Accurate Stochastic Simulation via the Step Anticipation τ -Leaping (SAL) Algorithm

MARY SEHL,¹ ALEXANDER V. ALEKSEYENKO,¹ and KENNETH L. LANGE^{1,2,3}

ABSTRACT

Stochastic simulation methods are important in modeling chemical reactions, and biological and physical stochastic processes describable as continuous-time discrete-state Markov chains with a finite number of reactant species and reactions. The current algorithm of choice, τ -leaping, achieves fast and accurate stochastic simulation by taking large time steps that leap over individual reactions. During a leap interval $(t, t + \tau)$ in τ -leaping, each reaction channel operates as a Poisson process with a constant intensity. We modify τ -leaping to allow linear and quadratic changes in reaction intensities. Because our version of τ -leaping accurately anticipates how intensities change over time, we propose calling it the step anticipation τ -leaping (SAL) algorithm. We apply SAL to four examples: Kendall's process, a two-type branching process, Ehrenfest's model of diffusion, and Michaelis-Menten enzyme kinetics. In each case, SAL is more accurate than ordinary τ -leaping. The degree of improvement varies with the situation. Near stochastic equilibrium, reaction intensities are roughly constant, and SAL and ordinary τ -leaping perform about equally well.

Key words: algorithms, evolution, statistics, stochastic processes.

1. INTRODUCTION

MANY CHEMICAL AND BIOLOGICAL MODELS involve continuous-time Markov chains with a finite number of particle types. The particles interact via a finite number of reaction channels, and each reaction destroys and/or creates particles in a predictable way. The current paper takes a fresh look at the problem of simulating the behavior of such chains. Before we launch into simulation specifics, it is helpful to carefully define a typical process. If d denotes the number of particle types, then the chain follows the count vector X_t , whose i th component X_{it} is the number of particles of type i at time $t \geq 0$. We typically start the system at time 0 and let it evolve via a succession of random reactions. Let c denote the number of reaction channels. Channel j is characterized by an intensity function $r_j(x)$ depending on the current vector of counts x . In a small time interval of length s , we expect $r_j(x)s + o(s)$ reactions of type j to occur. Reaction j changes the count vector by a fixed integer vector v^j . Some components v_k^j of v^j may be positive, some 0, and some negative.

¹Department of Biomathematics, David Geffen School of Medicine, University of California, Los Angeles, Los Angeles, California.

²Department of Human Genetics, David Geffen School of Medicine, University of California, Los Angeles, Los Angeles, California.

³Department of Statistics, University of California, Los Angeles, California.

From the wait and jump perspective of Markov chain theory, we wait an exponential length of time until the next reaction. If the chain is currently in state $X_t = x$, then the intensity of the waiting time until the next reaction is $r_0(x) = \sum_{j=1}^c r_j(x)$. Once the decision to jump is made, we jump to the neighboring state $x + v^j$ with probability $r_j(x)/r_0(x)$.

Table 1 lists typical reactions, their intensities $r(x)$, and increment vectors v . In the table, S_i denotes a single particle of type i . Only the nonzero increments v_i are shown. The reaction intensities invoke the law of mass action (Higham, 2008) and depend on rate constants a_i . Because they are counting random variables, the components of X_t are necessarily nonnegative. Each discipline has its own vocabulary. Chemists use the name propensity instead of the name intensity and call the increment vector a stoichiometric vector. Physicists prefer creation to immigration. Biologists speak of death and mutation rather than of decay and isomerization.

Despite the variety of processes covered, the allowed chains form a subset of all continuous-time Markov chains. Chains with an infinite number of reaction channels or random increments are not allowed. For instance, most continuous-time branching processes do not qualify. Birth-death processes grow by budding and serve as useful substitutes for more general branching processes, where each particle produces a random burst of daughter particles at the time of its death. The two kinds of processes can be at least approximately reconciled by matching the means of their offspring distributions and lifetimes (Lange, 2003).

The wait and jump mechanism constitutes a perfectly valid method of simulating one of these chains. Gillespie first recognized the practicality of this approach in chemical kinetics (Gillespie, 1977). Although his stochastic simulation algorithm (SSA) works well in some contexts, it can be excruciatingly slow in others. Unfortunately, reaction rates can vary by orders of magnitude, and the fastest reactions dominate computational expense in stochastic simulation. For the fast reactions, stochastic simulation takes far too many small steps. The τ -leaping method subsequently introduced by Gillespie and others (Cao et al., 2006; Gillespie, 2001) takes larger less frequent steps and exploits the tendency of many reaction intensities to change slowly over time. Before describing how one can take advantage of this feature, it is worth mentioning the chemical master equations, which are just the forward equations of Markov chain theory by a different name.

Let $p_{xy}(t)$ denote the finite-time transition probability of going from state x at time 0 to state y at time t . The usual reasoning leads to the expansions

$$p_{xy}(t+s) = p_{xy}(t) \left[1 - \sum_{j=1}^c r_j(y)s \right] + \sum_{j=1}^c p_{x, y-v^j}(t) r_j(y-v^j)s + o(s). \quad (1)$$

Forming the corresponding difference quotient and sending s to 0 produce the master equations

$$\frac{d}{dt} p_{xy}(t) = \sum_{j=1}^c \left[p_{x, y-v^j}(t) r_j(y-v^j) - p_{xy}(t) r_j(y) \right] \quad (2)$$

with initial conditions $p_{xy}(0) = 1_{\{x=y\}}$. Only in special cases can the master equations be solved. In deterministic models where particle counts are high, one is usually content to follow mean particle counts.

