This corresponds to $\gamma_j \to 0$ as $\rho_j \to 1$.

Based on above considerations, we take the jth IS parameter of $R_j \in G_E$ as

$$\gamma_j(\rho_j(\mathbf{x})) = \begin{cases} 1 & \text{if } \rho_j(\mathbf{x}) \geq \rho_j^0, \\ g_j(\rho_j(\mathbf{x})) & \text{if } \rho_j(\mathbf{x}) < \rho_j^0 \end{cases} \tag{8}$$

where $g_j(\rho_j)$ is a parabolic function of ρ_j that has the following properties:

$$\begin{aligned} &g_j(0) = \gamma_j^{\max}, \\ &g_j(\rho_j^0) = 1, \\ &g_j'(\rho_j^0) = 0. \end{aligned} \tag{9}$$

For $R_j \in G_D$, we select γ_j as the following:

$$\gamma_j(\rho_j(\mathbf{x})) = \begin{cases} 1 & \text{if } \rho_j(\mathbf{x}) \leq \rho_j^0 \\ \dfrac{1}{h_j(\rho_j(\mathbf{x}))} & \text{if } \rho_j(\mathbf{x}) > \rho_j^0, \end{cases} \tag{10}$$

where $h_j(\rho_j)$ is a parabolic function of ρ_j that has the following properties:

$$\begin{aligned} &h_j(1) = \gamma_j^{\max}, \\ &h_j(\rho_j^0) = 1, \\ &h_j'(\rho_j^0) = 0. \end{aligned} \tag{11}$$

Graphical representations of the function $\gamma_j(\rho_j(\mathbf{x}))$ are illustrated in Fig. 2, for $R_j \in G_E$ and $R_j \in G_D$.

There are two parameters, $\gamma_j^{\max}$ and ρ_j^0, for each $\gamma_j(\rho_j(\mathbf{x}))$. $\gamma_j^{\max}$ is defined as the parameter associated with the maximum perturbation allowed on the jth reaction. Given the values for $\gamma_j^{\max}$ and ρ_j^0, $g_j(\rho_j)$, and $h_j(\rho_j)$ as defined in Eqs. (9) and (11) are unique parabolas whose formula is

$$\begin{aligned} g_j(\rho_j) &= \left(\frac{\gamma_j^{\max} - 1}{(\rho_j^0)^2} \right) (\rho_j^0 - \rho_j)^2 + 1 \quad (\text{if } R_j \in G_E), \\ h_j(\rho_j) &= \left(\frac{\gamma_j^{\max} - 1}{(\rho_j^0 - 1)^2} \right) (\rho_j^0 - \rho_j)^2 + 1 \quad (\text{if } R_j \in G_D). \end{aligned} \tag{12}$$

An important point to keep in mind is that perturbing $a_j(\mathbf{x})$ by an unnecessarily large amount not only potentially increases the number of rare event observations but also increases the variance of an estimate as well. We want to observe enough rare events to calculate necessary statistics, yet at the same time to minimize the variance. The advantage of using $\gamma_j(\rho_j)$ is that a user has the freedom to choose $\gamma_j^{\max}$ and ρ_j^0, whose optimal values are problem dependent. Currently, there is no fully automated method to find the optimal value for the two parameters. For the examples in Sec. IV, we first choose a value for ρ_j^0 by assigning a value near 0.15 ± 0.05 $(R_j \in G_D)$ or 0.55 ± 0.05 $(R_j \in G_E)$. We then vary $\gamma_j^{\max}$ to find an estimate with the lowest parameter.

