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

Min K. Roh\textsuperscript{1,a}, Dan T. Gillespie\textsuperscript{2,b} and Linda R. Petzold\textsuperscript{1,c}

\textsuperscript{1}Department of Computer Science, University of California Santa Barbara, Santa Barbara, California 93106, USA
\textsuperscript{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.

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\textsuperscript{1} to efficiently estimate the probabilities of rare events in stochastic chemical systems. The wSSA was developed to estimate \( p(x_0, \varepsilon; t) \), which is the probability that the system, starting at \( x_0 \), will reach any state in the set of states \( \varepsilon \) 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 IS 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 \( x_j \) is the number of molecules of species \( S_j \). The propensity function \( a_j \) of reaction \( R_j \) is defined so that \( a_j(\mathbf{x}) \) 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 \( a_0(\mathbf{x}) = \sum_{j=1}^{M} a_j(\mathbf{x}) \). The SSA is based on the fact that the probability that the next reaction will carry the system to \( \mathbf{x} + \mathbf{v}_j \), where \( \mathbf{v}_j \) is the state change vector for reaction \( j \), between times \( t+\tau \) and \( t+\tau + d\tau \) is

\[
\text{Prob}\{\mathbf{x} \rightarrow \mathbf{x} + \mathbf{v}_j \text{ in } (t+\tau, t+\tau + d\tau)\} = a_0(\mathbf{x}) e^{-a_0(\mathbf{x})\tau} d\tau \times \frac{a_j(\mathbf{x})}{a_0(\mathbf{x})}.
\]

In the direct method implementation of the SSA, we choose \( \tau \), 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 \( \tau \) 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-

---

\textsuperscript{a}Electronic mail: min.roh@gmail.com.
\textsuperscript{b}Electronic mail: gillespiedt@mailaps.org.
\textsuperscript{c}Electronic mail: petzold@cs.ucsb.edu.
pling parameter for \( R_i \). Choosing \( \gamma_j > 1 \) will make \( R_i \) 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

\[
a_0(x)e^{-\alpha_0(x)\tau}d\tau \times \frac{b_j(x)}{b_0(x)},
\]

(2)

where \( b_0(x) = \sum_{i=1}^{M} b_i(x) \). This biased probability can be restored by multiplying Eq. (2) by the weighting factor,

\[
w_j(x) = \frac{a_j(x)/a_0(x)}{b_j(x)/b_0(x)}.
\]

(3)

Together, we have

\[
\text{Prob}(x \to x + v_j \text{ in } (t + \tau, t + \tau + d\tau)) = a_0(x)e^{-\alpha_0(x)\tau}d\tau \times \frac{b_j(x)}{b_0(x)} \times \frac{a_j(x)/a_0(x)}{b_j(x)/b_0(x)}.
\]

(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 \( w_j(x) \) in Eq. (3), the entire trajectory needs to be weighted by \( w = \Pi_j w_j(x) \).

The aim of the wSSA is to estimate

\[
p(x_0, \epsilon; t) = \text{the probability that the system,}
\]

starting at time 0 in state \( x_0 \),

will first reach any state in the set \( \epsilon \)

before \( t \).

(5)

It is important to note that \( p(x_0, \epsilon; t) \) is not the probability that the system will reach some state in \( \epsilon \) at time \( t \), but will have reached that set at least once before time \( t \).

Estimating \( p(x_0, \epsilon; t) \) with the SSA is straightforward. After running \( n \) simulations of SSA, we record \( m_n \), the number of trajectories that reached any state in \( \epsilon \) before time \( t \). Since each trajectory in the ensemble is equally statistically significant, we can estimate \( p(x_0, \epsilon; t) \) as \( m_n/n \), which approaches the true probability as \( n \to \infty \). While estimating \( p(x_0, \epsilon; t) \) with the SSA is a simple procedure, an extremely large \( n \) is required to obtain an estimate with low uncertainty when \( p(x_0, \epsilon; t) \ll 1 \).

