Proceedings of the 2016 Winter Simulation Conference
T. M. K. Roeder, P. I. Frazier, R. Szechtman, E. Zhou, T. Huschka, and S. E. Chick, eds.

EXACT SIMULATION VS EXACT ESTIMATION

Peter W. Glynn

Department of Management Science and Engineering
Stanford University
475 Via Ortega
Stanford, CA 94305, USA

ABSTRACT

This paper contrasts exact simulation against exact estimation in two different computational settings, namely that of numerical solution of stochastic differential equations and also in the context of equilibrium calculations for Markov chains. Both exact simulation and exact estimation methods can provide unbiased estimators capable of converging at square root rate in the computational effort c in problems in which conventional methods lead to sub-square root rates. We argue that the relaxation from exact simulation to exact estimation is often useful, because exact estimation algorithms can be easier to design and they can apply in settings in which exact simulation methods are currently unavailable.

1 INTRODUCTION

Suppose that we wish to compute $\alpha = Ef(X)$, where X is an S-valued random element and $f : S \to \mathbb{R}_+$ is a given performance functional defined on S. If $f(X)$ is easily generated by a random variate generation algorithm, computing α is straightforward conceptually and practically.

But there are a number of applications settings in which *exact simulation* of $f(X)$ is either algorithmically impossible or (very) challenging. For example, suppose that $X = (X(t) : t \geq 0)$ is the solution of a stochastic differential equation (SDE). As for deterministic differential equations, the standard numerical scheme includes a time-discretization that leads to a discrete-time approximation of the SDE that can then be simulated using forward time-stepping. Thus, while it is easy to exactly simulate approximations to the SDE, it is difficult to exactly simulate the SDE itself.

As a second example, suppose that $X : (X_n : n \geq 0)$ is a Markov chain (with stationary transition probabilities) for which X_∞ is a random variable (rv) having its equilibrium (or stationary) distribution π. In many applications contexts, computing $\alpha = Ef(X_\infty)$ is of central interest. Unfortunately, X_∞ involves the infinite-time behavior of X, so simulating X_∞ exactly is challenging. We note that in this Markov chain setting, exact simulation is equivalently called *perfect simulation*. As in the SDE setting, generating approximations to X_∞ is easy, however. In particular, if X is aperiodic, X_n has a distribution close to that of X_∞ when n is large.

As an alternative to exact simulation or approximations thereof, one can consider a problem relaxation in which one seeks a rv Y for which $EY = Ef(X_\infty)$. When such an unbiased rv Y can be generated in almost surely (a.s.) finite computer time, we say that α can be *exactly estimated*. This paper is concerned with contrasting exact simulation against exact estimation.

We note that when one has an exact simulation algorithm, this solves the exact estimation problem. But the converse is generally false. As we shall see, there are many settings in which developing an exact estimation algorithm is straightforward, whereas the corresponding exact simulation algorithm may be unavailable. Consequently, the concept of exact estimation is a useful relaxation of the concept of exact simulation.

Exact simulation has received a great deal of attention in the Markov chain setting. The first paper establishing the existence of such exact simulation algorithms was Asmussen, Glynn, and Thorisson (1992), while Propp and Wilson (1996) showed how such algorithms can be efficiently implemented in the finite state setting. The relaxation to exact estimation is more recent, and was introduced by Glynn and Rhee (2014).

Such exact algorithms are important, because biased estimation can create many algorithmic complications. The biggest problem is that bias is not reduced by simulating multiple independent replicates of the estimator, nor can its magnitude be easily assessed. Furthermore, in many settings, bias can be a dominant factor in determining the algorithm's rate of convergence. This is precisely the case in computational environments in which a large number of replications can be cheaply computed, such as that of multi-processor parallel simulation; see Glynn and Heidelberger (1991).

This paper focuses on the two main settings in which both exact simulation and exact estimation methods are available, namely the setting of stochastic differential equations and that of equilibrium computations for Markov chains. Section 2 discusses exact simulation for stochastic differential equations, while Section 3 describes its exact estimation counterparts. Similarly, Section 4 provides a discussion of the most commonly used algorithms for implementing exact simulation for Markov chains, and Section 5 describes exact estimation in the setting of Markov chain equilibrium computations.

It should be noted that the key idea underlying exact estimation, namely that of transforming an asymptotically unbiased sequence of estimators into an exactly unbiased estimator (see Section 3), applies more generally (outside the Markov chain setting). In particular, this idea can be applied to quantities that can be expressed as a nonlinear function of an expectation (as in Blanchet and Glynn (2015)). Further applications of this idea are certain to appear in the years ahead, given the generality of the method.

2 EXACT SIMULATION FOR SDE's

As might be expected, the development of efficient algorithms for solving the exact simulation problem depends heavily on problem structure. Consequently, there is no general toolbox for constructing exact simulation algorithms. Rather, an exact simulation algorithm needs to be tailored to the specific problem under consideration. In this section, we provide a brief account of how such exact simulation algorithms can be obtained in the SDE context.

We start by noting that if $x = (x(t) : t \geq 0)$ is an $\mathbb{R}^d$-valued solution to the deterministic differential equation

$$\frac{dx}{dt} = \mu(x(t)) \tag{1}$$

subject to $x(0) = x_0$, numerical schemes can typically not solve for x exactly. Somewhat surprisingly, it turns out that when noise is added to (1), thereby leading to an SDE, exact simulation is generally possible when $d = 1$. In particular, consider an $\mathbb{R}^d$-valued SDE of the form

$$dX(t) = \mu(X(t))dt + \sigma(X(t))dB(t), \tag{2}$$

subject to $X(0) = x_0$, where $\mu(\cdot)$ and $\sigma(\cdot)$ are given (deterministic) functions and B is an m-dimensional standard Brownian motion. The function $\mu(\cdot)$ is called the (infinitesimal) *drift* of X, and $\sigma(\cdot)$ is called the (infinitesimal) *volatility* of X. Under suitable technical conditions (e.g. global Lipschitz and growth conditions, on $\mu(\cdot)$ and $\sigma(\cdot)$; see Steele (2000)), (2) has a unique (strong) solution X.