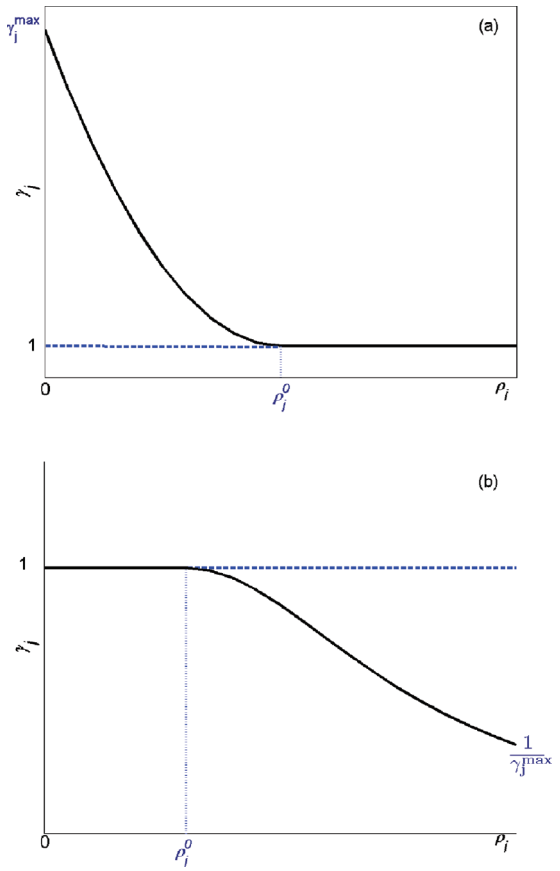

FIG. 2. (a) A graphical representation of $\gamma_j(\rho_j(\mathbf{x}))$ for $R_j \in G_E$. (b) A graphical representation of $\gamma_j(\rho_j(\mathbf{x}))$ for $R_j \in G_D$.

After incorporating the new strategy for choosing γ_j, we obtain the algorithm in Fig. 3.

IV. NUMERICAL EXAMPLES

In this section we illustrate our new biasing algorithm with three examples, comparing the results with those obtained using the original algorithm. As we will see, the new biasing method increases the computational speed not only relative to the SSA but also relative to the scheme used in the original wSSA papers.[1,2] The measure used to calculate gain in computational efficiency is the same as in Ref. 2, given by

$$g \equiv \frac{n^{\text{SSA}}}{n^{\text{wSSA}}}, \tag{13}$$

where n^{SSA} and n^{wSSA} are the numbers of runs in each of the two methods required to achieve comparable accuracy.

A. Two-state conformational transition

Consider the following system:

$$A \underset{k_2}{\overset{k_1}{\rightleftarrows}} B, \quad \text{with } k_1 = 0.12 \text{ and } k_2 = 1. \tag{14}$$

The initial state is set to $\mathbf{x}_0 = [100\ 0]$, i.e., all 100 molecules are initially in A form. This model concerns two conformational isomers—isomers that can be interconverted by rotation about single bonds. For this system we are interested in $p(0,30;10)$ for B; that is, the probability that given no B

Algorithm 2 wSSA procedure using state-dependent IS parameters

1° Partition all reactions into three groups: G_E, G_D, and G_N
2° **for all** $R_j \in G_E$ **do**
3° choose $\rho_j{}^0 \in [0.5, 0.6]$
4° choose the initial value for γ_j^{max}
5° **end for**
6° **for all** $R_j \in G_D$ **do**
7° choose $\rho_j{}^0 \in [0.1, 0.2]$
8° choose the initial value for γ_j^{max}
9° **end for**
10° **for all** $R_j \in G_N$ **do**
11° $\gamma_j = 1$ $\{b_j(t) = a_j(t)$ for $\forall\ t\}$
12° **end for**
13° $m_n^{(1)} \leftarrow 0, m_n^{(2)} \leftarrow 0$
14° **for** k=1 to n **do**
15° $s \leftarrow 0, \mathbf{x} \leftarrow \mathbf{x}_0, w \leftarrow 1$
16° evaluate all $a_i(\mathbf{x})$; calculate $a_0(\mathbf{x})$
17° calculate $\rho_j(\mathbf{x})$ for all $R_j \in G_E$ and all $R_j \in G_D$
18° calculate all $\gamma_j(\mathbf{x})$; evaluate $b_i(\mathbf{x})$; calculate $b_0(\mathbf{x})$
19° **while** $s \leq t$ **do**
20° **if** $\mathbf{x} \in \epsilon$ **then**
21° $m_n^{(1)} \leftarrow m_n^{(1)} + w,\ m_n^{(2)} \leftarrow m_n^{(2)} + w^2$
22° break out of the while loop
23° **end if**
24° generate two unit-interval uniform random numbers r_1 and r_2
25° $\tau \leftarrow a_0^{-1}(\mathbf{x}) \ln(1/r_1)$
26° $j \leftarrow$ smallest integer satisfying $\sum_{i=0}^{j} b_i(\mathbf{x}) \geq r_2 b_0(\mathbf{x})$.
27° $w \leftarrow w \times (a_j(\mathbf{x})/b_j(\mathbf{x})) \times (b_0(\mathbf{x})/a_0(\mathbf{x}))$
28° $s \leftarrow s + \tau,\ \mathbf{x} \leftarrow \mathbf{x} + \nu_j$
29° update $a_i(\mathbf{x})$ and $\rho_j(\mathbf{x})$
30° recalculate $\gamma_j(\mathbf{x})$; update $b_i(\mathbf{x})$; recalculate $b_0(\mathbf{x})$
31° **end while**
32° **end for**
33° $\sigma^2 = \left(m_n^{(2)}/n\right) - \left(m_n^{(1)}/n\right)^2$
34° repeat from 13 using different γ_j^{max} values **or** from 1 using different b functions to minimize σ^2
35° estimate $p(\mathbf{x}_0, \epsilon; t) = m_n^{(1)}/n$ with a 68% uncertainty of $\pm\sigma/\sqrt{n}$