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 \( \sigma \) 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 \( \pm \sigma/\sqrt{n} \)). 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)

\[
\text{relative uncertainty} = \frac{\text{uncertainty}}{m_n/n} = \pm \sqrt{1 - (m_n/n)/m_n}.
\]

(6)

When \( p(x_0, \epsilon; t) \ll 1 \) and \( m_n/n \ll 1 \), Eq. (6) reduces to

\[
\text{relative uncertainty} \approx \pm \sqrt{\frac{1}{m_n}}.
\]

(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 \( \epsilon \) using SSA is \( 1/p(x_0, \epsilon; t) \), we need about \( 1/p(x_0, \epsilon; t) \times 10 000 \) SSA simulations, which quickly becomes computationally infeasible as \( p(x_0, \epsilon; t) \) 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 \( b(x) \), instead of the original propensities \( a(x) \). Since each trajectory in the wSSA has a different weight, we redefine \( m_n/n = m_n^{(1)}/n = (1/n)\sum_{k=1}^{n} w_k \), where \( w_k = 0 \) if the \( k \)th trajectory did not reach \( \epsilon \) before time \( t \). We also keep track of the second moment of the trajectory weights, \( m_n^{(2)}/n = (1/n)\sum_{k=1}^{n} w_k^2 \), 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 \( b(x) \) (12th in Algorithm 1), and the bias is corrected with an appropriate weight, \( (a_j(x)/b_j(x)) \times (b_0(x)/a_0(x)) \) (13th). 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 \( \gamma_j \), i.e., \( b_j(x) = \gamma_j a_j(x) \). The fixed multiplier facilitates the wSSA.
The current state underperturb frequently, one may waste a lot of computational power searching for an accurate estimate. Unless the initial value for $P_j$ can have only a single value which can avoid only one of the two cases. Thus, there will be very few values for $\gamma_j$ that produce an accurate estimate. Unless the initial value for $\gamma_j$ 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 $\gamma_j$ 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 $\gamma_j$ 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 $\gamma_j$ according to the current state $x$. Although an arbitrary function may be used to achieve such goal, we proceed as follows.

First, we partition reactions into three groups: $G_D$, $G_G$, 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 $\rho_j(x) = a_j(x)/a_0(x)$. $\rho_j(x)$ 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 $x$. Since the value of $\rho_j$ at each time step gives a qualitative indication of the amount of perturbation needed by $R_j$, the new biasing method will define $\gamma_j$ to be a function of $\rho_j$.

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 $b_j(x) > a_j(x)$ 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 $\rho_j$ at which no further encouragement is applied. Similarly for $R_j \in G_D$, no further discouragement is

| 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° $s \leftarrow 0, x \leftarrow x_0, w \leftarrow 1$ |
| 4° evaluate all $a_i(x)$ and $b_i(x)$; calculate $a_0(x)$ and $b_0(x)$ |
| 5° while $s \leq t$ do |
| 6° if $x \in \epsilon$ 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° $\tau \leftarrow a_0^{-1}(x) \ln(1/r_1)$ |
| 12° $j \leftarrow$ smallest integer satisfying $\sum_{i=0}^{j} b_i(x) \geq r_2 b_0(x)$. |
| 13° $w \leftarrow w \times (a_j(x)/b_j(x)) \times (b_0(x)/a_0(x))$ |
| 14° $s \leftarrow s + \tau, x \leftarrow x + \nu_j$ |
| 15° update $a_i(x)$ and $b_i(x)$; recalculate $a_0(x)$ and $b_0(x)$ |
| 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 $\sigma^2$ |
| 20° estimate $p(x_0, \epsilon; t) = m_n^{(1)}/n$ with a 68% uncertainty of $\pm \sigma/\sqrt{n}$ |

FIG. 1. The weighted stochastic simulation algorithm using constant importance sampling parameters.
necessary when \( \rho_j = 0 \). The value of \( \rho_j \) for which we stop the perturbation is defined as \( \rho_j^0 \), 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 \( \rho_j \) increases. This corresponds to \( \gamma_j \to 0 \) as \( \rho_j \to 1 \).