TABLE 1. SOME EXAMPLES OF REACTION CHANNELS

Name	Reaction	$r(x)$	v
Immigration	$0 \rightarrow S_1$	a_1	$v_1 = 1$
Decay	$S_1 \rightarrow 0$	$a_2 x_1$	$v_1 = -1$
Dimerization	$S_1 + S_1 \rightarrow S_2$	$a_3 \binom{x_1}{2}$	$v_1 = -2, v_2 = 1$
Isomerization	$S_1 \rightarrow S_2$	$a_4 x_1$	$v_1 = -1, v_2 = 1$
Dissociation	$S_2 \rightarrow S_1 + S_1$	$a_5 x_2$	$v_1 = 2, v_2 = -1$
Budding	$S_1 \rightarrow S_1 + S_2$	$a_6 x_1$	$v_2 = 1$
Replacement	$S_1 + S_2 \rightarrow S_2 + S_2$	$a_7 x_1 x_2$	$v_1 = -1, v_2 = 1$
Complex	$S_1 + S_2 \rightarrow S_3 + S_4$	$a_8 x_1 x_2$	$v_1 = v_2 = -1$
Reaction			$v_3 = v_4 = 1$

Intensities $r(x)$ and increment vectors v for some typical reactions. Here, S_i denotes a single particle of type i , and a_i denotes the reaction rate constant.

The mean behavior $\mu(t) = \mathbb{E}(X_t)$ is then approximately modeled by the system of ordinary differential equations

$$\frac{d}{dt}\mu(t) = \sum_{j=1}^c r_j[\mu(t)]v^j, \quad (3)$$

known as the reaction rate equation (RRE). This equation represents a coarse grain approximation, in which stochastic fluctuations in reactant species are ignored. The RRE can be very useful and efficient in simulating systems in which there are large numbers of molecules present and concentrations are assumed to vary continuously in time.

The τ -leaping method (Cao et al., 2006; Gillespie, 2001) is a bridge from exact stochastic simulation to numerical solution of macroscale differential equations. It has the advantages of (a) skipping over events rather than simulating each and every one of them, (b) maintaining the intrinsic noise of Markov trajectories, and (c) capturing first and higher order moments accurately. However, it is harder to implement than SSA and requires careful monitoring. In this paper, we address the twin issues of linear and quadratic extrapolation of reaction intensities and better sampling of Poisson deviates. The first of these is the more important in practice, so we will relegate most of the discussion of the second to the Appendix.

2. METHODS

The key to understanding τ -leaping lies in substituting Poisson kinetics for Markovian kinetics (Kurtz, 1981). The reaction intensity $r_j(x)$ represents the instantaneous probability of reaction j occurring. If the reaction intensities depended on time rather than on the state of the system x , then we could run an independent Poisson process for each reaction channel and sum the results. In τ -leaping one does precisely this over short time intervals. To improve τ -leaping, we seek to anticipate the changes in the various intensities. This allows us to improve accuracy or take longer time steps. Unfortunately, intensities change stochastically while we must anticipate deterministically. The best we can do is anticipate future mean counts. Poisson sampling also becomes more complicated because we must deal with inhomogeneous Poisson processes. If we have such a process with intensity $\lambda(t)$ that depends on time rather than the state of the system, then the number of events over $(t, t + \tau)$ is Poisson distributed with mean

$$\omega(t, t + \tau) = \int_t^{t+\tau} \lambda(s) ds. \quad (4)$$

Our new versions of τ -leaping anticipate by expanding $r_j(X_s)$ in a Taylor's series around time t with base value $r_j(X_t)$. We will explore linear and quadratic approximations derived heuristically.

2.1. Ordinary τ -leaping

Let us first review the particulars of τ -leaping. Under the τ -leaping approximation, we set the intensity of reaction j equal to the constant $r_j = r_j(X_t)$. Over the time interval $(t, t + \tau)$, reaction channel j will fire $r_j\tau$ times on average. In accord with the Poisson process assumption, we update the particle count vector $X_t = x$ by the sum $X_{t+\tau} = x + \sum_{j=1}^c N_j v^j$, where the N_j are independent Poisson variates with means $r_j\tau$. As just noted, the reaction intensities change as the particle count vector X_s changes. Thus, we restrict the time increment $\tau > 0$ to sufficiently small values such that each intensity $r_j(x)$ suffers little change over $(t, t + \tau)$ (Cao et al., 2006; Gillespie, 2001). Before we discuss exactly how to achieve this, let us pass to our more sophisticated anticipation updates.

2.2. SAL

The simplest improvement is to expand each intensity to first order. Hence, we replace the constant intensity r_j of ordinary τ -leaping by the linear intensity

$$r_j(X_s) \approx r_j(X_t) + \frac{d}{dt}r_j(X_t)(s - t). \quad (5)$$

Using this approximation, reactions of type j occur during $(t, t + \tau)$ according to an inhomogeneous Poisson process with mean

$$\omega_j(t, t + \tau) = \int_0^\tau \left[r_j(X_t) + \frac{d}{dt} r_j(X_t) s \right] ds \quad (6)$$

$$= r_j(X_t) \tau + \frac{d}{dt} r_j(X_t) \frac{1}{2} \tau^2. \quad (7)$$

At time $t + \tau$, we have $X_{t+\tau} = X_t + \sum_{j=1}^c N_j v^j$, where the N_j are independent Poisson variates with means $\omega_j(t, t + \tau)$. This is all to the good, but how do we compute the time derivatives of $r_j(X_t)$? The most natural heuristic is to invoke the chain rule

$$\frac{d}{dt} r_j(x) = \sum_{k=1}^d \frac{\partial}{\partial x_k} r_j(x) \frac{d}{dt} x_k, \quad (8)$$

and set

$$\frac{d}{dt} x_k = \sum_{j=1}^c r_j(x) v_k^j \quad (9)$$

as dictated by the handy RRE (3). In most realistic models, the differential matrix $dr(x) = \left[\frac{\partial}{\partial x_k} r_j(x) \right]$ is sparse, with nontrivial entries that are constant or of low order in x (Table 1). One could object that the RRE does not perfectly capture the mean behavior of a system, but theoretical values for mean particle counts are seldom available, and we make do with the best possible substitute.

