Proceedings of the 2003 Winter Simulation Conference
S. Chick, P. J. Sánchez, D. Ferrin, and D. J. Morrice, eds.

EFFICIENT SIMULATION OF GAMMA AND VARIANCE-GAMMA PROCESSES

Athanassios N. Avramidis
Pierre L'Ecuyer
Pierre-Alexandre Tremblay

Département d'Informatique et de Recherche Opérationnelle
Université de Montréal, C.P. 6128, Succ. Centre-Ville
Montréal, H3C 3J7, CANADA

ABSTRACT

We study algorithms for sampling discrete-time paths of a gamma process and a variance gamma process, defined as a Brownian process with random time change obeying a gamma process. The attractive feature of the algorithms is that increments of the processes over longer time scales are assigned to the first sampling coordinates. The algorithms are based on having in explicit form the process' conditional distributions, are similar in spirit to the Brownian bridge sampling algorithms proposed for financial Monte Carlo, and synergize with quasi-Monte Carlo techniques for efficiency improvement. We compare the variance and efficiency of ordinary Monte Carlo and quasi-Monte Carlo for an example of financial option pricing with the variance-gamma model, taken from (Madan, Carr, and Chang 1998).

1 INTRODUCTION

For numerical integration via randomized quasi-Monte Carlo (QMC) techniques, there have been recent publications on the subject of structuring the sampling algorithm so as to concentrate the variance of the integrand to a few coordinates (Caflisch and Moskowitz 1995; Moskowitz and Caflisch 1996; Acworth, Broadie, and Glasserman 1997; Åkesson and Lehoczy 2000; Owen 1998; Liu and Owen 2003). The book of Fox (1999) is centered on such ideas and their synergy with QMC. Caflisch and Moskowitz (1995) and Moskowitz and Caflisch (1996) arose interest by introducing an algorithm that exploits the synergy of such ideas with QMC by sampling discretely paths of a Brownian motion, recursively halfing the sampling horizon, conditional on previously generated values of the process. This method is is known as *Brownian bridge sampling*. Several variants of the structuring approach have been proposed, with Acworth, Broadie, and Glasserman (1997) suggesting an approach based on the principal components of the covariance matrix of a discretely sampled Brownian motion, and Åkesson and Lehoczy (2000) extending the ideas to more general Gaussian processes. Caflisch, Morokoff, and Owen (1997) and Åkesson and Lehoczy (2000) report computational experience with integrals arising in pricing mortgage-backed securities, and Acworth, Broadie, and Glasserman (1997) also report experience with high-dimensional integrals arising in option pricing. The empirical consensus is that the above path generation schemes, when combined with quasi-Monte Carlo, outperform ordinary Monte Carlo (MC) in many situations, sometimes by orders of magnitude. On the other hand, brute-force QMC without the structuring approach has been found to outperform ordinary Monte Carlo less consistently in problems of high dimension.

The above phenomenon can be understood by combining the concepts of *ANOVA decomposition* of a function and *effective dimension* of an integral (Caflisch, Morokoff, and Owen 1997; L'Ecuyer and Lemieux 2000b) with the well-known fact that QMC integration error decreases at a faster rate than ordinary Monte Carlo when the integral's dimension is small. Briefly and loosely speaking, the ANOVA decomposition of a function expresses the variance of a s-dimensional function of random inputs (coordinates) as a sum of variance terms, with a term corresponding to each of the 2^s subsets of coordinates. In many high-dimensional integration problems, and depending on how the coordinates are defined, there exists a subset of coordinates of relatively small cardinality to which most of the variance (e.g., 99%) is due; equivalently, the remaining subset of coordinates, while having large cardinality, contributes little to the variance of the integral. In the case where the first d coordinates account for at least $100p\%$ of the variance, we say that the integral has *effective dimension d in proportion p in the truncation sense* (Caflisch, Morokoff, and Owen 1997; Hickernell 1998b). If p is close to one, this implies that the variance depends essentially only on the uniformity of the d-dimensional QMC point set defined as the projection of the original

QMC point set on its first d coordinates. The smaller d is, the easier it is to make this point set more uniform. Other measures of effective dimension are defined and studied, e.g., in Caflisch, Morokoff, and Owen (1997), L'Ecuyer and Lemieux (2000b), Hickernell (1998a).

In this paper, we begin by introducing the gamma process, a continuous-time process with stationary, independent gamma increments. Madan and Seneta (1990), Madan, Carr, and Chang (1998) introduced in the context of financial option pricing a continuous-time stochastic process termed *variance gamma* that is a Brownian motion with random time change, where the random time change is a gamma process. The authors argued that the variance gamma model permits more flexibility in modelling skewness and kurtosis relative to Brownian motion. They developed closed-form solutions for European option pricing with the VG model and provided empirical evidence that the VG option pricing model gives a better fit to market option prices than the classical Black-Scholes model. Another potential use of the gamma process (more precisely, the analogous process in discrete-time) is as a model of partial sums of positive random variables such as inter-arrival and service times in queueing systems.