Suppose that we can find a suitable invertible function $g : \mathbb{R}^d \to \mathbb{R}^d$ for which

$$dg(X(t)) = \underset{\sim}{\mu}(X(t))dt + d\tilde{B}(t) \tag{3}$$

for some $\underset{\sim}{\mu}$, where $\tilde{B}$ is a d-dimensional standard Brownian motion. In this case, we can set $Y(t) = g(X(t))$ and $\tilde{\mu}(\cdot) = \underset{\sim}{\mu} \circ g^{-1}$, and note that

$$dY(t) = \tilde{\mu}(Y(t))dt + d\tilde{B}(t). \tag{4}$$

In other words, the existence of such a transformation (called the *Lamperti transformation*) allows one to reduce the SDE to one with constant volatility.

To see how to choose g, we assume that g is smooth and apply Itô's formula, thereby yielding

$$dg(x(t)) = (\mathscr{L}g)(X(t))dt + \nabla g(X(t))\sigma(X(t))dB(t)$$

where $\mathscr{L}$ is the (partial) differential operator given by

$$\mathscr{L} = \sum_{i=1}^{d} \mu_i(x)\frac{\partial}{\partial x_i} + \frac{1}{2}\sum_{i,j=1}^{d}\sum_{k=1}^{m} \sigma_{ik}(x)\sigma_{jk}(x)\frac{\partial^2}{\partial x_i \partial x_j}.$$

Hence, we can obtain (3) if we select g so that

$$\nabla g(x)\sigma(x)\sigma(x)^T \nabla g(x)^T = I \tag{5}$$

for $x \in \mathbb{R}$. Note that (5) implies $d(d+1)/2$ constraints on the d partial derivatives of g. Hence, when $d > 1$, such a transformation g is typically unavailable (unless σ has very special structure). However, when $d = 1$, an invertible solution g to (5) is readily available:

$$g(x) = \int_{x_0}^{x} \frac{1}{\sigma(y)} dy.$$

Once the equation is in the form (4), we are now in a position to generate Y by use of acceptance-rejection ideas. A natural choice for the "candidate" is that associated with Brownian motion, since many functionals $f(X)$ can be exactly generated when $X = B$. We therefore seek a likelihood ratio $L(t)$ for which

$$P(Y_t \in d\omega) = L(t,\omega)P(B_t \in dw), \tag{6}$$

where $Y_t = (Y(s) : 0 \le s \le t)$ and $B_t = (B(s) : 0 \le s \le t)$.

Assume now that $d = m = 1$. Girsanov's formula (see, for example, Chapter 13 of Steele (2000)) asserts that

$$h(t) = \exp\left(\int_0^t \tilde{\mu}(B(s))dB(s) - \frac{1}{2}\int_0^t \tilde{\mu}^2(B(s))ds\right), \tag{7}$$

under suitable conditions on $\tilde{\mu}$. But the stochastic integral appearing in (7) can ,via Itô's formula, be expressed as

$$h(B(t)) - h(B(0)) - \frac{1}{2}\int_0^t h''(B(s))ds,$$

provided that we choose h so that $h'(x) = \tilde{\mu}(x)$ or, equivalently,

$$h(x) = \int_{x_0}^{x} \tilde{\mu}(y)dy + \tilde{\mu}(x_0).$$

By making this choice, we find that

$$L(t) = \exp\left(h(B(t)) - h(B(0)) - \frac{1}{2}\int_0^t \left[h'(B(s))^2 + h''(B(s))\right] ds\right).$$

If h and h'' are bounded, then

$$L(t) \le \exp\left(2||h||_\infty + \frac{1}{2}t||h''||_\infty\right) \triangleq \kappa_0,$$

where $||w||_\infty \triangleq \sup\{|w(s)| : x \in \mathbb{R}\}$. It follows that if U is a uniform rv on $[0,1]$ independent of B, then

$$P(Y_t \in \cdot) = P(B_t \in \cdot | U \leq L(t)/\kappa_0), \tag{8}$$

yielding, our acceptance-rejection algorithm.

However, testing the inequality $U \leq L(t)/\kappa_0$ requires that we be able to exactly generate the rv

$$\int_0^t \left[h'(B(s))^2 + h''(B(s))\right] ds$$

jointly with $B(t)$. Since this may be challenging for complicated choices for h, a further idea is needed. Write

$$L(t) = \frac{\exp\left(h(B(t)) - h(B(0))\right)}{\kappa_1} \exp\left(-\int_0^t \phi(B(s))\right) \exp(\kappa_2 t),$$

where κ_1 is chosen so that $\exp\left(h(B(t)) - h(B(0))\right)/\kappa_1 \leq 1$ and ϕ and κ_2 are chosen so that $\phi(x) \triangleq \left(h'(x)^2 + h''(x)\right)/2 + \kappa_2 \geq 0$ for $x \in \mathbb{R}$. If we let N be a unit rate Poisson process independent of B and U, we observe that

$$P\left(N\left(\int_0^t \phi(B(s))ds\right) = 0 | B, U\right) = \exp\left(-\int_0^t \phi(B(s))ds\right) \quad a.s. \tag{9}$$

The advantage of (9) is that we can generate $N\left(\int_0^t \phi(B(s))ds\right)$ by thinning (see Lewis and Shedler (1979)), so that $\phi(B(\cdot))$ need only be evaluated at the finite number of points at which the "dominating" Poisson process is thinned.