For quadratic approximation, we differentiate equation (8) and calculate

$$\frac{d^2}{dt^2} r_j(x) = \sum_k \frac{d}{dt} \frac{\partial}{\partial x_k} r_j(x) \frac{d}{dt} x_k + \sum_k \frac{\partial}{\partial x_k} r_j(x) \frac{d^2}{dt^2} x_k \quad (10)$$

$$= \sum_k \frac{\partial}{\partial x_k} \frac{d}{dt} r_j(x) \frac{d}{dt} x_k + \sum_k \frac{\partial}{\partial x_k} r_j(x) \sum_l \frac{d}{dt} r_l(x) v_l^i \quad (11)$$

based on equation (9), where

$$\frac{\partial}{\partial x_k} \frac{d}{dt} r_j(x) = \sum_m \frac{\partial^2}{\partial x_k \partial x_m} r_j(x) \frac{d}{dt} x_m. \quad (12)$$

In this case, the number of reactions of type j is sampled from a Poisson distribution with mean

$$\omega_j(t, t + \tau) = r_j(X_t) \tau + \frac{d}{dt} r_j(X_t) \frac{\tau^2}{2} + \frac{d^2}{dt^2} r_j(X_t) \frac{\tau^3}{6}. \quad (13)$$

2.3. Selection of time increments

To gain insight into how to choose the increment τ in SAL, let the small positive constant ε denote an acceptable relative change in propensity. Because ε is not a direct measure of accuracy, it should be chosen empirically. In any event, it seems reasonable to take the largest τ such that

$$\left| \frac{d}{dt} r_j(x) \right| \tau \leq \varepsilon r_j(x) \quad (14)$$

holds for all j (Cao et al., 2006). If $r_j(x) = 0$ is possible, then we might amend this to

$$\left| \frac{d}{dt} r_j(x) \right| \tau \leq \varepsilon \max\{r_j(x), a_j\}, \quad (15)$$

where a_j is the rate constant for reaction j . In each instance in Table 1, a_j is the smallest possible change in $r_j(x)$. In common with other τ -leaping strategies, we revert to the stochastic simulation update whenever the intensity $r_0(x)$ for leaving a state x falls below a pre-set threshold δ .

2.4. Negative population counts

In τ -leaping, the number of reaction events of each type within $(t, t + \tau)$ is Poisson distributed. When a reactant population becomes small, there is always a risk that it may be driven negative within the next leap. In SAL, we avoid negative populations by assessing at the end of each step whether any species counts are negative, and if so, successively contracting the time step τ until all specie counts are non-negative. Step contraction is achieved by multiplying the current τ by a constant α , which in our simulations equals $\frac{3}{4}$. The reaction firings are then thinned by random acceptance. Thus for each firing, we sample a uniform random number U and accept the firing when

$$U \leq \alpha \left[\frac{r_j(X_t)\tau + \frac{d}{dt}r_j(X_t)\frac{1}{2}\alpha\tau^2}{r_j(X_t)\tau + \frac{d}{dt}r_j(X_t)\frac{1}{2}\tau^2} \right] \quad (16)$$

The right-hand side of this inequality is just the fraction of the expected number of firings over $(t, t + \tau)$ attributable to firings over $(t, t + \alpha\tau)$. If after thinning, the negativity violation disappears, then we accept the contracted time increment. Otherwise, we contract and thin again, and so forth. Observe that this tactic has the tendency to diminish randomness and is apt to slightly bias sample paths.

2.5. Generation of Poisson deviates

Sampling of Poisson deviates is one of the key ingredients of τ -leaping and SAL. It is straightforward to generate a Poisson deviate X with a small mean μ . For large μ a rounded Gaussian deviate with mean and variance μ is often employed. Poisson deviates generated in this way fail to capture the slight skewness of the Poisson distribution. The appendix describes a mixture method that corrects this defect and retains the speed of the Gaussian method.

3. NUMERICAL EXAMPLES

As a test of SAL, we now consider some simple numerical examples. In each of these, we use a combination of SAL and ordinary stochastic simulation (SSA). With SAL, we use either the linear derivative correction (*l*-SAL) or the sum of the linear and quadratic corrections (*q*-SAL). Our composite algorithm proceeds as follows: Before taking a step from $X_t = x$, we calculate the cumulative intensity $r_0(x) = \sum_{j=1}^c r_j(x)$. If $r_0(x)$ is below a threshold δ , then we perform one step of SSA. Otherwise, we compute the differential matrix $dr(x)$, the rates of change $\frac{d}{dt}r_j(x)$ of the reaction intensities, and the time step τ . Once τ is set, we generate a Poisson number of reactions for each channel and apply the standard update to x to derive $X_{t+\tau}$. For the sake of comparison, we also include ordinary τ -leaping, which we now abbreviate as OTL. Note that for OTL we continue to determine τ by the leap criterion (15).

Example 1. Kendall's process

As our first test case, we apply SAL to Kendall's birth, death, and immigration process. In the time homogeneous case, this Markov chain is governed by the birth rate α per particle, the death rate μ per particle, and the overall immigration rate ν . The three reactions

governing the process represent birth, death, and immigration.