FIG. 3. The weighted stochastic simulation algorithm using state-dependent importance sampling parameters. Changes from Fig. 1 are highlighted in red.

molecules at time 0, its population reaches 30 before time 10. The steady-state population of B for the rate constants in Eq. (14) is approximately 11. The rare event description of $x_2=30$ is about three times its steady-state value, so we expect its probability to be very small. Because this is a simple closed system, it is possible to calculate the exact probability of $p(0,30;10)$ using a generator matrix (or a probability transition matrix).[4] Using MATLAB's matrix exponential function, we have calculated $p(0,30;10)=1.191\times10^{-5}$.

Since the steady-state population of B is much less than the rare event description of 30, it is necessary to bias the system so that the population of B increases. This can be achieved by either encouraging R_1 or discouraging R_2. Note that changing the propensities of both reactions is not necessary, as only the relative ratio of the two propensities matters in reaction selection.

For simulation of system (14) with the original wSSA, R_1 was encouraged with different values of $\gamma_1 > 1$. Thus, the b functions are given by

$$b_1(\mathbf{x}) = \gamma_1 a_1(\mathbf{x}), \quad b_2(\mathbf{x}) = a_2(\mathbf{x}). \tag{15}$$

To find the γ_1 that produces the minimum variance, γ_1 was varied from 1.05 to 1.65 in increments of 0.05. Of these values, $\gamma_1=1.4$ produced the lowest variance. Taking $n=10^7$ and $\gamma_1=1.4$, the following estimate $\pm$ twice the uncertainty (95% confidence interval) was obtained:

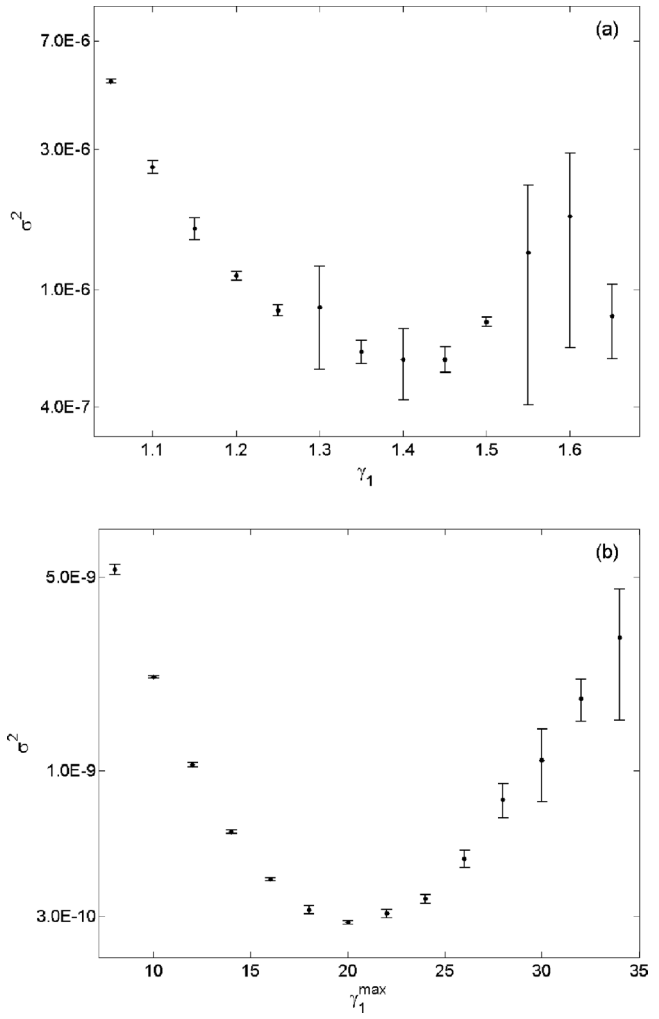

FIG. 4. (a). A plot of σ^2 vs γ_1 obtained in wSSA runs of reaction Eq. (14) using the algorithm in Fig. 1. Here we estimate $p(0,30;10)$ for B. Each vertical bar shows the estimated mean and one standard deviation of σ^2 at that γ_1 value as found in four $n=10^7$ runs. (b) A plot of σ^2 vs $\gamma_1^{\max}$ obtained from using the algorithm in Fig. 3. Each vertical bar was obtained from four $n=10^5$ runs.