Specifically, let $T_1, T_2, \cdots$ be the event times of $(N(||\phi||_\infty s) : s \geq 0)$, and let $U_1, U_2, \cdots$ be a sequence of iid uniform $[0,1]$ rv's independent of U, B, and N. Then,

$$\exp\left(-\int_0^t \phi(B(s))ds\right) = P\left(U_1 > \frac{\phi(B(T_1))}{||\phi||_\infty}, \cdots, U_{\tilde{N}} > \frac{\phi(B(T_{\tilde{N}}))}{||\phi||_\infty} \middle| B, U\right) \quad a.s.,$$

where $\tilde{N} = N(||\phi||_\infty t)$. We conclude that

$$P(Y_t \in \cdot) = P\left(B_t \in \cdot \middle| U \leq \exp\left(h(B(t)) - h(B(0))\right)/\kappa_1, U_1 > \frac{\phi(B(T_1))}{||\phi||_\infty}, \cdots, U_{\tilde{N}} > \frac{\phi(B(T_{\tilde{N}}))}{||\phi||_\infty}\right) \tag{10}$$

Since it is easy to jointly generate $B(T_1), \cdots, B(T_{\tilde{N}})$, $B(t)$ conditional on $T_1, \cdots, T_{\tilde{N}}$, we conclude that (10) provides a practically implementable means of exactly simulating formulas at Y_t. In particular, we can now generate $f(Y_t)$ exactly, provided that we have the ability to jointly generate $f(Y_t), B(t_1), \cdots, B(t_n)$, and $B(t)$, for any selection of time points $0 < t_1 < \cdots < t_n < t$.

This elegant set of ideas was introduced by Beskos and Roberts (2005), and it has attracted much further research in the years since its introduction; see for example, Giesecke and Smelov (2013). Their initial paper made one further enhancement, by recognizing that h is typically unbounded, so that $\kappa_1 = \infty$ in many applications settings. Thus, they choose to use a modified candidate, involving the Brownian bridge process rather than Brownian motion.

As noted earlier, this idea depends upon the Lamperti transformation, so it requires either that $d = 1$ or that the multi-dimensional X have very special structure in its volatility specification. In the next section, we show how exact estimation provides the additional algorithmic flexibility needed to develop good estimation algorithms for multi-dimensional SDE's.

3 EXACT ESTIMATION FOR SDE's

In contrast to Section 2, the easiest means of exactly estimating functionals of SDE's is to start from the various discretization schemes that have been proposed historically for solving SDE's. Specifically, the idea is to approximate $X = (X(t) : t \geq 0)$ by a process $X_h = (X_h(t), t \geq 0)$ corresponding to a time-step discretization equal to h. The most obvious such discretization is the *Euler scheme* given by

$$X_h((k+1)h) - X_h(kh) = \mu(X_h(kh))h + \sigma(X_h(kh))(B((k+1)h) - B(kh))$$

for $h > 0$, thereby yielding X_h at the lattice points $0, h, 2h, \cdots$. To obtain a full continuous time path, one needs some mechanism for interpolating over each interval $[kh, (k+1)h]$. The easiest such interpolation is a linear interpolation, but other mechanisms are available also. For example, if one wishes to respect the Brownian fluctuations of the path within such intervals, one can interpolate using a Brownian bridge with a constant volatility given by (say) the average of the volatility values at the two end points. Whatever interpolation is used, we end up with a continuous time approximation $(X_h(t) : t \geq 0)$ to X. For a given functional $f(X)$, we therefore have an approximation $f(X_h)$.

We note that the approximation $f(X_h)$ is typically biased as an estimator of $Ef(X)$. For functionals f that take the form $f(X) = w(X(1))$ for some $w : \mathbb{R}^d \to \mathbb{R}_+$ (a so-called "final value" problem), the bias generally can be expressed as

$$Ef(X_h) = Ef(X) + bh^k + o(h) \tag{11}$$

as $h \downarrow 0$, where k is known as the *(weak) order* of the scheme (and $o(a(h))$ denotes a function $r(h)$ such that $r(h)/a(h) \to 0$ as $h \to 0$). Since it is usual that $X_h \Rightarrow X$ as $h \downarrow 0$ (where $\Rightarrow$ denotes weak convergence), it is generally the case that $Varf(X_h) \to Varf(X)$ as $h \downarrow 0$.Thus, if

$$\bar{W}_n(h) \triangleq \frac{1}{n}\sum_{i=1}^{n} f(X_h^i)$$

where $X_h^1, X_h^2, \cdots$ are iid replicates of X_h, the mean square error (MSE) is given by

$$MSE(\bar{W}_n(h)) = \frac{1}{n}Varf(X) + b^2h^{2k} + o(\frac{1}{n} + h^{2k}) \tag{12}$$

as $n \to \infty$ and $h \downarrow 0$.

Given the above expression for the MSE, it is now natural to seek the choice of n and h, for a given computational budget c, that minimizes the MSE. Fixing c effectively constrains $n/h = c$ (since the computational effort required to generate X_h over a unit amount of time scales in proportion to $1/h$, where h is the forward time-stepping increment). Subject to $n/h = c$, the minimizing prescription for n and h (in the presence of (12)) is to choose n to scale in proportion to $c^{\frac{2k}{2k+1}}$ and to let h decrease at rate $c^{-\frac{1}{2k+1}}$, in which case the MSE tends to 0 at rate $c^{-\frac{2k}{2k+1}}$. In other words, the root mean square error decays at rate $c^{-\frac{k}{2k+1}}$, in the presence of these optimal choices. For a related discussion, see Duffie and Glynn (1995).

A key observation is that the magnitude of k has an enormous impact on the rate of convergence of the associated numerical scheme. Thus, a second-order scheme (for which $k = 2$) converges much faster than does a first-order scheme (for which $k = 1$), and this motivates the extensive literature on higher order solution schemes; see Kloeden and Platen (1992). It is worth noting that $k \in (0, 1)$ is also possible; see Asmussen, Glynn, and Pitman (1995).

We next turn to exact estimation in the SDE context. We provide here a general account of a key idea that will be used not only in this section, but also later when we discuss exact estimation in the setting of Markov chains. We note that if $Y = f(X)$, we can view our discretization scheme as providing a family of approximations $Y_h = f(X_h)$ to Y. So, consider now a sequence $(Y_n : n \geq 0)$ of approximations to a rv Y in which $Y_n \to Y$ in L^2, where L^2 is the Hilbert space of square-integrable rvs, equipped with norm $||Z||_2 = \sqrt{EZ^2}$ for $Z \in L^2$.