We then define algorithms that sample discrete-time paths of the gamma process and the variance gamma process, recursively halfing the sampling horizon, conditional on previously generated values of the process. First, we clarify that exact sampling of gamma-process paths is straightforward, a fact that may be obscured by the discussion in Madan, Carr, and Chang (1998), as we explain in Section 3. Our sampling algorithms are similar in spirit and structure to the Brownian bridge algorithm discussed above; both are based on the premise that many integrals are of low effective dimension, with the macro-effects corresponding to increments of the process over large time scales being dominant in the ANOVA variance decomposition. These algorithms attempt to synergize with quasi-Monte Carlo techniques for efficiency improvement. We compare the variance and efficiency of ordinary Monte Carlo and quasi-Monte Carlo for an example of financial option pricing under the variance gamma model of Madan, Carr, and Chang (1998). We find that our bridge sampling algorithms combined with QMC methods effectively improve simulation efficiency by large factors.

While finalizing this paper, we became aware of related unpublished work by Ribeiro and Webber (2002), who have recently proposed bridge-based sampling algorithms that turn out to be identical to those described in our Figures 2 and 3. The sampling algorithm of Figure 4 seems new. We also experiment with different types of QMC point sets than Ribeiro and Webber (2002) and randomize our QMC point sets in order to obtain unbiased estimators of both the mean and variance (which these authors do not have).

The remainder of the paper is organized as follows. Section 2 reviews Brownian bridge sampling. In Section 3, we introduce the gamma and variance gamma processes, define the sampling algorithms, and discuss applications. In Section 4 we compare (in terms of variance and efficiency) the bridge+QMC algorithms to the QMC-without-bridge and MC algorithms for a simple illustrative example.

2 PREVIOUS RELATED WORK

For completeness and continuity, we review the Brownian bridge sampling in the context of discrete sampling of Brownian paths. Let $\{B(t) : t \geq 0\}$ be a standard Brownian motion with zero drift and unit variance, i.e., such that $B(0) = 0$ and $B(1) \sim \mathcal{N}(0,1)$, where $\sim$ means "is distributed as" and $\mathcal{N}(\mu, \sigma^2)$ denotes the Normal distribution with mean μ and variance σ^2. We wish to estimate via Monte Carlo an integral defined against paths of B for a given discrete-time partition $0 = t_0 < t_1 \leq \ldots \leq t_n = T$ for some given $T > 0$. To make our discussion more concrete, let us assume for example that the integrand in question has effective dimension four in the truncation sense, in proportion p close to one, so that most of the variance is due to the macro-effects represented by $B(T/4)$, $B(T/2)$, $B(3T/4)$, and $B(T)$. This setting, or variants thereof, are quite common in many integration problems arising in financial asset pricing, because $B(T)$ represents (up to a monotone transformation, e.g., the exponential function) the value of an asset or, more generally, a risk factor, and such quantities often capture a large part of the overall uncertainty in the future value of the asset to be priced by the integration algorithm.

The natural sampling algorithm is to sample the Brownian increments along the given partition; but the assumed low effective dimension of the integrand in the truncation sense, with coordinates corresponding to the inputs $B(T/4)$, $B(T/2)$, $B(3T/4)$, and $B(T)$, means that QMC will be very effective if instead we define input coordinates to correspond to the crucial inputs $B(T/4)$, $B(T/2)$, $B(3T/4)$, and $B(T)$, and then sample these inputs via the inverse transform method. This can easily be achieved as follows. We recall the standard property of Brownian motion that for any $t \geq 0$ and nonnegative time increments $\Delta t_1, \Delta t_2$, the conditional distribution of $B(t + \Delta t_1)$ given $B(t)$ and $B(t + \Delta t_1 + \Delta t_2)$ is $\mathcal{N}(aB(t) + (1-a)B(t + \Delta t_1 + \Delta t_2, a\Delta t_2)$, where $a = \Delta t_1/(\Delta t_1 + \Delta t_2)$. Moreover, since $B(\cdot)$ is a Markov process, additionally conditioning on any portion of the path before t and after $t + \Delta t_1 + \Delta t_2$ does

not change the conditional distribution. Based on this property, one samples discretely paths of a Brownian motion, recursively halfing the sampling horizon, conditional on previously generated values of the process, Thus, the Brownian path is sampled in the order $B(T)$, $B(T/2)$, $B(T/4)$, $B(3T/4)$,... Moskowitz and Caflisch (1996) combined this sampling algorithm with using QMC points for the integration. Assuming that all normal variates are generated by transforming a single (pseudo- or quasi-) uniform random number, the integral has low effective dimension and QMC is expected to outperform MC.

3 BRIDGE SAMPLING FOR GAMMA AND VARIANCE GAMMA PROCESSES

We let $\mathcal{G}(\alpha, \beta)$ denote the gamma distribution with density

$$f(x) = \frac{1}{\beta^\alpha \Gamma(\alpha)} x^{\alpha-1} e^{-x/\beta}, \quad x > 0, \tag{1}$$

where Γ is the usual gamma function, and we refer to α and β as the shape and scale parameters, respectively. The gamma process $\{G(t;\mu,\nu) : t \geq 0\}$ with mean parameter μ and variance parameter ν is a continuous-time process with stationary, independent gamma increments such that for any $h > 0$,

$$G(t+h;\mu,\nu) - G(t;\mu,\nu) \sim \mathcal{G}(\mu^2 h/\nu, \nu/\mu). \tag{2}$$

By definition, the distribution of the increments depends on the length h of the time increment but not on the time t. Note that the increment of length h has mean μh and variance νh. The gamma process is the Lévy process corresponding to the gamma distribution. For background on Lévy processes, including their existence, see Sato (1999). Some basic facts about Lévy processes are: (a) they are in one-to-one correspondence with infinitely divisible distributions; (b) the Lévy-Khintchine representation decomposes a Lévy process into a sum of three parts: the first part is a deterministic function of time, the second part is a stochastic process with continuous component, namely a scaled Brownian motion, and the third part is a stochastic process formed by the superposition of compound Poisson processes over a range of possible jump sizes, and where the Poisson jump processes have rate functions that derive from the (possibly-infinite) Lévy measure λ with support on the range of jump sizes. To lighten notation, we refer to the process as $G(t)$ or simply G when the parameters μ and ν are irrelevant or obvious from the context.