$$p(0,30;10) = 1.21 \times 10^{-5} \pm 0.06 \times 10^{-5}. \tag{16}$$

In the new biasing method, we replicated the reaction partition as done in the original system, i.e., $G_E=\{R_1\}$, $G_D=\{\varnothing\}$, $G_N=\{R_2\}$. After assigning ${\rho_1}^0=0.5$, $\gamma_1^{\max}$ was varied from 8 to 34 in increments of 2. Using $n=10^5$ and the optimal $\gamma_1^{\max}=20$, the following estimate was obtained with a 95% confidence interval:

$$p(0,30;10) = 1.190 \times 10^{-5} \pm 0.011 \times 10^{-5}. \tag{17}$$

Although both estimates contain the true probability in their 95% confidence interval, the new biasing method required only 1% of the total number of simulations used in the original algorithm. Furthermore, the new biasing method produced a more accurate estimate with a much tighter confidence interval. Figure 4 shows a side by side comparison of σ^2 for $p(\mathbf{x}_0,30;10)$ using both biasing methods. We note that the new algorithm not only decreased the variance by a factor of 1000 but also produced an accurate estimate with a larger range of $\gamma_1^{\max}$. Given $n=10^7$, the previous algorithm was able to produce an accurate estimate with a narrow range of $\gamma_1 \in [1.10\ \ 1.60]$. In contrast, any estimate from using the new biasing method with $\gamma_1^{\max} \in [8\ \ 34]$ produced more accurate estimate than Eq. (16).

Using the SSA to obtain an estimate with similar variance as in Eq. (17) would require a much greater computational expense. For our particular simulation, the calculated efficiency gain of the new algorithm over the SSA was 4.1×10^4, i.e., it would have taken a computer 4.1×10^4 times longer to obtain a similar result using the unweighted SSA.

B. Single species production and degradation

Our next example is taken from Refs. 1 and 2 and consists of the following two reactions:

$$S_1 \xrightarrow{k_1} S_1 + S_2, \quad S_2 \xrightarrow{k_2} \varnothing \quad \text{with} \quad k_1 = 1 \quad \text{and} \quad k_2 = 0.025. \tag{18}$$

The initial state of the system is $\mathbf{x}_0=[1\ \ 40]$, and we are interested in $p(40,80;100)$ for S_2—the probability that x_2 reaches 80 before time 100 given $x_2=40$ at $t=0$. This particular reaction set is well-studied, and it is known that the population of S_2 in its steady-state follows the Poisson distribution with mean (and variance) of k_1x_1/k_2. Since the initial state of S_2 is also its steady-state mean, x_2 is expected to fluctuate around 40 with a standard deviation of 6.3. To advance the system toward the rare event, R_1 was chosen to be encouraged in both of the following wSSA simulations.

First, 4×10^7 wSSA simulations were performed for each γ_1 ranging from 1.45 to 2.25 in increment of 0.05. The following estimate was obtained using the optimal $\gamma_1=1.85$:

$$p(40,80;100) = 2.985 \times 10^{-7} \pm 0.020 \times 10^{-7} \quad (95\%\ \text{confidence}). \tag{19}$$

We repeated the simulation with the new state-dependent biasing method which encouraged R_1 as was done with the original algorithm. After running only 10^5 simulations with ${\rho_1}^0=0.6$ and different values of $\gamma_1^{\max}$, we obtained the following result using the optimal $\gamma_1^{\max}=14$:

$$p(40,80;100) = 2.986 \times 10^{-7} \pm 0.019 \times 10^{-7} \quad (95\%\ \text{confidence.}) \tag{20}$$

Figure 5 shows a side by side comparison of the variance using both biasing methods. As is shown, the biasing method with a state-dependent importance sampling parameter yielded estimates with variance two orders of magnitude less than that produced using the constant parameter importance sampling parameter. We also note that the latter simulation required 100 times fewer simulations to obtain a result with equivalent accuracy and uncertainty (20), which is a significant improvement over Eq. (19). Furthermore, the variance of an estimate generated by using any $\gamma_1^{\max} \in (4,26)$ is lower