Let $\Delta_n = Y_n - Y_{n-1}$ for $n \geq 0$ (with $Y_{-1} \triangleq 0$) and note that the L^2 convergence implies that $EY_n \to EY$ as $n \to \infty$. Consequently,

$$EY = \lim_{n\to\infty} \sum_{k=0}^{n} E\Delta_k,$$

Note that if N is a finite-valued non-negative integer-valued rv, independent of $(Y_n : n \geq 0)$, for which $P(N \geq n) > 0$ for $n \geq 0$, then $E\tilde{Z}_n = EY_n$ for $n \geq 0$, where

$$\tilde{Z}_n = \sum_{k=0}^{N\wedge n} \Delta_k / P(N \geq k),$$

where $a \wedge b \triangleq \min(a,b)$. If

$$\sum_{n=1}^{\infty} \frac{||Y_n - Y||_2^2}{P(N \geq k)} < \infty, \tag{13}$$

then $\tilde{Z}_n \to \tilde{Z}$ a.s. as $n \to \infty$, $E\tilde{Z} = EY$, $\tilde{Z} \in L^2$, and

$$E\tilde{Z}^2 = \sum_{n=0}^{\infty} \frac{\left(||Y_{n-1} - Y||_2^2 - ||Y_n - Y||_2^2\right)}{P(N \geq n)};$$

see Rhee and Glynn (2015) and the closely related paper by McLeish (2011). By replicating iid copies of $\tilde{Z}$, this gives us a general recipe for constructing finite variance unbiased estimators from sequences of biased estimators. Since such finite variance unbiased estimators satisfy central limit theorems (CLT's) with square root convergence rate, the above recipe therefore has the potential to turn estimation schemes with sub-square root convergence rates into ones that enjoy square root convergence. In other words, enormous potential improvements in convergence can be obtained using this approach. This estimation approach can be viewed as being a randomized version of what is known as multi-level Monte Carlo; see Heinrich (2001) and Giles (2008) for details.

In order to implement this strategy, we need to construct approximations $(Y_n : n \geq 0)$ such that the Y_n's are close to one another for n large. This suggests that the key to success in this approach is "coupling" the Y_n's in such a way that $||Y_n - Y_{n-1}||_2 \to 0$ as $n \to \infty$.

Fortunately, in the SDE context, this is easily accomplished. In particular, suppose that Y_n corresponds to $f(X_h)$ with $h = 2^{-n}$, and note that Y_n uses a time increment half the size of that for Y_{n-1}. The idea is to use the same Brownian motion across all the Y_n's. Specifically, one can first generate the Brownian increments at the finest time scale appearing in $\tilde{Z}$, namely $h = 2^{-N}$, and then simply aggregate the increments to obtain the Brownian terms that appear within the coarser scales associated with $Y_0, \cdots, Y_{N-1}$.

With this idea in hand, suppose that one uses the Euler discretization to construct the X_h's, together with linear interpolation between the time discretization points. Then, according to Kloeden and Platen (1992), p.341-344,

$$||Y_n - Y||_2^2 = O(2^{-n})$$

as $n \to \infty$, and we note that $Var(\tilde{Z}) < \infty$ if $P(N \geq n)$ is then chosen to be of order 2^{-np} for $p \in (0,1)$. As a consequence, one is guaranteed square root convergence rate in the number n of replicates of $\tilde{Z}$ that are generated. When one replicates $\tilde{Z}$ many times, this effectively means that one is sampling at different frequencies across the various time-discretizations. This idea is the fundamental concept underlying *multi-level Monte Carlo*; see Giles (2008) and Heinrich (2001). Thus, given the randomization in N, we can view this algorithm as a randomized MLMC method.

Of course, what we really want is a square root convergence rate in the computational effort c (as measured, say, in the number of floating point operations). Typically, square root convergence in n and c go hand-in-hand, because the expected computational effort per replicate usually is finite; see Glynn and Whitt (1992) for such a result. But our randomized MLMC scheme can unfortunately yield estimators

with infinite expected computational effort per observation. Note that the computational effort required to generate $\tilde{Z}$ scales in proportion to 2^N. But if $P(N \geq n)$ is of order of 2^{-np} for $0 < p < 1$, this implies that $E2^N = \infty$, and thus the expected computational effort per $\tilde{Z}$ replicate is infinite.

As a consequence, it is natural to explore modifying the discretization scheme that is used. In particular, suppose that one approximates X via X_h, where X_h corresponds to a Milstein approximation to X; see p.345 of Kloeden and Platen (1992) for a description. If Y_n is again associated with $f(X_h)$ with $h = 2^{-n}$, it turns out that

$$||Y_n - Y||_2^2 = O(2^{-2n}) \tag{14}$$

as $n \to \infty$. Consequently, $Var(\tilde{Z}) < \infty$ ensures, provided that we choose (for example) $P(N \geq n)$ of order 2^{-np} for $0 < p < 2$.

Turning next to the question of the expected computational effort per replicate of $\tilde{Z}$, the Milstein scheme requires generating the iterated Itô integrals

$$\int_0^t B_i(s) dB_j(s)$$

for $1 \leq i, j \leq d$. There is, unfortunately, no good algorithm available currently for generating such integrals when $i \neq j$. (Of course, when $i = j$, it is well known that this is given by $(B_i^2(t) - t)/2$.) Clearly, when $d = 1$, the Milstein scheme involves no such non-simulatable iterated Itô integrals. In this one dimensional setting, the computational effort scales in proportion to 2^N, as in the Euler context. It follows that we can obtain finite expected computational time per replication if we choose N so that $P(N \geq n)$ is of order 2^{-np} for $p > 1$. Hence, we obtain a square root convergence rate in c if $p \in (1,2)$. Thus, square root convergence rate in c for one dimensional SDE's can be attained using either exact simulation or exact estimation methods.

In Rhee and Glynn (2015), a recently developed method of Giles and Szpruch (2013) is discussed, known as antithetic MLMC, that offers the discretization error estimate (14) when the functional f is of a suitable form. This antithetic method applies to multi-dimensional SDE's, so provides a mechanism for obtaining square root convergence rates in the multi-dimensional setting, at least when f is appropriately chosen. In addition, the paper explores the issue of how to choose the randomization distribution N optimally, so as to maximize the rate of convergence; see Section 3 of Rhee and Glynn (2015).