Suppose we wish to generate paths of a gamma process between times zero and T for some given $T > 0$. Clearly only a discrete-time skeleton of the process can be generated on a computer, so we assume the goal is to generate values of the process for a discrete-time partition of $(0, T)$, namely at the time values $t_i := i\delta T$, $i = 1, \ldots, 2^k$, where $\delta = 2^{-k}$ for some positive integer k. (This can be generalized easily to arbitrary observation times t_i.) Perhaps the most natural sampling algorithm is to sample the process increments along the above partition, known to be independent, identically distributed gamma variates. We term this natural approach *gamma sequential sampling* (GSS). A pseudocode is given in Figure 1.

$G(0) = 0$;
$h = 2^{-k}T$;
For $i = 1$ to 2^k
  Generate $Q \sim \mathcal{G}(\mu^2 h/\nu, \nu/\mu)$;
  $G(ih) = G((i-1)h) + Q$;
Next i

Figure 1: Gamma Sequential Sampling (GSS) of a Process $G(t;\mu,\nu)$ for a 2^k-Point Equal-Length Partition of $[0, T]$

We observe that Madan, Carr, and Chang (1998) say in their discussion of the gamma process, "the dynamics of the continuous-time gamma process is best explained by describing a simulation of the process," and proceed to describe the standard, general-purpose, but only approximate method for generating paths of a Lévy process with infinite Lévy measure, namely truncation of the Lévy measure near zero (i.e., ignoring jumps below a certain small threshold) and simulation from the appropriate compound Poisson processes. We emphasize that in the case where the process value needs to be observed only at fixed discrete points in time, this approximate and cumbersome approach is unnecessary; it is immediately clear from (2) that the process increments can be simulated exactly via a gamma variate generator.

3.1 Gamma Bridge Sampling

We now describe a sampling algorithm that concentrates the sampling of the macro-effects, i.e., increments of the process over longer time scales, to the first coordinates. A path of the process is sampled at the following time points, in order of generation: T, $T/2$, $T/4$, $3T/4$, $T/8$, $3T/8$, $5T/8$, $7T/8$, ..., δT,..., $(2^k - 1)\delta T$. The algorithm is based on the observation that the conditional distributions of $G(t;\mu,\nu)$ are available in closed form. Specifically, for any $t \geq 0$ and nonnegative time increments $\Delta t_1, \Delta t_2$, the conditional distribution of $G(t + \Delta t_1)$ given $G(t) = \gamma_0$ and

$G(0) = 0$;
Generate $G(T) \sim \mathcal{G}(\mu^2 T/\nu,\ \nu/\mu)$;
For $\ell = 1$ to k
  For $m = 1$ to $2^{\ell-1}$
    $i = 2m - 1$;
    Generate $Y \sim \mathcal{B}(\mu^2 T/(\nu 2^\ell), \mu^2 T/(\nu 2^\ell))$;
    $G(iT/2^\ell) = G((i-1)T/2^\ell)$
      $+ \left(G((i+1)T/2^\ell) - G((i-1)T/2^\ell)\right) Y$;
  Next m
Next ℓ

Figure 2: Gamma Bridge Sampling of a Process $G(t;\mu,\nu)$ for a 2^k-Point Equal-Length Partition of $[0,T]$

$G(t + \Delta t_1 + \Delta t_2) = \gamma_2$ is $\gamma_0 + (\gamma_2 - \gamma_0)Y$, where Y is distributed as $\mathcal{B}(\Delta t_1/\nu, \Delta t_2/\nu)$, and $\mathcal{B}$ denotes the beta distribution on (0,1). Moreover, because of the independent-increments property of $G(\cdot)$, additionally conditioning on any portion of the path before t and/or after $t + \Delta t_1 + \Delta t_2$ does not change the conditional distribution. The conditional distributions do not depend on the mean parameter μ (μ only appears in the unconditional distribution of $G(T)$). As the time scale decreases, the beta parameters decrease, resulting eventually in a bimodal density whose mass concentrates more on the extreme values 0 and 1. We name this conditional sampling of the gamma process *gamma bridge sampling*. A pseudocode for gamma bridge sampling is given in Figure 2.

3.2 The Variance Gamma Process and Bridge Sampling

Madan and Seneta (1990), Madan and Milne (1991), Madan, Carr, and Chang (1998) studied option pricing for a model where the asset log-return dynamics follow a continuous-time process obtained as a subordinate to Brownian motion, using a gamma process to model *randomized operational time* (Feller 1966) (the term "operational time" refers to the time scale of "operations" relevant to the phenomenon of study). We now introduce this model of asset returns, named the *VG process* in Madan, Carr, and Chang (1998).