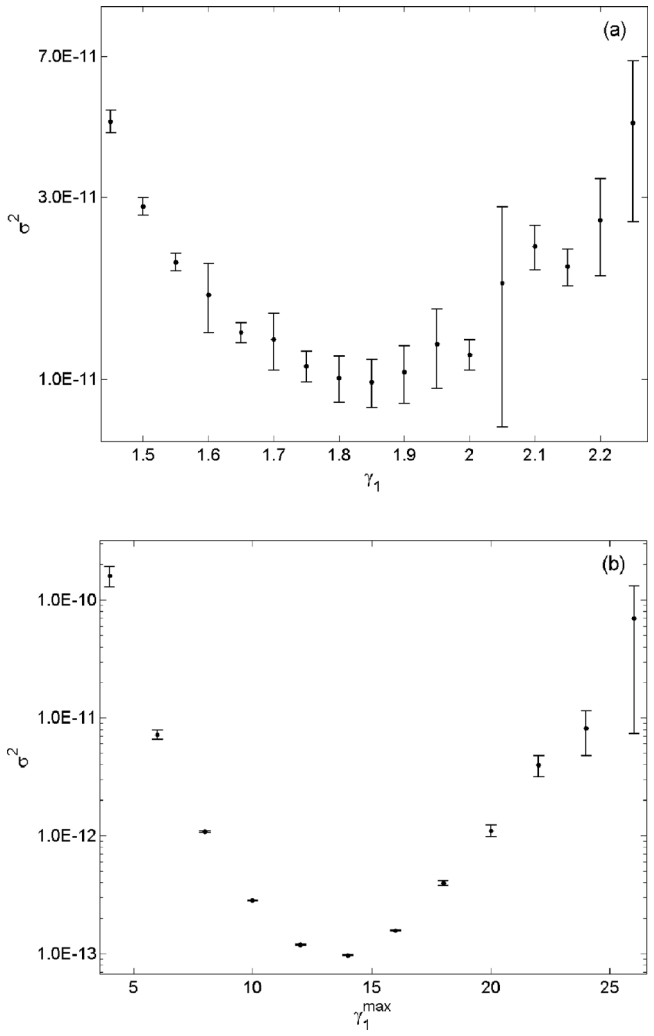

FIG. 5. (a) A plot of σ^2 vs γ_1 obtained in wSSA runs of reaction (18) using the algorithm in Fig. 1. Here we estimate $p(40,80;100)$ for S_2. Each vertical bar shows the estimated mean and one standard deviation of σ^2 at that γ_1 value as found in four $n=4\times10^7$ runs. (b) A plot of σ^2 vs $\gamma_1^{\max}$ obtained from using the algorithm in Fig. 3. Each vertical bar was obtained from four $n=10^5$ runs.

than the variance of estimate (19). Lastly, the computational efficiency gain of Eq. (20) over SSA was 3.1×10^6, which is more than 100 times the gain from using the original biasing method.

C. Modified yeast polarization

Our last example concerns pheromone induced G-protein cycle in Saccharomyces cerevisia[5] with a constant population of ligand, $L=2$. The model description was modified from Ref. 5, such that the system does not reach equilibrium. There are six species in this model, $\mathbf{x}=[R\ G\ RL\ G_a\ G_{bg}\ G_d]$ and eight reactions as follows:

$$\begin{aligned}
&\text{R1: } \varnothing \xrightarrow{k_{R_s}} R,\\
&\text{R2: } R \xrightarrow{k_{R_d}} \varnothing,\\
&\text{R3: } L + R \xrightarrow{k_{RLL}} RL + L,\\
&\text{R4: } RL \xrightarrow{k_R} R,\\
&\text{R5: } RL + G \xrightarrow{k_{G_a}} G_a + G_{bg},\\
&\text{R6: } G_a \xrightarrow{k_{G_d}} G_d,\\
&\text{R7: } G_d + G_{bg} \xrightarrow{k_G} G,\\
&\text{R8: } \varnothing \xrightarrow{k_{RL}} RL.
\end{aligned} \tag{21}$$