Starting with i particles at time 0, one can explicitly calculate (Lange, 2003) the mean number of particles at time t as

$$m_i(t) = ie^{(\alpha-\mu)t} + \frac{\nu}{\alpha-\mu} \left[e^{(\alpha-\mu)t} - 1 \right]. \quad (18)$$

This exact expression permits us to evaluate the accuracy of the competing forms of τ -leaping. For the sake of illustration, we take $t=4$, $i=5$, $\alpha=2$, $\mu=1$, and $\nu=\frac{1}{2}$. Particle counts are averaged over 10^5 trials. Table 2 lists the OTL, *l*-SAL, and *q*-SAL approximations to the mean, the computing time in seconds over all trials (for *q*-SAL), and the fraction of steps during which SSA is carried out rather than τ -leaping. The

TABLE 2. AVERAGE PARTICLE COUNTS IN KENDALL'S PROCESS

ε	OTL	l-SAL	q-SAL	True mean	Time, secs	SSA fraction
1.0000	152.593	250.102	288.088	299.790	7.720	0.971
0.5000	195.567	281.916	298.055	299.790	8.476	0.950
0.2500	234.085	293.890	299.744	299.790	10.650	0.909
0.1250	261.987	299.204	299.361	299.790	14.393	0.839
0.0625	279.727	299.445	298.900	299.790	19.188	0.728
0.0313	289.321	297.891	300.245	299.790	31.214	0.575
0.0156	293.257	299.067	299.330	299.790	47.125	0.407
0.0078	296.150	300.273	299.952	299.790	67.679	0.257

Average particle counts in Kendall's process obtained using OTL, *l*-SAL, and *q*-SAL methods at time $t = 4$ for various values of ε , the acceptable relative changes in propensity. Comparison with the true means demonstrates a clear improvement in accuracy using *l*-SAL and *q*-SAL. Computing times over 10^5 trials are listed for the *q*-SAL simulations. Here, the initial number of particles $n = 5$, the birth rate $\alpha = 2$, the death rate $\mu = 1$, and the immigration rate $\nu = \frac{1}{2}$. The proportion of steps that are carried out using SSA rather than τ -leaping are listed for the threshold constant $\delta = 100$. Larger leaps are taken for larger ε , causing the total number of leaps to decrease, and the fraction of SSA steps to increase.

threshold constant for performing an SSA step is $\delta = 100$. As we increase ε , the sizes of the τ leaps tend to be larger, and hence the total number of leaps tends to be smaller. For this reason, the fraction of steps attributable to SSA grows.

The table makes it evident that SAL outperforms OTL and that $\varepsilon \leq 0.125$ is a good choice in SAL. To drive home the point that the time $t = 4$ is representative, Figure 1 plots the exact mean $\mathbb{E}(X_t)$ and the OTL and *l*-SAL approximations to it as a function of time. The figure shows that OTL continues to deteriorate as time progresses. Figure 2 displays the frequency distributions generated by SSA, OTL, *l*-SAL, and *q*-SAL, using $\varepsilon = 1$ and time $t = 4$. It is clear that the *q*-SAL distribution is closest to the SSA distribution.

Example 2. A two-type branching process

Next we consider a continuous-time multiple-type branching process with two types, females and males. Females die with intensity μ and males with intensity ν . Births occur with intensity λ per female. At each birth, a mother bears a daughter with probability p and a son with probability $1 - p$. Let X_t represent the number of females at time t and Y_t the number of males. The four reactions governing this process may be represented as

FIG. 1. Average counts in Kendall's process over time. Plot of the mean number of particles in Kendall's process as a function of time. Comparing the simulation results under OTL and SAL with the theoretical means, it is clear that the OTL approximation seriously degrades over time.

FIG. 2. Frequency distributions of particle counts in Kendall's process. With $\varepsilon = 1$ and $t = 4$, we compare frequency distributions over 10^5 trials of Kendall's process using SSA, OTL, l -SAL, and q -SAL. The distribution obtained using q -SAL is closest to that of SSA.

$$X \rightarrow 2X, \quad X \rightarrow X + Y, \quad X \rightarrow 0, \quad Y \rightarrow 0. \quad (19)$$

The expected numbers of females and males turn out to be (Lange, 2003)

$$\mathbb{E}(X_t) = \mathbb{E}(X_0)e^{(\lambda p - \mu)t} \quad (20)$$

$$\mathbb{E}(Y_t) = \mathbb{E}(X_0)\frac{\lambda(1-p)}{\lambda p + \nu - \mu}e^{(\lambda p - \mu)t} + \left[\mathbb{E}(Y_0) - \mathbb{E}(X_0)\frac{\lambda(1-p)}{\lambda p + \nu - \mu} \right] e^{-\nu t} \quad (21)$$

For the choices $t = 4$, $X_0 = 5$, $Y_0 = 5$, $\mu = 1$, $\nu = 1$, $\lambda = 4$, and $p = 0.45$, Tables 3 and 4 record the OTL and SAL approximations to the exact means $\mathbb{E}(X_t) = 122.663$ and $\mathbb{E}(Y_t) = 149.901$ over 10^5 trials. In this example as in the last example, OTL consistently undershoots the mean. Time is again measured in seconds over all trials for q -SAL.

Example 3. Ehrenfest's process

In our first two examples, mean particle counts grow steadily. In contrast, Ehrenfest's process exhibits convergence to a stochastic equilibrium. In Ehrenfest's model there are n particles in a box that is divided in half by a rigid partition with a very small hole. During a short time interval s , each particle independently passes through the hole and changes sides with probability $\lambda s + o(s)$. If X_t denotes the number of particles on the left side, then the process is governed by two reactions

$$X \rightarrow X - 1, \quad X \rightarrow X + 1. \quad (22)$$

When we start with 0 particles on the left side of the box and n particles on the right side, the probability of having $X_t = j$ particles on the left side at time t is given by

$$p_{0j}(t) = \binom{n}{j} \left(\frac{1 - e^{-2\lambda t}}{2} \right)^j \left(\frac{1 + e^{-2\lambda t}}{2} \right)^{n-j}. \quad (23)$$