4 EXACT SIMULATION OF MARKOV CHAINS

Our second illustration of the contrast between exact simulation and exact estimation lies in the setting of Markov chains, specifically in computing equilibrium quantities. We start by recalling that many Markov chains have embedded regenerative structure, in which case the equilibrium distribution π (or stationary distribution) can be expressed as

$$\pi(\cdot) = \frac{\tilde{E} \sum_{j=0}^{\tau-1} I(X_j \in \cdot)}{\tilde{E}\tau},$$

where τ is the first regeneration time of X, and $\tilde{E}(\cdot)$ is the expectation operator under which X is initialized so that the chain starts at $n = 0$ with a regeneration; see Asmussen (2003), p.178, for details. Hence, if $\tilde{P}(\cdot)$ is the probability associated with $\tilde{E}(\cdot)$,

$$\begin{aligned}
\pi(\cdot) &= \sum_{j=0}^{\infty} \tilde{P}(X_j \in \cdot | \tau > j) \frac{\tilde{P}(\tau > j)}{\tilde{E}\tau} \\
&= \sum_{j=0}^{\infty} \tilde{P}(X_j \in \cdot | \tau > j) \tilde{P}(\Lambda = j) \\
&= \tilde{P}(X_\Lambda \in \cdot | \tau > \Lambda),
\end{aligned} \tag{15}$$

where Λ is a rv, independent of X, with a probability mass function given by $\tilde{P}(\Lambda = j) = \tilde{P}(\tau > j)/\tilde{E}\tau$. Hence, π can be exactly simulated, provided that sampling such a Λ can be implemented.

Suppose now that $X = (X_n : n \geq 0)$ is an aperiodic uniformly ergodic Markov chain; every irreducible aperiodic finite state Markov chain is uniformly ergodic. In this case, there exists $m \geq 1$, $\lambda > 0$, and a probability φ such that

$$P(X_m \in \cdot | X_0 = x) \geq \lambda \varphi(\cdot)$$

for all x; see Meyn and Tweedie (2009), p.394. As a consequence, we can write

$$P(X_m \in \cdot | X_0 = x) = \lambda \varphi(\cdot) + (1-\lambda) Q(x, \cdot), \tag{16}$$

where $Q(x,\cdot) \triangleq (P(X_m \in |X_0 = x) - \lambda\varphi(\cdot))/(1-\lambda)$. In view of (16), we can view an m-step transition from x as arising due to a coin flip (with probability of heads equal to λ). If the coin comes up heads, X_m is distributed according to φ, whereas tails implies that one distributes X_m according to $Q(x,\cdot)$. Note that we can view the times at which X distributes itself according to φ as regeneration times for X. It follows that τ is geometrically distributed with parameter λ, and hence Λ is also geometrically distributed with parameter λ. Hence, we arrive at an implementable algorithm for sampling from $\pi(\cdot)$. This is one of the exact simulation algorithms introduced by Asmussen, Glynn, and Thorisson (1992). However, it suffers from the defect that knowledge of λ, φ, and $P(X_m \in \cdot | X_0 = x)$ play a role in the algorithm.

We now describe the coupling-from-the-past (CFTP) algorithm provided by Propp and Wilson (1996) for irreducible aperiodic finite state S-valued Markov chains. Any such Markov chain $X = (X_n : n \geq 0)$ can be viewed as the solution of a stochastic recursion

$$X_{n+1} = r(X_n, Z_{n+1}),$$

where the Z_i's are independent and identically distributed (iid) and r is a deterministic map. Hence,

$$X_{n+1} = \phi_{n+1}(X_n),$$

where $\phi_{n+1}(x) = r(x, Z_{n+1})$. So, conditional on $X_0 = x$,

$$X_n = (\phi_n \circ \phi_{n-1} \circ \cdots \circ \phi_1)(x).$$

But since the ϕ_i's are iid, $(\phi_n \circ \phi_{n-1} \circ \cdots \circ \phi_1) \stackrel{D}{=} (\phi_1 \circ \phi_2 \circ \cdots \circ \phi_n)$ (where $\stackrel{D}{=}$ denotes "equality in distribution"). So, $X_n \stackrel{D}{=} Y_n(x)$, where

$$Y_n(x) \stackrel{D}{=} (\phi_1 \circ \phi_2 \circ \cdots \circ \phi_n)(x).$$

It can be shown that $Y_n(x) \to Y_\infty, a.s.$ as $n \to \infty$; see Asmussen and Glynn (2007), p.122. In particular, if $(Y_m(x) : x \in S)$ has a common value Y_∞ for some m, then $Y_n(x) = Y_m(x)$ for all $n \geq m$. So, if β is the smallest m for which the cardinality of $\{Y_m(x) : x \in S\}$ is one, then $Y_\infty = (\phi_1 \circ \cdots \circ \phi_\beta)(x)$. Furthermore, since $Y_n(x) \stackrel{D}{=} X_n$ for $n \geq 0$, it follows that Y_∞ has the equilibrium distribution π.

One has significant flexibility in one's choice of the mappings $(\phi_n : n \geq 1)$. In particular, note that the ϕ_n's must respect the fact that

$$P(\phi_n(x) = \cdot) = P(X_n = \cdot | X_{n-1} = x)$$

for $x \in S$. Thus, the marginal distributions of the $\phi_n(x)$'s are fixed. But one is then free to choose the joint distribution of the $\phi_n(x)$'s judiciously, subject to this constraint on the marginals. One can generate the $\phi_n(x)$'s independently in x, or one can use common randomness (as in the representation $\phi_n(x) = r(x, Z_n)$) to generate the $\phi_n(x)$'s.