The kinetic parameters are

$$k_{R_s}=0.0038,\quad k_{R_d}=4.00e-4,\quad k_{RLL}=0.042,$$

$$k_R=0.0100,\quad k_{G_a}=0.011,\quad k_{G_d}=0.100,$$

$$k_G=1.05e+3,\quad k_{RL}=3.21.$$

The state representing the initial condition is $\mathbf{x}_0=[50\ \ 50\ \ 0\ \ 0\ \ 0\ \ 0]$—there are 50 molecules of R and G, but none of the other species are initially present. For this system, we define the rare event to be $p(\mathbf{x}_0,\varepsilon_{G_{bg}};20)$, where $\varepsilon_{G_{bg}}$ is the set of all states $\mathbf{x}$ for which the population of G_{bg} is equal to 50. We first partition the reactions as $G_E=\{\varnothing\}$, $G_D=\{R_6\}$, and $G_N=\{R_1,\ldots,R_5,R_7,R_8\}$. The only reaction chosen to be perturbed is R_6, which indirectly discourages the consumption of G_{bg} by delaying the production of G_d. Here we note that a more intuitive choice of reactions to include in G_D would be R_7, since it directly consumes a G_{bg} molecule. Upon numerical testing, however, we found that the estimate from perturbing R_7 showed much higher variance than the one obtained from discouraging R_6. This difference in performance is due to the four orders of magnitude separating the reaction constants k_{G_d} and k_G. Because the reaction constant k_G is so large, an extremely small IS parameter is required to effectively discourage R_7. The results from our testing indicate that such a small IS parameter confers high variance. In contrast, the IS parameter needed by R_6 was more modest and led to much better performance.

Following the partitioning with $G_D=\{R_6\}$, the b functions for the constant parameter biasing method are

$$\begin{aligned}
&b_6=\frac{1}{\gamma_6}a_6,\\
&b_j=a_j,\quad \mathrm{j}=1,2,3,4,5,7,8.
\end{aligned} \tag{22}$$

First, we ran 10^8 simulations of wSSA for each of the constant IS parameter γ_6, where γ_6 ranged from 1.2 to 2.0 in increments of 0.1. Then 10^8 simulations with the state-dependent biasing method was conducted for each $\gamma_6^{\max}=\{12,14,\ldots,22,24\}$ with $\rho_6{}^0=0.15$, where the b functions are given by

$$b_6=\gamma_6(\rho_6)a_6, \tag{23}$$

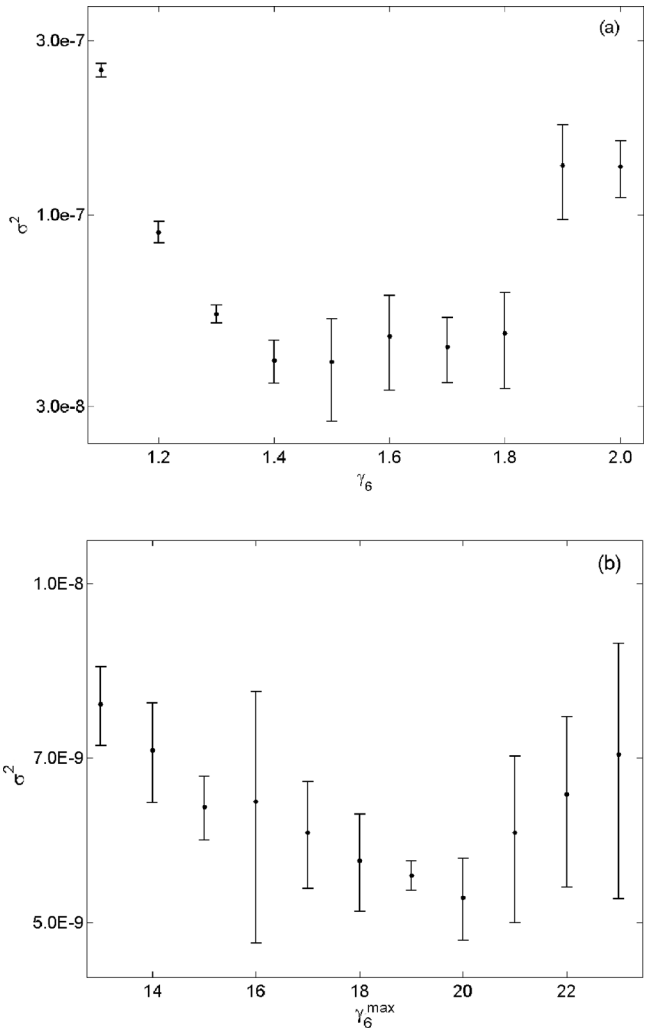

FIG. 6. (a) A plot of σ^2 vs γ_6 obtained in wSSA runs of reaction (21) using the algorithm in Fig. 1. Here we estimate $p(0,\varepsilon_{G_{bg}};20)$, where $\varepsilon_{G_{bg}}$ is the set of all states $\mathbf{x}$ in which the population of G_{bg} is 50. Each vertical bar shows the estimated mean and one standard deviation of σ^2 at that γ_6 value as found in four $n=10^8$ runs. (b) A plot of σ^2 vs γ_6^{max} obtained from using the algorithm in Fig. 3. Each vertical bar was obtained from four $n=10^8$ runs.