TABLE 3. AVERAGE COUNTS FOR FEMALES IN A TWO-TYPE BRANCHING PROCESS

ε	OTL, $\bar{X}_t$	l-SAL, $\bar{X}_t$	q-SAL, $\bar{X}_t$	Time, secs	SSA fraction
1.000	71.891	108.442	119.530	8.437	0.968
0.500	87.701	117.677	122.367	10.656	0.944
0.250	100.465	121.443	122.556	13.152	0.902
0.125	110.653	122.081	123.023	20.505	0.829
0.063	116.084	122.233	123.166	27.747	0.717
0.031	119.466	122.722	122.431	38.490	0.565
0.016	120.797	122.336	123.269	54.759	0.396
0.008	121.609	122.764	122.969	88.176	0.249

Mean number of females calculated using OTL, l-SAL, and q-SAL at the time $t=4$ of a continuous-time two-type branching process. The initial number of females = 5, the initial number of males = 5, the female and male death rates $\mu = \nu = 1$, the birth intensity $\lambda = 4$, and the probability of a daughter $p = 0.45$. Comparing these results with the true mean = 122.663 for the number of females, it is clear that l-SAL outperforms OTL, and q-SAL outperforms l-SAL. Computing times over 10^5 trials and fraction of leaps from SSA steps under q-SAL are also listed.

The expected number of particles on the left side is $\mathbb{E}(X_t) = n(1 - e^{-2\lambda t})/2$ (Lange, 2003). Table 5 summarizes the average particle counts over 10^5 trials for OTL and SAL. Here we take $n = 10$, $t = 1$, and $\lambda = 0.5$. The fraction of times SSA was used in OTL, l-SAL, and q-SAL was 0.0. Although SAL is again more accurate, the differences between OTL and SAL are modest. When we increase the time interval to $t = 4$ units, Table 6 shows that even these modest gains in accuracy disappear. Observe that at time $t = 4$ the average number of particles is 4.9084, very close to the equilibrium value of 5. It is also of note that adding the second-derivative correction does not increase accuracy at either time.

Formula (23) allows us to calculate the full distribution of the number of particles on the left-hand side of the box, given that we start with 0 particles there. Thus, we can compare the empirical distributions under OTL and SAL. Figure 3 plots the l_1 error between the true probabilities and their empirical counterparts as a function of ε . As expected, choosing ε smaller leads to shorter steps and better accuracy. Here we let Ehrenfest's process run for 0.5 seconds over 10^6 trials. Although l-SAL enjoys a smaller l_1 error than OTL, the improvement is not striking. More encouraging is l-SAL's reduction in computation time for a given accuracy.

The modest improvements seen with SAL in this example may be explained by the relatively constant behavior of the reaction intensity at equilibrium. This behavior is apt to be generic for systems close to

TABLE 4. AVERAGE COUNTS FOR MALES IN A TWO-TYPE BRANCHING PROCESS

ε	OTL, $\bar{Y}_t$	l-SAL, $\bar{Y}_t$	q-SAL, $\bar{Y}_t$
1.000	87.125	133.018	147.984
0.500	106.852	143.830	150.077
0.250	122.700	148.478	149.921
0.125	135.172	149.164	150.334
0.063	141.822	149.456	150.489
0.031	146.016	149.921	149.591
0.016	147.631	149.511	150.697
0.008	148.635	150.079	150.251

Mean number of males in a continuous-time two-type branching process, calculated over 10^5 trials of OTL, l-SAL, and q-SAL for varying ε values. Parameter values are as listed in Table 3. The true mean number of males at $t = 4$ is 149.901.

TABLE 5. AVERAGE COUNTS OF LEFT-SIDE PARTICLES
IN EHRENFEST'S PROCESS AT $t = 1$

ε	OTL	l-SAL	q-SAL	True mean	Time, secs
1.0000	3.5626	3.1360	3.2024	3.1606	0.520
0.5000	3.4000	3.1512	3.1821	3.1606	0.785
0.2500	3.2947	3.1598	3.1718	3.1606	1.525
0.1250	3.2331	3.1616	3.1645	3.1606	2.745
0.0625	3.1963	3.1577	3.1585	3.1606	5.785
0.0312	3.1832	3.1663	3.1663	3.1606	9.474
0.0156	3.1695	3.1596	3.1595	3.1606	21.429
0.0078	3.1683	3.1635	3.1634	3.1606	37.306

Average left-side particle counts for Ehrenfest's process over 10^5 trials of OTL, l-SAL, and q-SAL for varying ε . The initial number of particles $n = 10$ and the intensity of a particle changing sides $\lambda = 0.5$. Computational times are listed for q-SAL; in every case the fraction of SSA steps was 0.

equilibrium. Because we expect the derivative $\frac{d}{dt}\mu_k(t)$ of the mean number of particles $\mu_k(t)$ of type k to be approximately zero at equilibrium, our approximate rate of change

$$\frac{d}{dt}x_k = \sum_{j=1}^c r_j(x)\nu_k^j \quad (24)$$

must be close to zero as well. This translates into small time derivatives for the reaction intensities $r_j(x)$. Thus, the derivative correction of SAL adds little to the projected constant intensities under OTL.