Let $B = \{B(t;\theta,\sigma) : t \geq 0\}$ be a Brownian motion with drift parameter θ and variance parameter σ. Let $G = \{G(t;1,\nu) : t \geq 0\}$ be a unit-mean gamma process ($\mu = 1$) independent of the process B. The variance gamma (VG) process $X(t;\theta,\sigma,\nu)$ is obtained as a subordinate of the Brownian motion process $B(t;\theta,\sigma)$ using the operational time $G(t;1,\nu)$ (Feller 1966):

$$X(t;\theta,\sigma,\nu) := B(G(t;1,\nu),\theta,\sigma). \tag{3}$$

In simple terms, the VG process is obtained by subjecting the Brownian motion to a random time change obeying a gamma process.

In the interest of brevity, we proceed directly to the specification of the asset price dynamics (under the VG model) relevant to pricing, known as *risk-neutral asset dynamics*. We refer to Madan, Carr, and Chang (1998) for aspects of the VG model that are relevant to the financial literature. Let $\{S(t) : t \geq 0\}$ denote the risk-neutral asset price process. Under the risk-neutral dynamics, the paths of the asset price process obey:

$$S(t) = S(0)\exp\{rt + X(t;\theta_{\rm RN},\sigma_{\rm RN},\nu_{\rm RN}) + \omega_{\rm RN} t\}, \tag{4}$$

where r is the risk-free interest rate, the subscript "RN" indicates that these are the risk-neutral parameters (as opposed to the statistical parameters), and $\omega_{\rm RN} = \log(1 - \theta_{\rm RN}\nu_{\rm RN} - \sigma_{\rm RN}^2\nu_{\rm RN}/2)/\nu_{\rm RN}$. Under the VG model, option prices can be expressed as expectations against paths of the process S over some fixed horizon under the above risk-neutral dynamics.

To the best of our knowledge, except for standard European options, pricing formulas under the VG model are not available in closed form. Hence, numerical or Monte Carlo integration appear to be the only viable approaches; and for the case of high-dimensional integrals arising in the pricing of path-dependent options, Monte Carlo integration is often the leading practical approach.

With such applications in mind, we describe two algorithms for sampling paths of the asset price process (4) that concentrate the sampling of the macro-effects (of X and S) to the first coordinates, where "macro-effects" are understood to be the increments of these processes over longer time scales. In analogy with Brownian bridge sampling and gamma bridge sampling, both algorithms sample the VG process at a time partition that becomes increasingly fine. First, note that to simulate paths of the asset price dynamics, it suffices to generate paths of the VG process $X(t;\theta,\sigma,\nu)$ and then transform these to paths of S via the transformation (4). For the first algorithm, named *Brownian-gamma bridge sampling* (BGBS), we observe that the assumed independence of processes G and B implies that conditional on any collection of increments of the gamma process G, the increments of the Brownian process B are independent normals. Thus, we may first sample increments of G via gamma bridge sampling, and then sample increments of $B(G(t))$ by Brownian bridge sampling, conditional on the corresponding G increments. This can be done in two ways: (a) sampling

$G(0) = 0;\ \ X(0) = 0;$
Generate $G(T) \sim \mathcal{G}(\mu^2 T/\nu,\ \nu/\mu);$
Generate $X(T) \sim \mathcal{N}(\theta G(T), \sigma^2 G(T));$
For $\ell = 1$ to k
  For $m = 1$ to $2^{\ell-1}$
    $i = 2m - 1;$
    Generate $Y \sim \mathcal{B}(\mu^2 T/(\nu 2^\ell),\ \mu^2 T/(\nu 2^\ell));$
    $G(iT/2^\ell) = G((i-1)T/2^\ell)$
      $+ \left[G((i+1)T/2^\ell) - G((i-1)T/2^\ell)\right] Y;$
    $b = G((i+1)T/2^\ell) - G(iT/2^\ell);$
    Generate $Z \sim \mathcal{N}(0, b\sigma^2 Y);$
    $X(iT/2^\ell) = Y \cdot X((i+1)T/2^\ell)$
      $+ (1-Y) \cdot X((i-1)T/2^\ell) + Z;$
  Next m
Next ℓ

Figure 3: Brownian-Gamma Bridge Sampling (BGBS) of a VG process $X(t) = B(G(t;\mu,\nu),\theta,\sigma)$ for a 2^k-Point Equal-Length Partition of $[0,T]$