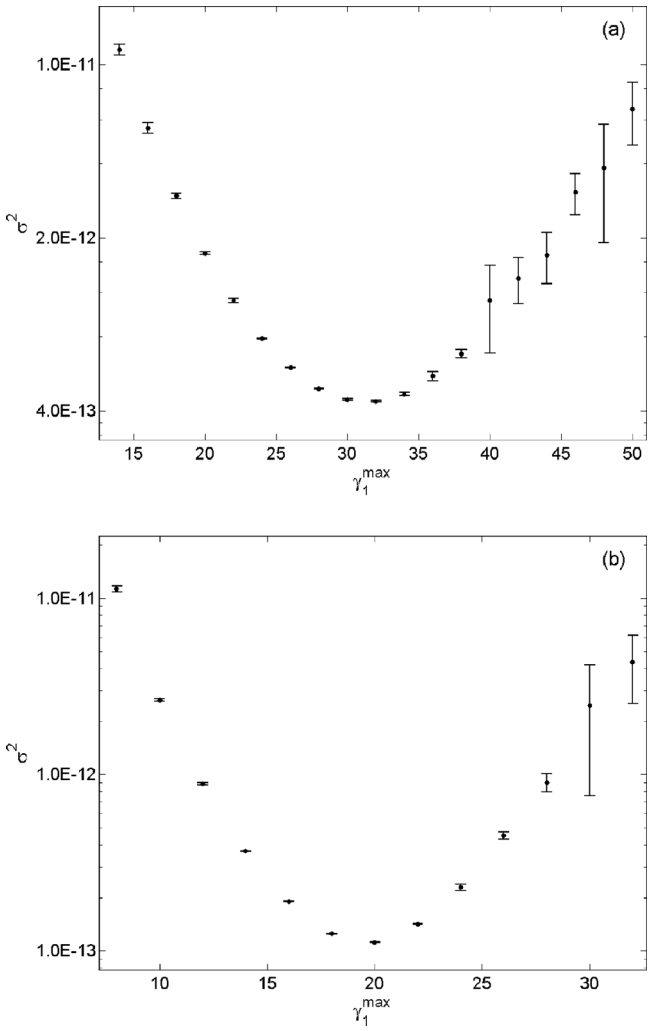

FIG. 7. (a) A plot of σ^2 vs γ_1^{max} obtained in wSSA runs of reaction (18) using $\rho_1^0=0.5$. Here we estimate $p(40,80;100)$ for S_2. Each vertical bar shows the estimated mean and one standard deviation of σ^2 at that γ_1^{max} value as found in four $n=10^5$ runs. (b) A plot of σ^2 vs γ_1^{max} obtained using the same setting as in Fig. 7(a), except $\rho_1{}^0=0.55$.

$$b_j = a_j, \quad j = 1,2,3,4,5,7,8.$$

Because $R_6 \in G_D$, the value of γ_6 at each time step is chosen according to $h_j(\rho_j)$ in Eq. (12). The best IS parameter, $\gamma = 1.5$, from the first set of simulations yields the following estimate:

$$p(\mathbf{x}_0,\varepsilon_{G_{bg}};20) = 1.23 \times 10^{-6} \pm 0.05 \times 10^{-6} \quad (95\% \text{ confidence}). \tag{24}$$

The best estimate from using the state-dependent IS method with $\gamma^{max}=3$ yields

$$p(\mathbf{x}_0,\varepsilon_{G_{bg}};20) = 1.202 \times 10^{-6} \pm 0.014 \times 10^{-6} \quad (95\% \text{ confidence}). \tag{25}$$

System (21) exhibits high intrinsic stochasticity, which causes difficulty in obtaining an estimate with low variance unless a large n is used. Despite this difficulty, we see that the uncertainty of the estimate obtained from using the state-dependent biasing algorithm is about three times less than the uncertainty in Eq. (24). Figure 6 shows a side by side comparison of σ^2 for $p(\mathbf{x}_0,\varepsilon_{G_{bg}};20)$, and we see that the variance in Fig. 6(b) is less than the variance in Fig. 6(a) by a factor of 10. In addition to increased accuracy, the new method also provides increased robustness, in that it provides a broader range of acceptable values for its parameter $\gamma_6^{max}(12-24)$ than for the old method's parameter $\gamma_6(1.3-1.8)$.