Based on above considerations, we take the \( j \)th IS parameter of \( R_j \in G_E \) as

\[
\gamma_j(\rho_j(x)) = \begin{cases} 
1 & \text{if } \rho_j(x) = \rho_j^0, \\
g_j(\rho_j(x)) & \text{if } \rho_j(x) < \rho_j^0 
\end{cases}
\]  

(8)

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

\[
g_j(0) = \gamma_j^{\max},
\]

\[
g_j(\rho_j^0) = 1,
\]

\[
g_j'(\rho_j^0) = 0.
\]

For \( R_j \in G_D \), we select \( \gamma_j \) as the following:

\[
\gamma_j(\rho_j(x)) = \begin{cases} 
1 & \text{if } \rho_j(x) = \rho_j^0, \\
1 - \frac{g_j(\rho_j(x))}{h_j(\rho_j(x))} & \text{if } \rho_j(x) > \rho_j^0,
\end{cases}
\]  

(10)

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

\[
h_j(1) = \gamma_j^{\max},
\]

\[
h_j(\rho_j^0) = 1,
\]

\[
h_j'(\rho_j^0) = 0.
\]

Graphical representations of the function \( \gamma_j(\rho_j(x)) \) 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 \( \rho_j^0 \), for each \( \gamma_j(\rho_j(x)) \). \( \gamma_j^{\max} \) is defined as the parameter associated with the maximum perturbation allowed on the \( j \)th reaction. Given the values for \( \gamma_j^{\max} \) and \( \rho_j^0 \), \( g_j(\rho_j) \), and \( h_j(\rho_j) \) as defined in Eqs. (9) and (11) are unique parabolas whose formula is

\[
g_j(\rho_j) = \left( 2 \max - 1 \right) (\rho_j^0 - \rho_j)^2 + 1 \quad (\text{if } R_j \in G_E),
\]

\[
h_j(\rho_j) = \left( \frac{\gamma_j^{\max}}{(\rho_j^0 - 1)} \right) (\rho_j^0 - \rho_j)^2 + 1 \quad (\text{if } R_j \in G_D).
\]

An important point to keep in mind is that perturbing \( a_j(x) \) 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 \( \rho_j^0 \), 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 \( \rho_j^0 \) by assigning a value near \( 0.15 \pm 0.05 \) (\( R_j \in G_D \)) or \( 0.55 \pm 0.05 \) (\( R_j \in G_E \)). We then vary \( \gamma_j^{\max} \) to find an estimate with the lowest parameter.

After incorporating the new strategy for choosing \( \gamma_j \), 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. The measure used to calculate gain in computational efficiency is the same as in Ref. 2, given by

\[
G = \frac{n^{\text{SSA}}}{n^{\text{wSSA}}},
\]

(13)

where \( n^{\text{SSA}} \) and \( n^{\text{wSSA}} \) 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:

\[
k_1 \quad A \rightleftharpoons B, \quad \text{with } k_1 = 0.12 \text{ and } k_2 = 1.
\]

(14)

The initial state is set to \( x_0 = [100 \ 0] \), 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 $\gamma_j^{max}$  
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 $\gamma_j^{max}$  
9° end for
10° for all $R_j \in G_N$ do
11° $\gamma_j = 1 \{b_j(t) = a_j(t) \text{ for } \forall t\}$  
12° end for
13° $m_n^{(1)} \leftarrow 0, m_n^{(2)} \leftarrow 0$
14° for k=1 to n do
15° $s \leftarrow 0, x \leftarrow x_0, w \leftarrow 1$
16° evaluate all $a_i(x)$ ; calculate $a_0(x)$
17° calculate $\rho_j(x)$ for all $R_j \in G_E$ and all $R_j \in G_D$
18° calculate all $\gamma_j(x)$; evaluate $b_i(x)$; calculate $b_0(x)$
19° while $s \leq t$ do
20° if $x \geq \epsilon$ 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° $\tau \leftarrow a_0^{-1}(x) \ln(1/r_1)$
26° $j \leftarrow$ smallest integer satisfying $\sum_{i=0}^{j} b_i(x) \geq r_2 b_0(x)$.
27° $w \leftarrow (a_j(x)/b_j(x)) \times (b_0(x)/a_0(x))$
28° $s \leftarrow s + \tau, x \leftarrow x + \gamma_j$
29° update $a_j(x)$ and $\rho_j(x)$
30° recalculate $\gamma_j(x)$; update $b_i(x)$; recalculate $b_0(x)$
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 $\gamma_j^{\max}$ values or from 1 using different $b$ functions to minimize $\sigma^2$
35° estimate $p(x_0, \epsilon; t) = m_n^{(1)}/n$ 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.