One setting in which this algorithmic freedom can be usefully exploited is when X is a stochastically monotone Markov chain. In this case, r can be chosen so that $r(\cdot, Z_n)$ is monotone. It follows that the

cardinality checking condition is then particularly simple. Specifically, β then corresponds to the first $m \geq 1$ at which $Y_m(x_S)$ agrees with $Y_m(x_L)$, where x_S and x_L are the smallest and largest states in S (assuming that such states exist). In particular, this applies even to continuous state space problems that are stochastically monotone, in which any attempt to check the cardinality condition via independent sampling of the $\phi_n(x)$'s is doomed to fail.

The area of exact simulation has been very active over the last twenty years, and many variates of these algorithms have now been proposed. In addition to the many results that relate to Markov chain Monte Carlo settings, there are also special exact simulation algorithms available for the single-server queue (Ensor and Glynn (2000), Blanchet and Wallwater (2015)), multi-server queues (Blanchet, Dong, and Pei (2015)), multi-dimensional reflecting Brownian motion (Blanchet and Chen (2015)), infinite server queues and related loss systems (Blanchet, Dong, et al. (2015)) and perpetuities (Blanchet, Lam, and Zwart (2012)), as well as many objects that arise from spatial processes.

However, these exact simulation methods are difficult to apply to discrete-event simulations; see Henderson and Tweedie (2000). The notion of exact simulation is also bound up with the notion of ϕ-irreducibility in the Markov chain setting. However, there are continuous state space Markov chains that arise naturally in some applications, in which this form of irreducibility fails to be valid, so that exact simulation is then typically unavailable.

5 EXACT ESTIMATION FOR MARKOV CHAINS

With the relaxation to exact estimation, it is often relatively easy to construct ad hoc schemes that provide unbiased estimators for equilibrium quantities. Consider, for example, the waiting time sequence $W = (W_n : n \geq 0)$ associated with the single-server queue with first-in/first-out(FIFO) queue discipline. Then, the W_n's satisfy the stochastic recursion

$$W_{n+1} = [W_n + Z_{n+1}]^+,$$

where $[x]^+ \triangleq \max(x,0)$ and the Z_n's are iid rv's. In order that the queue be stable, we require that $EZ_1 < 0$. While this Markov chain is stochastically monotone, there is no upper bound to its state space $\mathbb{R}_+$, so that the CFTP methods of Section 4 are not applicable. (Of course, the special purpose algorithms of Ensor and Glynn (2000) do apply here, at least when the Z_i's are light-tailed.)

It is well known that $W_n \Rightarrow W_\infty$ as $n \to \infty$, where $W_\infty \stackrel{D}{=} M_\infty = \max\{S_k : k \geq 0\}$ and $S_k = Z_1 + \cdots + Z_k$; see Asmussen (2003), p.267. So, $P(W_\infty > x) = P(M_\infty > x) = P(T(x) < \infty)$, where $T(x) = \inf\{n \geq 0 : S_n > x\}$. To form an unbiased estimator for $Ef(W_\infty)$, suppose that f is differentiable and non-decreasing, and write

$$\begin{aligned}
Ef(W_\infty) &= E[\int_0^{W_\infty} f'(x)dx + f(0)] \\
&= f(0) + E\int_0^\infty f'(x)I(W_\infty > x)dx \\
&= f(0) + \int_0^\infty f'(x)P(T(x) < \infty)dx \\
&= f(0) + \int_0^\infty \frac{f'(x)}{k(x)}P(T(x) < \infty)k(x)dx \\
&= f(0) + E\frac{f'(R)}{k(R)}P(T(R) < \infty),
\end{aligned}$$

where R is a rv independent of $(S_k : k \geq 0)$ having positive density k.

Of course, $I(T(R) < \infty)$ can not be simulated in finite time, since the entire random walk $(S_k : k \geq 0)$ must be guaranteed in order to ascertain when $T(R) < \infty$ or not. To circumvent this problem, one possibility is to employ importance sampling. In this case, we write

$$P(T(x) < \infty) = E^* L_{T(x)},$$

where $E^*(\cdot)$ is the expectation operator associated with a probability P^* under which $T(x) < \infty$ is a certain event (e.g., $E^* Z_1 > 0$), and $L_{T(x)}$ is the likelihood ratio of P relative to P^* associated with $(S_j : 0 \leq j \leq T(x))$. If the Z_j's are light-tailed with Cramér-Lundberg root $\theta^* > 0$, then the natural choice of P^* is such that

$$L_{T(x)} = \exp(-\theta^* S_{T(x)}).$$

Hence, if $f(x) = x$, one possible exact estimator for $Ef(W_\infty)$ is just

$$\frac{\exp(-\theta^* S_{T(R)})}{k(R)}.$$

Because $S_{T(R)} > R$, it follows that $P(T(x) < \infty) \leq \exp(-\theta^* x)$ for $x > 0$. Hence, if we choose $k(x) = \lambda e^{-\lambda x}$ for some $\lambda > 0$, we see that we should choose $\lambda < \theta^*$ in order to guarantee finite variance for our estimator. In view of the fact that the computational effort required to generate $S_{T(R)}$ under P^* scales in proportion to R, the computer time necessary to generate our exact estimator has finite expectation. Thus, it converges at square root rate in the computational effort c. This idea easily extends to square root convergence rate exact estimation algorithms for queues in which the Z_i's are heavy-tailed, provided that the importance sampling scheme is suitably modified. Hence, we have one exact estimation approach that covers both the light-tailed and heavy-tailed versions of the model. By contrast, exact simulation algorithm design can be quite sensitive to subtle model features related to issues such as the heaviness of the tails of the underlying distributions.