The computational gain of wSSA from using a constant importance sampling parameter is 21, while that of the state-dependent importance sampling method is 250. Both gains imply a significant speed up against the SSA considering that the simulation time for 10^8 wSSA trajectories of system (21) is several days.

Lastly, we comment that although the value of $\rho_j{}^0$ is chosen arbitrarily from a specified range (0.15 ± 0.05 for $R_j \in G_D$ and 0.55 ± 0.05 for $R_j \in G_E$), the performance of the state-dependent biasing algorithm remains almost the same for different values of $\rho_j{}^0$. For $R_j \in G_E$, the lower and upper boundary values for $\rho_j{}^0$ are 0.5 and 0.6, respectively. Figure 5(b) of system (18) was obtained using $\rho_1{}^0=0.6$, which is the

upper boundary value of ρ_j^0 range. To compare the performance of the new algorithm for system (18) using different values of ρ_1^0, we have created a plot of variance versus $\gamma_1^{\max}$ with two different values of $\rho_1^0 \in \{0.5, 0.55\}$, which are lower boundary and median value [Figs. 7(a) and 7(b)]. We see that the minimum variance from both subplots of Fig. 7 is of similar magnitude compared to the minimum variance in Fig. 5(b). Similar observation can be made for the other two examples.

V. CONCLUSIONS

In this paper we have introduced a state-dependent biasing method for the weighted stochastic simulation algorithm. As numerical results from Sec. IV support, the new state-dependent biasing method improves the accuracy of a rare event probability estimate and speeds up the simulation time. While the state-dependent biasing method excels in many aspects, it involves twice the number of parameters than the constant parameter biasing method. Currently, there is no automated method to assign an appropriate value to these parameters, and thus the computational effort associated with this task can be challenging as system size increases. It may be thought that using a line instead of a parabola in Eq. (12) would simplify the algorithm. However, owing probably to the nonlinearity of propensities that involve more than one species, we have found that the parabola usually works better. We have also observed in our numerical experiments that using a line has a negative impact on robustness, as compared to a parabola.

As noted in Sec. III, the new biasing algorithm toggles between the original propensities and the biased propensities to select the next reaction, depending on the value of $\rho_j(\mathbf{x})$. Therefore, the wSSA using the state-dependent biasing method can be regarded as an efficient adaptive algorithm. However, the value for two parameters ρ_j^0 and $\gamma_j^{\max}$ must be determined prior to the simulation, and correctly partitioning reactions into three groups (G_E, G_D, and G_N) can be challenging for large systems. These issues will be explored in future work.

ACKNOWLEDGMENTS

The authors acknowledge with thanks financial support as follows: D.T.G. was supported by the California Institute of Technology through Consulting Agreement No. 102-1080890 pursuant to Grant No. R01GM078992 from the National Institute of General Medical Sciences and through Contract No. 82-1083250 pursuant to Grant No. R01EB007511 from the National Institute of Biomedical Imaging and Bioengineering, and also from the University of California at Santa Barbara under Consulting Agreement No. 054281A20 pursuant to funding from the National Institutes of Health. M.R. and L.R.P. were supported by Grant No. R01EB007511 from the National Institute of Biomedical Imaging and Bioengineering, DOE Grant No. DEFG02-04ER25621, NSF IGERT Grant No. DG02-21715, and the Institute for Collaborative Biotechnologies through Grant No. DFR3A-8-447850-23002 from the U.S. Army Research Office.

[1] H. Kuwahara and I. Mura, J. Chem. Phys. **129**, 165101 (2008).
[2] D. T. Gillespie, M. Roh, and L. R. Petzold, J. Chem. Phys. **130**, 174103 (2009).
[3] T. F. Chan, G. H. Golub, and R. J. Leveque, Am. Stat. **37**, 242 (1983).
[4] R. Nelson, *Probability, Stochastic Processes, and Queuing Theory: The Mathematics of Computer Performance Modeling* (Springer, New York, 1995).
[5] B. Drawert, M. J. Lawson, L. R. Petzold, and M. Khammash, J. Chem. Phys. **132**, 074101 (2010).