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 $\gamma_j^{max}$  
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 $\gamma_j^{max}$  
9° end for
10° for all $R_j \in G_N$ do
11° $\gamma_j = 1 \{b_j(t) = a_j(t) \text{ for } \forall t\}$  
12° end for
13° $m_n^{(1)} \leftarrow 0, m_n^{(2)} \leftarrow 0$
14° for k=1 to n do
15° $s \leftarrow 0, x \leftarrow x_0, w \leftarrow 1$
16° evaluate all $a_i(x)$ ; calculate $a_0(x)$
17° calculate $\rho_j(x)$ for all $R_j \in G_E$ and all $R_j \in G_D$
18° calculate all $\gamma_j(x)$; evaluate $b_i(x)$; calculate $b_0(x)$
19° while $s \leq t$ do
20° if $x \geq \epsilon$ 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° $\tau \leftarrow a_0^{-1}(x) \ln(1/r_1)$
26° $j \leftarrow$ smallest integer satisfying $\sum_{i=0}^{j} b_i(x) \geq r_2 b_0(x)$.
27° $w \leftarrow w \times (a_j(x)/b_j(x)) \times (b_0(x)/a_0(x))$
28° $s \leftarrow s + \tau, x \leftarrow x + \gamma_j$
29° update $a_j(x)$ and $\rho_j(x)$
30° recalculate $\gamma_j(x)$; update $b_i(x)$; recalculate $b_0(x)$
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 $\gamma_j^{\max}$ values or from 1 using different $b$ functions to minimize $\sigma^2$
35° estimate $p(x_0, \epsilon; t) = m_n^{(1)}/n$ 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). Using MATLAB’s matrix exponential function, we have calculated $p(0,30;10) = 1.191 \times 10^{-5}$.

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

$$b_1(x) = \gamma_1 a_1(x), \quad b_2(x) = a_2(x).$$

(15)

To find the $\gamma_1$ that produces the minimum variance, $\gamma_1$ 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:
of $\gamma_i \in [1.10 \ 1.60]$. 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 $4.1 \times 10^4$, i.e., it would have taken a computer $4.1 \times 10^4$ 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:

$$S_1 \rightarrow S_1 + S_2, \quad S_2 \rightarrow \emptyset \quad \text{with} \quad k_1 = 1 \quad \text{and} \quad k_2 = 0.025.$$  

(18)

The initial state of the system is $x_0 = [1 \ 40]$, 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 $k_1 x_1 / k_2$. 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, $4 \times 10^7$ wSSA simulations were performed for each $\gamma_1$ ranging from 1.45 to 2.25 in increments of 0.05. The following estimate was obtained using the optimal $\gamma_1 = 1.85$:

$$p(40,80;100) = 2.985 \times 10^{-7} \pm 0.020 \times 10^{-7}$$

(95% confidence).  

(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 $\gamma_1^{\max} = 14$:

$$p(40,80;100) = 2.986 \times 10^{-7} \pm 0.019 \times 10^{-7}$$

(95% confidence.).