As a second illustration of exact estimation, we consider a "fork-and-join" queue in which each entering job is split into d individual sub-tasks. The sub-tasks are processed in parallel by d servers in FIFO order, and the job is considered complete when the last sub-task finishes. Note that the queue at each parallel server behaves, in isolation, like a single-server queue. Consequently, if A_n is the arrival time of job n and $V_n(i)$ is the processing time of the i'th sub-task for job n, the waiting time at server i for job n (exclusive of service) satisfies the standard single-server waiting time recursion, namely

$$W_n(i) = [W_{n-1}(i) + V_{n-1}(i) - (A_n - A_{n-1})]^+$$

for $n \geq 1$. Thus, the time-in-system (including its processing time) for the n'th job at server i is

$$\mathscr{T}_n(i) = W_n(i) + V_n(i)$$

so that the total time-in-system for job n (i.e., the time that elapses from the job's arrival to the job's completion) is

$$\mathscr{T}_n = \max_{1 \leq i \leq d} [W_n(i) + V_n(i)].$$

We assume that each of the $d+1$ sequences $(\chi_n : n \geq 1)$, $(V_n(1) : n \geq -1)$, $\cdots$, $(V_n(d) : n \geq -1)$ are independent of one another and that each is itself a sequence of iid light-tailed positive rv's. Set $S_n(i) = \sum_{j=1}^n [V_{j-1}(i) - \chi_j]$ and $M_n(i) = \max\{S_j(i) : 0 \leq j \leq n\}$. Note that if the system starts empty,

$$\begin{aligned}
\mathscr{T}_n &= \max_{i \leq i \leq d} [W_n(i) + V_n(i)] \\
&\stackrel{D}{=} \max_{i \leq i \leq d} [W_n(i) + V_{-1}(i)] \\
&\stackrel{D}{=} \max_{i \leq i \leq d} [M_n(i) + V_{-1}(i)] \\
&\nearrow \max_{i \leq i \leq d} [M_\infty(i) + V_{-1}(i)] \stackrel{D}{=} \mathscr{T}_\infty
\end{aligned}$$

We now follow the same approach as for $(W_n : n \geq 0)$ to construct an unbiased estimator for $Ef(\mathscr{T}_\infty)$. Hence, if f is non-decreasing and differentiable, the question comes down to unbiasedly estimating $P(\mathscr{T}_\infty > x)$ in finite time. Note that if $T_i(x) = \inf\{n \geq 0 : S_n(i) > x\}$, then

$$P(\mathscr{T}_\infty > x) = P\left(\min_{1 \leq i \leq d} T_i(x - V_{-1}(i)) < \infty\right).$$

Put $\mathscr{I}(x) = \min\{i \in \{1, \cdots, d\} : T_i(x - V_{-1}(i)) \leq T_k(x - V_{-1}(k)), 1 \leq k \leq d, T_i(x - V_{-1}) < \infty\}$. For $1 \leq i \leq d$, let $\theta_i^* > 0$ be the Cramér-Lundberg root for which $E\exp(\theta_i^*(V_0(i) - \chi_1)) = 1$ (assumed to exist). Then,

$$\begin{aligned} P(\mathscr{T}_\infty > x) &= \sum_{i=1}^{d} P(\mathscr{I}(x) = i) \\ &= \sum_{i=1}^{d} E_i^* \exp(-\theta_i^* S_{T_i(x - V_{-1}(i))}(i)) I(\mathscr{I}(x) = i), \end{aligned}$$

where $E_i^*(\cdot)$ is the expectation operator associated with a distribution P_i^* under which the marginal distributions of the $V_n(i)$'s and χ_n's are modified, respectively, so that $P_i^*(V_n(i) \in dv) = \exp(\theta_i^* v) P(V_n(i) \in dv)$ and $P_i^*(\chi_n \in d\chi) = \exp(-\theta_i^* \chi) P(\chi_n \in d\chi)$. Since $\mathscr{I}(x) = i$ can be determined by simulating each of the d random walks up to $T_i(x - V_{-1}(x))$ (which is finite a.s. under P_i^*), $P(\mathscr{T}_\infty > x)$ can be estimated without bias in finite time, yielding a finite-variance unbiased estimator, provided that we choose $k(x) = \lambda e^{-\lambda x}$ with $\lambda < \min\{\theta_i^* : 1 \leq i \leq d\}$. In contrast to $(W_n : n \geq 0)$, this model is one for which no exact simulation algorithm currently exists.

We turn next to discussion of a more systematic approach for constructing exact estimators in the context of Markov chain equilibrium computations. We return to the idea underlying CFTP algorithms, and represent the Markov chain X_n through the composition

$$X_n = (\phi_n \circ \phi_{n-1} \cdots \circ \phi_1)(x)$$

of random iterated maps. Again, we note that $Y_n(x) = (\phi_1 \circ \phi_2 \circ \cdots \circ \phi_n)(x)$ has the same distribution as X_n. As noted in Section 4, there are many Markov chain settings in which

$$Y_n(x) \to Y_\infty \quad a.s. \tag{17}$$

as $n \to \infty$, where Y_∞ has the equilibrium distribution π. However, rather than using (17) as a starting point for constructing an exact simulation algorithms, we move directly to using Section 3's idea for developing an exact estimator. Specifically, let $\Delta_n = f(Y_n(x)) - f(Y_{n-1}(x))$ and set

$$\tilde{Z} = \sum_{k=0}^{N} \frac{\Delta_k}{P(N \geq k)}.$$

For a broad class of Markov chains, the ϕ_n's have a certain contractive property that guarantees that the distance between $Y_n(x)$ and Y_∞ goes to zero geometrically fast a.s., so that if f is Lipschitz, then Δ_n converges to 0 geometrically fast, as does

$$||f(Y_n(x)) - f(Y_\infty)||_2^2;$$

see Glynn and Rhee (2014). As a consequence, if $P(N \geq n)$ is of order $n^{-\alpha}$ for n large with $\alpha > 0$, $\tilde{Z}$ is guaranteed to have finite variance; see (13). Since computing $(\phi_1 \circ \cdots \circ \phi_j)(x)$ takes order j units of computational effort, it follows that computing $\tilde{Z}$ takes a computational effort of roughly order N^2. (Recall that $\tilde{Z}$ is defined in terms of $(\phi_1 \circ \cdots \circ \phi_j)(x)$ for $1 \leq j \leq N$.) Hence, if we choose $\alpha > 2$, it follows that

$EN^2 < \infty$ and we end up with an exact estimation algorithm that exhibits square root convergence rate in the computational effort c.