$\gamma_{\rm p}(0) = 0;\ \ \gamma_{\rm n}(0) = 0;$
Generate $\gamma_{\rm p}(T) \sim \mathcal{G}(\mu_{\rm p}^2 T/\nu_{\rm p},\ \nu_{\rm p}/\mu_{\rm p});$
Generate $\gamma_{\rm n}(T) \sim \mathcal{G}(\mu_{\rm n}^2 T/\nu_{\rm n},\ \nu_{\rm n}/\mu_{\rm n});$
For $\ell = 1$ to k
  For $m = 1$ to $2^{\ell-1}$
    $i = 2m - 1;$
    Generate $Y_{\rm p} \sim \mathcal{B}(\mu_{\rm p}^2 T/\nu_{\rm p} 2^\ell,\ \mu_{\rm p}^2 T/(\nu_{\rm p} 2^\ell));$
    $\gamma_{\rm p}(iT/2^\ell) = \gamma_{\rm p}((i-1)T/2^\ell)$
      $+ \left[\gamma_{\rm p}((i+1)T/2^\ell) - \gamma_{\rm p}((i-1)T/2^\ell)\right] Y_{\rm p};$
    Generate $Y_{\rm n} \sim \mathcal{B}(\mu_{\rm n}^2 T/(\nu_{\rm n} 2^\ell),\ \mu_{\rm n}^2 T/(\nu_{\rm n} 2^\ell));$
    $\gamma_{\rm n}(iT/2^\ell) = \gamma_{\rm n}((i-1)T/2^\ell)$
      $+ \left[\gamma_{\rm n}((i+1)T/2^\ell) - \gamma_{\rm n}((i-1)T/2^\ell)\right] Y_{\rm n};$
    $X(iT/2^\ell) = \gamma_{\rm p}(iT/2^\ell) - \gamma_{\rm n}(iT/2^\ell);$
  Next m
Next ℓ

Figure 4: Double Gamma Bridge Sampling (DGBS) of a VG Process $X(t) = B(G(t;\mu,\nu),\theta,\sigma)$ for a 2^k-Point Equal-Length Partition of $[0,T]$

first all increments of G and then all increments of B; or (b) sampling them in alternance. The pseudocode given in Figure 3 uses the second approach. This method is in fact equivalent to the one sketched in Section 4 of Ribeiro and Webber (2002).

Our second algorithm, named *double-gamma bridge sampling* (DGBS), depends crucially on an alternative representation of the VG process, as the difference between two independent gamma processes (Madan, Carr, and Chang 1998) as follows:

$$X(t;\theta,\sigma,\nu) = \gamma_{\rm p}(t;\mu_{\rm p};\nu_{\rm p}) - \gamma_{\rm n}(t;\mu_{\rm n};\nu_{\rm n}), \qquad (5)$$

where

$$\begin{aligned}
\mu_{\rm p} &= (1/2)\sqrt{\theta^2 + 2\sigma^2/\nu} + \theta/2 \\
\mu_{\rm n} &= (1/2)\sqrt{\theta^2 + 2\sigma^2/\nu} - \theta/2 \\
\nu_{\rm p} &= \left((1/2)\sqrt{\theta^2 + 2\sigma^2/\nu} + \theta/2\right)^2 \nu \\
\nu_{\rm n} &= \left((1/2)\sqrt{\theta^2 + 2\sigma^2/\nu} - \theta/2\right)^2 \nu.
\end{aligned}$$

In view of the above representation of the process X, it is possible to concentrate the sampling of macro-effects of paths of X (and thus also of macro-effects of paths of S) to the first sampling coordinates. Algorithm DGBS samples each of $\gamma_{\rm p}$ and $\gamma_{\rm n}$ by gamma bridge sampling, while interleaving the sampling of the two processes, so that sampling occurs in the following order: $\gamma_{\rm p}(T)$, $\gamma_{\rm n}(T)$, $\gamma_{\rm p}(T/2)$, $\gamma_{\rm n}(T/2)$, $\gamma_{\rm p}(T/4)$, $\gamma_{\rm n}(T/4)$, $\gamma_{\rm p}(3T/4)$, $\gamma_{\rm n}(3T/4)$,... A pseudocode is given in Figure 4.

These sampling algorithms, combined with QMC methods, are experimented and compared with standard MC for a simple numerical example in Section 4.2.

3.3 Queueing Simulation With Gamma Increments

Recall our discussion motivating the effectiveness of Brownian bridge sampling in the context of financial simulations. In the context of queueing, one could argue that effective structuring of queueing simulations may bring efficiency improvement by reducing the effective dimension in a similar way. One line of thought would be that large-scale effects, measured by increments of the partial-sum processes of interarrival and service times over large—as opposed to small—blocks of successive customers, may contain a large fraction of the relevant uncertainty.

For a simple concrete illustration, consider a single-server queue with i.i.d. interarrival times $A = \{A_i : i = 1,2,\ldots\}$ and i.i.d. customer service times $S = \{S_i : i \geq 1,2,\ldots\}$, where the sequences A and S are independent. Assume that A_i and S_i are gamma random variables with means $\mu_{\rm A}$ and $\mu_{\rm S}$ and variances $\nu_{\rm A}$ and $\nu_{\rm S}$, respectively.

Many performance measures can be estimated by

simulating the system's evolution over a fixed number d of customers. For convenience of exposition, we take $d = 2^k$ for some positive integer k. Consider the *partial-sum processes* $\{\sum_{i=1}^{j} A_i : j = 1, 2, \ldots, d\}$ and $\{\sum_{i=1}^{j} S_i : j = 1, 2, \ldots, d\}$. Observe that the joint distributions of the interarrival time partial sums $\{\sum_{i=1}^{j} A_i : j = 1, 2, \ldots, d\}$ coincide with the finite dimensional distributions of a continuous-time gamma process sampled at integer values, $\{G(j; \mu_A, \nu_A), j = 1, 2, \ldots, d\}$, and analogously for the service-time partial sums. Thus the gamma bridge sampling algorithm immediately applies for sampling the partial sums, with the key feature that macro-effects are concentrated on the first sampling coordinates.