(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...
fied from Ref. 5, such that the system does not reach equi-

deration of ligand, \( x \) from using the algorithm in Fig. 3. Each vertical bar was obtained from four

bar shows the estimated mean and one standard deviation of

method.

more than 100 times the gain from using the original biasing

over SSA was 3.1 \( \times 10^6 \). Lastly, the computational

efficiency gain of Eq. (19) over SSA was 3.1 \( \times 10^6 \), which is

than the variance of estimate (19). Lastly, the computational
efficiency gain of Eq. (20) over SSA was 3.1 \( \times 10^6 \), 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 modi-

fied from Ref. 5, such that the system does not reach equi-

librium. There are six species in this model, \( \mathbf{x} = \{ R, G, RL, G_a, G_{bg}, G_d \} \) and eight reactions as follows:

R1: \( \varnothing \rightarrow R \),

R2: \( R \rightarrow \varnothing \),

R3: \( L + R \rightarrow RL + L \),

\[ k_R \]

R4: \( RL \rightarrow R \),

R5: \( RL + G \rightarrow G_a + G_{bg} \),

R6: \( G_a \rightarrow G_d \),

R7: \( G_d + G_{bg} \rightarrow G \),

R8: \( \varnothing \rightarrow RL \).

The kinetic parameters are

\[ k_R = 0.0038, \quad k_{Rd} = 4.00 \times 10^{-4}, \quad k_{RLL} = 0.042, \]

\[ k_R = 0.0100, \quad k_{Ga} = 0.011, \quad k_{Gd} = 0.100, \]

\[ k_{G} = 1.05 \times 10^{-3}, \quad k_{RL} = 3.21. \]

The state representing the initial condition is \( \mathbf{x}_0 = [50, 50, 0, 0, 0] \) —there are 50 molecules of \( R \) and \( G \), but none of the other species are initially present. For this sys-

tem, we define the rare event to be \( p(\mathbf{x}_0, e_{Gbg}; 20) \), where \( e_{Gbg} \)

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 differ-

e in performance is due to the four orders of magnitude

separating the reaction constants \( k_{Gd} \) and \( k_{G} \). Because the

reaction constant \( k_{G} \) is so large, an extremely small IS pa-

rameter is required to effectively discourage \( R_7 \). The results

from our testing indicate that such a small IS parameter con-

fers 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 \( b \) func-

tions for the constant parameter biasing method are

\[ b_6 = \frac{1}{\gamma_6 a_6}, \]

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

First, we ran \( 10^8 \) simulations of wSSA for each of the

constant IS parameter \( \gamma_6 \), where \( \gamma_6 \) 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

\[ b_6 = \gamma_6(\rho_6) a_6. \]
dependent biasing algorithm is about three times less than the uncertainty of the estimate obtained from using the state-dependence algorithm in Fig. 3. Each vertical bar was obtained from four of all states $h_j$ according to found in four $n=10^6$ runs. Because $R_j\in G_D$, the value of $\gamma_j$ at each time step is chosen according to $h_j(p_j)$ in Eq. (12). The best IS parameter, $\gamma = 1.5$, from the first set of simulations yields the following estimate:

$$ p(x_0, e_{G_{bg}}; 20) = 1.23 \times 10^{-6} \pm 0.05 \times 10^{-6} $$

(95% confidence).

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

$$ p(x_0, e_{G_{bg}}; 20) = 1.202 \times 10^{-6} \pm 0.014 \times 10^{-6} $$

(95% confidence).

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 $\sigma^2$ for $p(x_0, e_{G_{bg}}; 20)$, 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 $\gamma^{\text{max}} = 12$–24 than for the old method’s parameter $\gamma_0 = 1.3$–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 $p_j^{(0)}$ 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 $p_j^{(0)}$. For $R_j \in G_E$, the lower and upper boundary values for $p_j^{(0)}$ are 0.5 and 0.6, respectively. Figure 5(b) of system (18) was obtained using $p_j^{(0)} = 0.6$, which is the
upper boundary value of $\rho_j^0$ range. To compare the performance of the new algorithm for system (18) using different values of $\rho_j^0$, we have created a plot of variance versus $\gamma_j^\text{max}$ with two different values of $\rho_j^0 \in \{0.5, 0.55\}$, 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 $\rho_j(x)$. Therefore, the wSSA using the state-dependent biasing method can be regarded as an efficient adaptive algorithm. However, the value for two parameters $\rho_j^0$ and $\gamma_j^\text{max}$ 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.