Example 4. *Michaelis-Menten enzyme kinetics*

Finally, we examine Michaelis-Menten enzyme kinetics, a process involving a combination of first- and second-order chemical reactions. In this system, there are four reactants: the substrate S_1 , the enzyme S_2 , the substrate-enzyme complex S_3 , and the product S_4 . The three reactions connecting them are

with rates a_1 , a_2 , and a_3 , respectively. We take the initial species counts to be 300 for the substrate, 120 for the enzyme, and 0 for the complex and product, and let $a_1 = 0.00166$, $a_2 = 0.001$, and $a_3 = 0.1$ (Higham, 2008; Wilkinson, 2006). Because no exact solution of the system is known, Tables 7 and 8 compare OTL and SAL's substrate and product averages $\bar{X}_t$ and $\bar{Y}_t$ at time $t = 20$ to the corresponding values 7.38 and 261.3 under SSA. In these simulations there were 10^5 trials. Again the fraction of times SSA was employed in the OTL, l-SAL, and q-SAL runs was 0.0. When we record averages at the range of time points (10, 20, 30, 40, 50, 60, 70), it is clear that SAL offers the greatest improvements in accuracy in the interval (20, 50).

TABLE 6. AVERAGE COUNTS OF LEFT-SIDE PARTICLES
IN EHRENFEST'S PROCESS AT $t = 4$

ε	OTL	l-SAL	q-SAL	True mean	Time, secs
1.0000	5.1657	4.5928	5.1293	4.9084	1.311
0.5000	5.0600	4.8077	4.9633	4.9084	1.817
0.2500	4.9737	4.9003	4.9220	4.9084	2.937
0.1250	4.9399	4.8881	4.8973	4.9084	4.464
0.0625	4.9234	4.9047	4.9054	4.9084	8.482
0.0313	4.9105	4.9000	4.8993	4.9084	16.831
0.0156	4.9081	4.9030	4.9037	4.9084	28.141
0.0078	4.9087	4.9074	4.9067	4.9084	61.245

Average left-side particle counts for Ehrenfest's process at $t = 4$. Parameter values are as listed in Table 5. As time progresses and the average number of particles approaches its equilibrium value of 5, there are fewer accuracy gains using SAL.

FIG. 3. l_1 distance as a function of ε . If we let Ehrenfest's process run until $t = 0.5$ and average over 10^6 trials, then l -SAL has a slightly smaller l_1 error than OTL. This suggests that ε , the acceptable relative change in propensity, can be increased without compromising accuracy. A larger ε improves computational efficiency.

3.1. Algorithm speed

We compare the speed of the SAL algorithm to that of OTL and SSA for each example. Please note that our implementation of OTL still requires time for calculation of derivatives even though it does not use these in calculating the mean number of reactions. Table 9 lists the required time to run 10^5 trials, with $\varepsilon = 0.125$ and time interval $t = 4$, for each of the examples. SAL has roughly the same speed as OTL, which is faster than SSA except in the case of Ehrenfest's model. Of course, SAL is far more accurate for this level of ε in three of the four examples.

4. DISCUSSION

The scope of the models covered by the SSA and τ -leaping methods is impressive, ranging from chemical kinetics to queuing theory, DNA regulation, membrane channel models, and cancer stem cell models. If we view transitions between states as reactions, then any continuous-time Markov chain with a

TABLE 7. AVERAGE COUNTS FOR SUBSTRATE
IN THE MICHAELIS-MENTEN SYSTEM

ε	OTL, $\bar{X}_t$	l-SAL, $\bar{X}_t$	q-SAL, $\bar{X}_t$	SSA, $\bar{X}_t$	Time, secs
1.000	57.7002	100.7422	70.3761	71.2141	5.761
0.500	61.5324	75.4466	70.6671	71.2141	3.169
0.250	64.3095	69.9145	71.5577	71.2141	4.881
0.125	66.4166	68.9941	71.3997	71.2141	8.938
0.063	67.8077	69.0972	71.1011	71.2141	12.972
0.031	68.7873	69.4214	71.0243	71.2141	23.139
0.016	69.4456	69.7238	71.0134	71.2141	40.426
0.008	69.9936	70.1095	70.9753	71.2141	72.036

Average number of substrate particles at $t = 20$ using OTL, l -SAL, q -SAL, and SSA over 10^5 trials. The initial number of substrate and enzyme species counts are 300 and 120, respectively, and the rate constants for the three reactions are $a_1 = 0.00166$, $a_2 = 0.001$, and $a_3 = 0.1$. Computational time is listed for simulations using the q -SAL method. SAL shows better agreement with SSA than OTL does.

TABLE 8. AVERAGE COUNTS FOR PRODUCT IN THE MICHAELIS-MENTEN SYSTEM

ε	$OTL, \bar{Y}_t$	$l\text{-}SAL, \bar{Y}_t$	$q\text{-}SAL, \bar{Y}_t$	$SSA, \bar{Y}_t$
1.000	167.6473	131.3892	161.3240	155.3903
0.500	165.3329	152.7160	159.4972	155.3903
0.250	162.8655	157.5086	158.9152	155.3903
0.125	160.9644	158.4267	158.6139	155.3903
0.063	159.5247	158.2868	158.2180	155.3903
0.031	158.4813	157.8885	157.7946	155.3903
0.016	157.7254	157.4508	157.3886	155.3903
0.008	157.0872	156.9798	156.8742	155.3903

Average number of product particles over 10^5 trials of OTL, l -SAL, q -SAL, and SSA. Parameter values are as listed in Table 7.

finite number of reaction channels is fair game. Branching process models do not fit the paradigm, but they can often be reformulated as birth-death-immigration processes that do.

Simulation by τ -leaping is one method in a hierarchy of methods. The recent review articles (Gillespie and Petzold, 2006; Higham, 2008) highlight the pervasive applications of particle-based stochastic models and situate τ -leaping between SSA and simulation methods for solving macroscopic stochastic differential equations such as the chemical Langevin equation. The hierarchy ends with the standard numerical methods for the ordinary differential equations of classical dynamical systems. Each level in this hierarchy operates on a different time scale. Higham (2008) stresses the importance of multiscale methods that handle reactions with different time scales efficiently. Any advance that permits a graceful transition from coarse-grained to fine-grain models is bound to be welcomed by the scientific community.