In preliminary experiments comparing the efficiency of simulations for estimating the average sojourn time of the first d customers in a single queue using standard MC, QMC with gamma sequential sampling as in Figure 1, and QMC with gamma bridge sampling as in Figure 2, we found that QMC generally helps (this agrees with the results of Lemieux and L'Ecuyer 2000) but that using the gamma bridge is not always better than gamma sequential sampling. The explanation is that the interaction between customers that are close to each other in time can be much more important, in this example, than the interaction between customers that are far from each other (Lemieux and L'Ecuyer 2000). On the other hand, one can certainly construct queueing examples where using the gamma bridge would bring significant benefit. This issue is in need for more study.

4 NUMERICAL RESULTS

4.1 Experimental Setup

This section reports numerical results on an option pricing example. The simulations were made using the SSJ software package (L'Ecuyer, Meliani, and Vaucher 2002). For the QMC point sets, we used (a) the Korobov lattice rules proposed in Table 1 of L'Ecuyer and Lemieux (2000b) (we used the third value of a in the table for each value of n) and (b) Sobol' point sets with the same parameters as in Bratley and Fox (1988).

The lattice rules were randomized by random shifts modulo 1, as explained in L'Ecuyer and Lemieux (2000b), whereas the digital nets of Sobol' were randomized by the affine matrix scrambling in Definition 2.8 of Owen (2003). For each QMC point set, we made m independent replicates of the randomization, for a total of mn simulation runs, where n is the cardinality of the point set.

For $j = 1, \ldots, m$, let X_j be the average value of the performance measure of interest over the n runs of the jth randomization. These X_j are i.i.d. unbiased estimators of the integral of interest, say μ, so we can use their sample mean $\bar{X}$ and sample variance S^2_{qmc} to compute a confidence interval for μ in a standard way under the assumption that the X_j are also approximately normally distributed. For comparison, we make mn (independent) simulation runs with the MC method and compute the sample variance S^2_{mc}. A confidence interval on the *variance reduction factor* $\sigma^2_{\text{qmc}}/\sigma^2_{\text{mc}}$ (the variance of QMC over that of MC) can be computed by using the well-known fact that under the normality assumption,

$$\frac{(m-1)mnS^2_{\text{qmc}}\sigma^2_{\text{mc}}}{m(mn-1)S^2_{\text{mc}}\sigma^2_{\text{qmc}}}$$

has the F distribution with (m, mn) degrees of freedom.

We define the *efficiency improvement factor* as $c_{\text{qmc}}\sigma^2_{\text{qmc}}/[c_{\text{mc}}\sigma^2_{\text{mc}}]$, where c_{qmc} and c_{mc} are the average cost (CPU time) per simulation run for QMC and MC, respectively. A confidence interval on this factor can also be computed via the F distribution if we neglect noise in the estimation of c_{qmc} and c_{mc}. This is what we shall do.

4.2 Asian Option Pricing With the VG Model

We consider pricing a simple example of *asian call option* under the variance gamma model for the asset price dynamics. The payoff depends on the arithmetic mean of the underlying asset value at prespecified points in time, as follows (Hull 2000): The option holder receives at expiration time T the *payoff*

$$C_A = \max\left(0, \frac{1}{d}\sum_{j=1}^{d} S(t_j) - K\right) \tag{6}$$

where $S(t)$ is the asset price at time t, d is the number of instants where $S(\cdot)$ is sampled, t_j is the jth sampling time (here we assume that $t_j = jT/d$ for all j), and K is a constant called the *strike price* of the option. The option value, is $\mu = e^{-rT}\mathrm{E}[C_A]$, where the expectation is taken with respect to the risk-neutral asset price process defined in (4). The goal is to estimate this μ as efficiently as possible, i.e., with the smallest variance for a given computing budget, by simulation. This is an integration problem in $s = 2d$ dimensions.

We have implemented several QMC-based sampling methods for the random variable C_A. BGBS(b) and DGBS are the methods described in Figures 3 and 4. BGBS(a) is the variant of BGBS where the d values of G are sampled first (using gamma-bridge sampling) and then the d values of B are sampled via the Brownian bridge method. BGSS(a), BGSS(b), and DGSS refer to versions of these three methods where bridge sampling

is replaced by sequential sampling for both the gamma and Brownian processes. Finally, MC means the standard MC method with sequential sampling for both processes. All normal, gamma, and beta random variables are generated by inversion, using numerical approximations of the inverse distribution functions. We used the methods given in Kennedy and Gentle (1980) for the normal distribution, DiDonato and Moris (1987) for the gamma distribution, and the "inverse of the incomplete beta integral" function from the Cephes Math Library (Moshier 2000) for the beta distribution. With these approximations and our implementations, the inverse is roughly fifty times more costly to compute for the beta than for the gamma, and four times more costly for the gamma than for the normal. This has to be considered when comparing estimator's efficiencies. We estimate the variance reduction and efficiency improvement factors of each QMC method compared with MC. In order to be able to use the F distribution for computing confidence intervals on the variance reduction factors, all the methods were simulated independently of each other. (A different approach would have taken common random numbers across the methods and might have provided less noisy estimators of the variance reduction factors.)