So, we see that with a modest amount of algorithmic design effort, we can easily construct exact estimation algorithms. In addition to the relative ease of their design (as contrasted to exact simulation algorithms), these algorithms can have theoretical advantages relative to their exact simulation counterparts. In particular, exact estimation can be feasible to implement even in Markov chain settings where the chain fails to be irreducible in the sense typically required for exact simulation to be valid.

ACKNOWLEDGMENTS

The author gratefully acknowledges the support of the National Science Foundation under Award Number DMS-1320158, as well as Zeyu Zheng for his assistance with LaTeX.

REFERENCES

Asmussen, S. 2003. *Applied Probability and Queues*. Springer, New York.

Asmussen, S., and P. Glynn. 2007. *Stochastic Simulation: Algorithms and Analysis*. Springer, New York.

Asmussen, S., P. Glynn, and J. Pitman. 1995. "Discretization Error in Simulation of One-dimensional Reflecting Brownian Motion". *The Annals of Applied Probability* 5 (4): 875–896.

Asmussen, S., P. W. Glynn, and H. Thorisson. 1992. "Stationarity Detection in the Initial Transient Problem". *ACM Transactions on Modeling and Computer Simulation* 2 (2): 130–157.

Beskos, A., and G. Roberts. 2005. "Exact Simulation of Diffusions". *The Annals of Applied Probability* 15 (4): 2422–2444.

Blanchet, J., and X. Chen. 2015. "Steady-state Simulation of Reflected Brownian Motion and Related Stochastic Networks". *The Annals of Applied Probability* 25 (6): 3209–3250.

Blanchet, J., J. Dong et al. 2015. "Perfect Sampling for Infinite Server and Loss Systems". *Advances in Applied Probability* 47 (3): 761–786.

Blanchet, J., J. Dong, and Y. Pei. 2015. "Perfect Sampling of GI/GI/c Queues". *arXiv preprint arXiv:1508.02262*.

Blanchet, J., H. Lam, and B. Zwart. 2012. "Efficient Rare-event Simulation for Perpetuities". *Stochastic Processes and their Applications* 122 (10): 3361–3392.

Blanchet, J., and A. Wallwater. 2015. "Exact Sampling of Stationary and Time-reversed Queues". *ACM Transactions on Modeling and Computer Simulation* 25 (4): 26.

Blanchet, J. H., and P. W. Glynn. 2015. "Unbiased Monte Carlo for Optimization and Functions of Expectations via Multi-level Randomization". In *Proceedings of the 2015 Winter Simulation Conference*, edited by L. Yilmaz, W. K. V. Chan, I. Moon, T. M. K. Roeder, C. Macal, and M. D. Rossetti, 3656–3667. Piscataway, NJ.

Duffie, D., and P. Glynn. 1995. "Efficient Monte Carlo Simulation of Security Prices". *The Annals of Applied Probability*:897–905.

Ensor, K. B., and P. W. Glynn. 2000. "Simulating the Maximum of a Random Walk". *Journal of Statistical Planning and Inference* 85 (1): 127–135.

Giesecke, K., and D. Smelov. 2013. "Exact Sampling of Jump Diffusions". *Operations Research* 61 (4): 894–907.

Giles, M. B. 2008. "Multilevel Monte Carlo Path Simulation". *Operations Research* 56 (3): 607–617.

Giles, M. B., and L. Szpruch. 2013. "Antithetic Multilevel Monte Carlo Estimation for Multidimensional SDEs". In *Monte Carlo and Quasi-Monte Carlo Methods 2012*, 367–384. Springer.

Glynn, P. W., and P. Heidelberger. 1991. "Analysis of Parallel Replicated Simulations under a Completion Time Constraint". *ACM Transactions on Modeling and Computer Simulation* 1 (1): 3–23.

Glynn, P. W., and C.-H. Rhee. 2014. "Exact Estimation for Markov Chain Equilibrium Expectations". *Journal of Applied Probability* 51:377–389.

Glynn, P. W., and W. Whitt. 1992. "The Asymptotic Efficiency of Simulation Estimators". *Operations Research* 40 (3): 505–520.

Heinrich, S. 2001. "Multilevel Monte Carlo Methods". In *Large-scale Scientific Computing*, 58–67. Springer.

Henderson, S. G., and R. L. Tweedie. 2000. "Perfect Simulation for Markov Chains Arising from Discrete-event Simulation". In *Proceedings of the 38th Annual Allerton Conference on Communication, Control, and Computing*, 1125–1134.

Kloeden, P., and E. Platen. 1992. *Numerical Solutions of Stochastic Differential Equations*. Springer-Verlag.

Lewis, P., and G. Shedler. 1979. "Simulation of Nonhomogeneous Poisson Processes with Degree-two Exponential Polynomial Rate Function". *Operations Research* 27 (5): 1026–1040.

McLeish, D. 2011. "A General Method for Debiasing a Monte Carlo Estimator.". *Monte Carlo Methods and Applications* 17 (4): 301–315.

Meyn, S., and R. L. Tweedie. 2009. *Markov Chains and Stochastic Stability*. Cambridge University Press, Cambridge.

Propp, J. G., and D. B. Wilson. 1996. "Exact Sampling with Coupled Markov Chains and Applications to Statistical Mechanics". *Random Structures and Algorithms* 9 (1-2): 223–252.

Rhee, C.-H., and P. W. Glynn. 2015. "Unbiased Estimation with Square Root Convergence for SDE Models". *Operations Research* 63 (5): 1026–1043.

Steele, J. M. 2000. *Stochastic Calculus and Financial Applications*. Springer.

AUTHOR BIOGRAPHIES

PETER W. GLYNN is the Thomas Ford Professor in the Department of Management Science and Engineering (MS&E) at Stanford University. He is a Fellow of INFORMS and of the Institute of Mathematical Statistics, has been co-winner of Best Publication Awards from the INFORMS Simulation Society in 1993 and 2008, and was the co-winner of the John von Neumann Theory Prize from INFORMS in 2010. In 2012, he was elected to the National Academy of Engineering. His research interests lie in stochastic simulation, queueing theory, and statistical inference for stochastic processes. His email address is glynn@stanford.edu.