Our examples demonstrate that SAL can be substantially more accurate than OTL. Anticipating changes in reaction intensities also allows simulation to proceed by larger steps. In the previous literature (Gillespie, 2001; Anderson, 2007, 2008), the focus has been on taking smaller steps to achieve greater accuracy. When we account for inhomogeneity, we are able to take larger steps while maintaining accuracy. The method of step size selection we use is comparable to other methods (Cao et al., 2006), and we claim no credit for originality.

There is no reason to doubt the effectiveness of anticipation in large scale systems with many species, such as gene regulatory networks, tumor suppressor networks, and protein folding. We have avoided such systems in the current paper because they require lengthy explanations and almost always lack exact solutions for comparison. Our examples strongly suggest that SAL compares most favorably with OTL when specie counts are undergoing major changes far from equilibrium. Our examples all involve first or second order reaction kinetics. Although models with higher-order kinetics occasionally occur, they entail more rapid changes in intensities. In any event, SAL should enjoy an even greater advantage over OTL in dealing with them.

To describe our derivations as rigorous would stretch the definition of the word charitable. It will take hard work to prove rigorous error bounds. We are reluctant to conjecture the nature of these bounds. One of

TABLE 9. TIME (SECONDS) TO RUN EACH ALGORITHM FOR EXAMPLES 1–4

	SSA	OTL	$l\text{-}SAL$	$q\text{-}SAL$
Kendall's process	62.7	12.4	13.1	14.4
Two-type branching process	57.4	22.5	25.2	20.5
Ehrenfest's model	1.43	4.28	4.47	4.46
Michaelis-Menten reaction	14.4	8.51	7.42	7.29

Algorithm speeds for l -SAL and q -SAL with SSA and OTL over 10^5 trials. SAL has roughly the same speed as OTL. Both are faster than SSA except in the case of Ehrenfest's process.

the problems is that our projections rely on the RRE rather than on theoretical means. Another limitation of both OTL and SAL is their difficulty in handling extreme cases where a particle count hovers near 0 or some absolute maximum. Our step-back mechanism attempts to avoid some of the bias involved in simply rejecting forbidden steps. Nonetheless, we worry about its effects on accuracy and speed. A new method which provides a way around this issue (Anderson, 2008) involves representing reaction times as firing times of independent, unit rate Poisson processes, storing information gained from leaps, and enforcing leap conditions via a post-leap check. This method both naturally avoids negative population counts and avoids biasing the statistics of sample paths by rejection of leaps. Cao et al. (2005) also describe a variant of τ -leaping that divides reactions into critical and non-critical types. The former type consists of those that are likely to cause a negative particle count. Their algorithm only allows one critical reaction to fire in a single leap. There is nothing to prevent this useful elaboration from being combined with SAL.

Our implementations of OTL and SAL require that the relative change in each intensity function be uniformly bounded. This is consistent with the reasonable requirement that all intensity functions remain approximately constant. The original Gillespie-Petzold procedure (Gillespie and Petzold, 2003) deviates from this philosophy in the sense that the estimated change in each intensity function is bound by a specified fraction of the sum of all intensity functions. This choice has the unintended effect of allowing low reaction intensities to change by relatively large amounts and compromises simulation accuracy (Cao et al., 2006).

In summary, our derivative correction to τ -leaping provides a more accurate method for simulating chemical reactions and other stochastic processes. The increase in accuracy is greatest when particle counts are far from equilibrium, and the behavior of the system is changing rapidly. Readers interested in our still crude software should contact the first author. We anticipate that it will be helpful in simulating larger, more complex systems.

5. APPENDIX: FAST GENERATION OF POISSON DEVIATES

Both versions of τ -leaping rely on fast, accurate generation of Poisson random deviates. The quickest way to generate an approximate Poisson deviate with large mean λ is to round a normal random variable X with mean and variance λ to the nearest integer. The central limit theorem provides the rationale. There are two drawbacks to this scheme. Although quite improbable, the possibility $X < 0$ cannot be altogether ruled out. A more fundamental objection is that a Poisson deviate has a slightly positive skew while a normal random variable X has none. This defect in the normal prompts us to ask whether we can sample from another distribution that more closely mimics the Poisson. It turns out a gamma distributed random variable with shape parameter λ and scale parameter 1 is a suitable candidate. Once again the central limit theorem gives a good reason for making this choice.

Table 10 displays the moments of various distributions pertinent to sampling from a Poisson distribution with mean λ . It is evident from the table that the gamma distribution overcorrects skewness. If we turn to a half and half mixture of a normal and a gamma with the suggested parameters, then the first three moments exactly match those of the Poisson. The mixture column of the table confirms this fact. The kurtosis of the mixture distribution is twice what it should be, but this is less damaging because kurtosis varies with λ^{-1} rather than $\lambda^{-1/2}$.

TABLE 10. MOMENTS FOR VARIOUS RANDOM VARIABLES

<i>Moment</i>	<i>Poisson</i> (λ)	<i>Normal</i> (λ, λ)	<i>Gamma</i> ($\lambda, 1$)	<i>Mixture</i>
Mean	λ	λ	λ	λ
Variance	λ	λ	λ	λ
Skewness	$\lambda^{-1/2}$	0	$2\lambda^{-1/2}$	$\lambda^{-1/2}$
Kurtosis	λ^{-1}	0	$6\lambda^{-1}$	$3\lambda^{-1}$

Moments of various distributions pertinent to sampling from a Poisson distribution with mean λ . While the gamma distribution overcorrects skewness, the mixture method matches the Poisson distribution in its first three moments.