For the numerical illustration, we take parameter values from Madan, Carr, and Chang (1998): Let $\theta_{\mathrm{RN}} = -0.1436$, $\sigma_{\mathrm{RN}} = 0.12136$, $\nu_{\mathrm{RN}} = 0.3$, $r = 0.1$, $T = 1$, $K = 101$, and $S(0) = 100$. We made $m = 100$ independent randomizations for each QMC method. Table 1 provides estimates of the improvement factors of each of the six QMC sampling method compared with MC, for three values of n, using the Korobov lattice rules and the Sobol' point sets. Each table entry gives a 98% confidence interval on the variance reduction factor (top pair) and a 98% confidence interval on the efficiency improvement factor (bottom pair).

We see a clear evidence of variance reduction and efficiency improvement for QMC compared with MC, and these improvements are more pronounced when QMC is combined with bridge sampling. The Sobol' point sets perform better, with the bridge methods, than Korobov rules. The DGBS method generally provides the best variance reduction, but the efficiency improvement is often better with BGBS, after the higher cost of DGBS (for generating the beta random variables) has been factored out. The eventual availability of good approximations that are fast to compute for the inverse beta distribution could definitely make DGBS the method of choice.

One can also observe that for Sobol' point sets, BGBS(b) is much better than BGBS(a), whereas for Korobov rules these is no clear winner between these two sampling methods. The explanation is that in contrast to the Korobov rules, the Sobol' point sets have been designed to have better uniformity over their first coordinates than over their last ones, and the BGBS(b) sampling method exploits this by transfering more of the variance to the first few random variables (i.e., it achieves a lower effective dimension in the truncation sense).

Table 1: 98% Confidence Intervals on Variance Reduction Factors (Above) and Efficiency Improvement Factors (Below) of Randomized QMC Over MC, for the Asian Option Example with $d = 16$

Korobov Lattice Rules			
	$n = 4093$ $a = 1397$	$n = 8191$ $a = 7151$	$n = 16381$ $a = 5693$
BGSS(a)	(5, 10) (5, 10)	(6, 12) (6, 12)	(10, 19) (10, 19)
BGSS(b)	(5, 10) (5, 10)	(7, 15) (7, 15)	(10, 21) (10, 21)
DGSS	(15, 30) (7, 15)	(7, 15) (4, 7)	(27, 54) (14, 28)
BGBS(a)	(32, 65) (18, 36)	(34, 68) (19, 38)	(65, 130) (36, 72)
BGBS(b)	(16, 32) (9, 18)	(45, 90) (25, 50)	(50, 100) (27, 53)
DGBS	(38, 76) (11, 21)	(73, 145) (20, 41)	(87, 174) (24, 49)
Sobol' Nets			
	$n = 4096$	$n = 8192$	$n = 16384$
BGSS(a)	(11, 22) (11, 22)	(7, 15) (7, 15)	(12, 24) (12, 24)
BGSS(b)	(13, 26) (13, 26)	(8, 16) (8, 16)	(19, 38) (19, 38)
DGSS	(27, 54) (11, 22)	(31, 62) (16, 31)	(40, 81) (20, 41)
BGBS(a)	(85, 170) (42, 85)	(56, 111) (28, 56)	(95, 190) (47, 95)
BGBS(b)	(149, 298) (84, 168)	(175, 351) (99, 198)	(285, 570) (76, 152)
DGBS	(234, 468) (65, 131)	(359, 718) (101, 201)	(321, 642) (90, 180)

It should be pointed out that these variance reduction factors also depend a lot on the choice of parameters for the QMC point sets (e.g., the value of a for the Korobov rules and the "direction numbers" for the Sobol' point sets; see L'Ecuyer and Lemieux 2002 or L'Ecuyer 2003 for an explanation of these parameters). Changing the values of a in Table 1, for example, may improve or degrade some factors significantly. In general, larger values of n should give larger reduction

factors for QMC, but this is not totally clear from the table because (1) of the noise in the variance estimators and (2) for a given value of n and given type of point set, the specific parameters selected here might be significantly better or worse than other parameters for the same n, for the particular integral considered here.

In other experiments (whose detailed results are not given here), we found that changing the number d of observation times while keeping all other parameters constant seems to have no significant effect on the variance reduction factor. On the other hand, increasing the expiration date T while keeping all other parameters constant tends to increase the variance reduction (and efficiency improvement) factors with the QMC methods.

ACKNOWLEDGMENTS

This work has been supported by NSERC-Canada grant No. ODGP0110050, NATEQ-Québec grant No. 02ER3218, and a Killam Research Fellowship to the second author.

REFERENCES

Acworth, P., M. Broadie, and P. Glasserman. 1997. A comparison of some Monte Carlo and quasi-Monte Carlo techniques for option pricing. In *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing*, ed. P. Hellekalek and H. Niederreiter, Number 127 in Lecture Notes in Statistics, 1–18. Springer-Verlag.

Åkesson, F., and J. P. Lehoczy. 2000. Path generation for quasi-Monte Carlo simulation of mortgage-backed securities. *Management Science* 46:1171–1187.

Bratley, P., and B. L. Fox. 1988. Algorithm 659: Implementing Sobol's quasirandom sequence generator. *ACM Transactions on Mathematical Software* 14 (1): 88–100.

Caflisch, R. E., W. Morokoff, and A. Owen. 1997. Valuation of mortgage-backed securities using Brownian bridges to reduce effective dimension. *The Journal of Computational Finance* 1 (1): 27–46.