TABLE 11. COMPARISON OF METHODS FOR GENERATING POISSON DEVIATES

<i>Performance</i>	<i>Poisson</i>	<i>Normal</i>	<i>Gamma</i>	<i>Mixture</i>
TV ($\lambda = 10$)	0.00137	0.03969	0.04037	0.00580
Max k ($\lambda = 10$)	9	8	7	5
Time ($\lambda = 10$)	4.026	1.021	1.502	1.713
TV ($\lambda = 25$)	0.00210	0.02523	0.02540	0.00324
Max k ($\lambda = 25$)	29	22	21	24
Time ($\lambda = 25$)	4.287	1.131	1.652	1.842
TV ($\lambda = 50$)	0.00211	0.01792	0.01806	0.00284
Max k ($\lambda = 50$)	61	46	44	47
Time ($\lambda = 50$)	4.447	0.991	1.652	1.843

The mixture method provides a much more accurate approximation of the Poisson distribution than the gamma and normal methods over 10^6 random variates. TV denotes the empirical total variation distance. Max k refers to the index giving the greatest contribution to TV.

To derive the entries of the table, recall that skewness and kurtosis are defined by the formulas $k_3/k_2^{3/2}$ and k_4/k_2^2 , respectively, where k_n is the n -th cumulant of the distribution in question. If μ_n denotes the corresponding n th moment, then the formulas

$$\mu_1 = k_1 \quad (26)$$

$$\mu_2 = k_2 + k_1^2 \quad (27)$$

$$\mu_3 = k_3 + 3k_1k_2 + k_1^3 \quad (28)$$

$$\mu_4 = k_4 + 4k_1k_3 + 3k_2^2 + 6k_1^2k_2 + k_1^4 \quad (29)$$

and

$$k_1 = \mu_1 \quad (30)$$

$$k_2 = \mu_2 - \mu_1^2 \quad (31)$$

$$k_3 = \mu_3 - 3\mu_1\mu_2 + 2\mu_1^3 \quad (32)$$

$$k_4 = \mu_4 - 4\mu_1\mu_3 - 3\mu_2^2 + 12\mu_1^2\mu_2 - 6\mu_1^4. \quad (33)$$

allow one to convert between cumulants and moments and vice versa. The moments of the mixture distribution are equally weighted averages of the moments of their normal and gamma components.

These considerations are promising, but empirical validation is crucial. We have conducted a small simulation trial over a limited range of λ . For each value listed in Table 11 we sampled 10^6 independent random variates from the Poisson, normal, gamma, and mixture distributions. In the latter three cases, we rounded to the nearest integer greater than or equal to 0. The entry TV in the table is the total variation distance $\frac{1}{2} \sum_{k=0}^{\infty} |p_k - \hat{p}_k|$, where p_k is the Poisson probability $\lambda^k e^{-\lambda}/k!$ and $\hat{p}_k$ is its empirical estimate from the 10^6 trials. The entry **Max** k refers to the value k giving the largest absolute difference $|p_k - \hat{p}_k|$. The time required to generate all 10^6 deviates is given in seconds. Normal random deviates were generated by the polar method and gamma random deviates by the method of Marsaglia and Tsang (2000).

Table 11 clearly demonstrates the superiority of the mixture method over the normal and gamma methods. It is true that the mixture method takes almost twice as long as the normal method, but it is much more accurate and still easily beats Poisson sampling in speed. Of course, the overfall improvement is minor compared to the more fundamental improvement in τ -leaping wrought by anticipating linear trends in intensity.

ACKNOWLEDGMENTS

We would like to thank Dr. Janet Sinsheimer and Lindsay Riley for helpful comments and suggestions. This research is supported in part by USPHS (grants GM53275 and MH59490) and NIH (service award GM008185).

DISCLOSURE STATEMENT

No competing financial interests exist.

REFERENCES

- Anderson, D.F. 2007. A modified next reaction method for simulating chemical systems with time dependent propensities and delays. *J. Chem. Phys.* 127, 214107.
- Anderson, D.F. 2008. Incorporating post-leap checks in tau-leaping. *J. Chem. Phys.* 128, 054103.
- Cao, Y., Gillespie, D.T., and Petzold, L. R. 2005. Avoiding negative populations in explicit tau leaping. *J. Chem. Phys.* 123, 054104.
- Cao, Y., Gillespie, D.T., and Petzold, L.R. 2006. Efficient step size selection for the tau-leaping simulation method. *J. Chem. Phys.* 124, 1–11.
- Gillespie, D.T. 1977. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.* 81, 2340–2361.
- Gillespie, D. T. 2001. Approximate accelerated stochastic simulation of chemically reacting systems. *J. Chem. Phys.* 115, 1716–1733.
- Gillespie, D. T., and Petzold, L. R. 2003. Improved leap-size selection for accelerated stochastic simulation. *J. Chem. Phys.* 119, 8229–8234.
- Gillespie, D., and Petzold, L. 2006. Numerical Simulation for Biochemical Kinetics. In Szallasi, Z., Stelling, J., and Periwé, V., eds., *System Modelling in Cellular Biology*, MIT Press, Cambridge, MA, 1–29.
- Higham, D. J. 2008. Modeling and simulating chemical reactions. *SIAM Rev.* 50, 347–368.
- Kurtz, T.G. 1981. Approximation of population processes. SIAM, Philadelphia, 51–62.
- Lange, K.L. 2003. *Applied Probability*. Springer-Verlag, New York.
- Marsaglia, G., and Tsang, W. W. 2000. A simple method for generating gamma variables. *ACM Trans. Math. Software.* 26, 363–372.
- Wilkinson, D. J. 2006. *Stochastic Modelling for Systems Biology*. Chapman & Hall/CRC, Boca Raton, FL.

Address correspondence to:

Dr. Kenneth L. Lange
UCLA Biomathematics/Human Genetics
Box 951766, AV-515 CHS/5357A
Los Angeles, CA 90095-1766

E-mail: klange@ucla.edu