Caflisch, R. E., and B. Moskowitz. 1995. Modified Monte Carlo methods using quasi-random sequences. In *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing*, ed. H. Niederreiter and P. J.-S. Shiue, Number 106 in Lecture Notes in Statistics, 1–16. New York: Springer-Verlag.

DiDonato, A. R., and A. H. Moris. 1987. Incomplete gamma function ratios and their inverse. *ACM Transactions on Mathematical Software* 13:318–319.

Feller, W. 1966. *An introduction to probability theory and its applications, vol. 2*. first ed. New York: Wiley.

Fox, B. L. 1999. *Strategies for quasi-Monte Carlo*. Boston, MA: Kluwer Academic.

Hickernell, F. J. 1998a. A generalized discrepancy and quadrature error bound. *Mathematics of Computation* 67:299–322.

Hickernell, F. J. 1998b. Lattice rules: How well do they measure up? In *Random and Quasi-Random Point Sets*, ed. P. Hellekalek and G. Larcher, Volume 138 of *Lecture Notes in Statistics*, 109–166. New York: Springer.

Hull, J. 2000. *Options, futures, and other derivative securities*. fourth ed. Englewood-Cliff, N.J.: Prentice-Hall.

Kennedy Jr., W. J., and J. E. Gentle. 1980. *Statistical computing*. New York: Dekker.

L'Ecuyer, P. 2003. Quasi-Monte Carlo methods for simulation. In *Proceedings of the 2003 Winter Simulation Conference*. Piscataway, NJ: IEEE Press.

L'Ecuyer, P., and C. Lemieux. 2000b. Variance reduction via lattice rules. *Management Science* 46 (9): 1214–1235.

L'Ecuyer, P., and C. Lemieux. 2002. Recent advances in randomized quasi-Monte Carlo methods. In *Modeling Uncertainty: An Examination of Stochastic Theory, Methods, and Applications*, ed. M. Dror, P. L'Ecuyer, and F. Szidarovszki, 419–474. Boston: Kluwer Academic Publishers.

L'Ecuyer, P., L. Meliani, and J. Vaucher. 2002. SSJ: A framework for stochastic simulation in Java. In *Proceedings of the 2002 Winter Simulation Conference*, ed. E. Yücesan, C.-H. Chen, J. L. Snowdon, and J. M. Charnes, 234–242: IEEE Press.

Lemieux, C., and P. L'Ecuyer. 2000. A comparison of Monte Carlo, lattice rules and other low-discrepancy point sets. In *Monte Carlo and Quasi-Monte Carlo Methods 1998*, ed. H. Niederreiter and J. Spanier, 326–340. Berlin: Springer.

Liu, R., and A. B. Owen. 2003. Estimating mean dimensionality. Manuscript.

Madan, D. B., P. P. Carr, and E. C. Chang. 1998. The variance gamma process and option pricing. *European Finance Review* 2:79–105.

Madan, D. B., and F. Milne. 1991. Option pricing with V.G. martingale components. *Mathematical Finance* 1:39–55.

Madan, D. B., and E. Seneta. 1990. The variance gamma (V.G.) model for share market returns. *Journal of Business* 63:511–524.

Moshier, S. L. 2000. Cephes math library. See `http:`

//www.moshier.net.

Moskowitz, B., and R. E. Caflisch. 1996. Smoothness and dimension reduction in quasi-Monte Carlo methods. *Journal of Mathematical and Computer Modeling* 23:37–54.

Owen, A. B. 1998. Latin supercube sampling for very high-dimensional simulations. *ACM Transactions of Modeling and Computer Simulation* 8 (1): 71–102.

Owen, A. B. 2003. Variance and discrepancy with alternative scramblings. *ACM Transactions of Modeling and Computer Simulation* 13 (4). To appear.

Ribeiro, C., and N. Webber. 2002, December. Valuing path-dependent options in the variance-gamma model by Monte Carlo with a gamma bridge. Manuscript.

Sato, K.-I. 1999. *Lévy processes and infinitely divisible distributions*. Cambridge, UK: Cambridge University Press.

AUTHOR BIOGRAPHIES

ATHANASSIOS (THANOS) N. AVRAMIDIS is an Invited Researcher at the "Département d' Informatique et de Recherche Opérationnelle" at the Université de Montréal, Canada. He has been on the faculty at Cornell University and a consultant with SABRE Decision Technologies. His primary research interests are Monte Carlo simulation, particularly efficiency improvement, the interface to probability and statistics, and applications in computational finance, call center operations, and transportation. His web page is <http://www.iro.umontreal.ca/~avramidi>.

PIERRE L'ECUYER is Professor in the Département d'Informatique et de Recherche Opérationnelle, at the Université de Montréal, Canada. His main research interests are random number generation, quasi-Monte Carlo methods, efficiency improvement via variance reduction, sensitivity analysis and optimization of discrete-event stochastic systems, and discrete-event simulation in general. He obtained the prestigious *E. W. R. Steacie* fellowship in 1995-97 and a *Killam* fellowship in 2001-03. His recent research articles are available on-line at http://www.iro.umontreal.ca/~lecuyer.

PIERRE-ALEXANDRE TREMBLAY is a M.Sc. student in the Computational and Mathematical Finance program at the Université de Montréal, Canada. He currently works on the application of simulation methods to mathematical finance and the development of simulation software in Java.