

SIMULATION OF A LÉVY PROCESS BY PCA SAMPLING TO REDUCE THE EFFECTIVE DIMENSION

Pierre L'Ecuyer
Jean-Sébastien Parent-Chartier
Maxime Dion

DIRO, Université de Montréal
C.P. 6128, Succ. Centre-Ville
Montréal (Québec), H3C 3J7, CANADA

ABSTRACT

We consider a Lévy process monitored at s (fixed) observation times. The goal is to estimate the expected value of some function of these s observations by (randomized) quasi-Monte Carlo. For the case where the process is a Brownian motion, clever techniques such as Brownian bridge sampling and PCA sampling have been proposed to reduce the effective dimension of the problem. The PCA method uses an eigen-decomposition of the covariance matrix of the vector of observations so that a larger fraction of the variance depends on the first few (quasi)random numbers that are generated. We show how this method can be applied to other Lévy processes, and we examine its effectiveness in improving the quasi-Monte Carlo efficiency on some examples. The basic idea is to simulate a Brownian motion at s observation points using PCA, transform its increments into independent uniforms over $(0,1)$, then transform these uniforms again by applying the inverse distribution function of the increments of the Lévy process. This PCA sampling technique is quite effective in improving the quasi-Monte Carlo performance when the sampled increments of the Lévy process have a distribution that is not too far from normal, which typically happens when the process is observed at a large time scale, but may turn out to be ineffective in cases where the increments are far from normal.

1 INTRODUCTION

We are interested in the problem of estimating the mean μ of a random variable X defined as a function of a stochastic process monitored at a finite number of observation times. This problem is encountered in many situations, notably in computational finance (Glasserman 2004). The method we will discuss is also applicable to closely related problems such as estimating a quantile of X , or optimizing its mean

with respect to some parameters of the stochastic process (Asmussen and Glynn 2007, Henderson and Nelson 2006).

The standard Monte Carlo (MC) method estimates μ by simulating n independent realizations of X and taking the average. Randomized quasi-Monte Carlo (RQMC) tries to obtain a more accurate estimator by inducing a negative dependence between the n copies of X (Owen 1998, L'Ecuyer and Lemieux 2002, L'Ecuyer 2008a). One of the important ingredients for the effectiveness of this technique is a low effective dimension of X viewed as a function of the uniform random numbers that drive the simulation. That is, the realization of X should be determined mainly by the first few random numbers of the simulation, or (at least) one should be able to closely approximate X by a sum of functions that depend only on a few random numbers (i.e., a sum of low-dimensional functions). A given random variable X can indeed be defined in many different ways as a function of the underlying uniforms. While the choice of definition has no impact on the mean and MC variance, it can have a large impact on the RQMC variance (and effectiveness). We will give examples of this later on.

In this paper, we focus on the case where the stochastic process that determines X is a Lévy process, i.e., a stationary process with independent increments. To simulate the Lévy process at the specified monitoring times, we simulate a Brownian motion at a set of monitoring times of the same cardinality (either the same times or different ones), transform the increments of this Brownian motion into independent $U(0,1)$ random variates (i.e., uniform over the interval $(0,1)$) by applying the normal distribution function, then transform back these uniforms to the increments of the Lévy process by applying the inverse distribution function of these increments. The motivation for this sampling scheme is that it permits one to apply well-known effective dimension reduction techniques to the Brownian motion, hoping that the Lévy process will inherit (at least some of)

this dimension reduction. The vector of observations of the Brownian motion has a multivariate normal distribution. It can be generated in particular via an eigen-decomposition of its covariance matrix, also called PCA sampling; this concentrates most of its variance on the first few generated uniforms and reduces the effective dimension of most well-behaved functions of these observations. The aim of this paper is to explore the effectiveness of this PCA sampling technique for Lévy processes.

The main idea exploited here could in fact apply more generally. For any simulation whose output is a random variate X function of s independent random variates Y_1, \dots, Y_s with known distribution functions, one can generate an arbitrary Brownian motion at s observation times using some effective dimension reduction technique, transform its increments into s independent uniforms, then apply the inverse distribution function of Y_j to the j th uniform to generate Y_j , for each j . It remains to be seen in which situations this technique is really effective, and how to optimize (or select) the s observation times of the Brownian motion. More generally, one could try to optimize the decomposition of the covariance matrix of the Brownian motion in order to minimize the RQMC variance for the given random variable X of interest. We will not pursue these generalizations in the present paper.

The remainder of the paper is organized as follows. In Section 2, we briefly recall the RQMC methodology and the motivation for a low effective dimension. In Section 3, we describe Lévy processes and how they can be simulated by PCA.

2 RANDOMIZED QUASI-MONTE CARLO AND EFFECTIVE DIMENSION

Suppose we want to estimate an integral of the form

$$\mu = \mathbb{E}[f(\mathbf{U})] = \int_{[0,1]^s} f(\mathbf{u}) d\mathbf{u} \quad (1)$$

where \mathbf{U} has the uniform distribution over $[0,1]^s$. RQMC estimates μ by

$$\hat{\mu}_{n,\text{rqmc}} = \frac{1}{n} \sum_{i=0}^{n-1} f(\mathbf{U}_i),$$

where $P_n = \{\mathbf{U}_0, \dots, \mathbf{U}_{n-1}\}$ is a set of n points in $[0,1]^s$ with the following properties: (a) each \mathbf{U}_i is a random vector with the uniform distribution over $[0,1]^s$, and (b) with probability one, the point set $\mathbf{U}_0, \dots, \mathbf{U}_{n-1}$ covers the unit cube $[0,1]^s$ very evenly, in some sense; see Niederreiter (1992), L'Ecuyer and Lemieux (2002), Glasserman (2004), L'Ecuyer (2008a) for how to construct such point sets. From another viewpoint, we can say that there is a negative dependence between the points \mathbf{U}_i .

When s is large, covering the unit cube $[0,1]^s$ very uniformly seems to require a number of points that increases at least exponentially with s (e.g., because the number of corners to cover increases exponentially). On the other hand, the function f can sometimes be well-approximated by a sum of low-dimensional functions, in which case it suffices that these low-dimensional functions are integrated with high accuracy. This requires high uniformity of the RQMC point set only for its projections on the subspaces in which these low-dimensional functions are defined. High uniformity of these projections is much easier to achieve than high uniformity in the entire (large-dimensional) unit cube, especially if these important projections are those that corresponds to the first few coordinates of the points \mathbf{U}_i . Changes of variables can be applied to transform the integrand (i.e., the definition of X as a function of \mathbf{u}) so that more of the variance depends on the first few uniforms (Glasserman 2004, Imai and Tan 2006, L'Ecuyer 2008a).

To be more explicit, whenever $X = f(\mathbf{U})$ has finite variance σ^2 , where \mathbf{U} is uniformly distributed over $(0,1)^s$, the function f can be decomposed uniquely as

$$f(\mathbf{u}) = \mu + \sum_{\mathbf{u} \subseteq \mathcal{S}, \mathbf{u} \neq \emptyset} f_{\mathbf{u}}(\mathbf{u}) \quad (2)$$

where $\mathcal{S} = \{1, \dots, s\}$, each $f_{\mathbf{u}} : [0,1]^s \rightarrow \mathbb{R}$ depends only on $\{u_i, i \in \mathbf{u}\}$, the $f_{\mathbf{u}}$'s integrate to zero and are orthogonal, and the variance decomposes as $\sigma^2 = \sum_{\mathbf{u} \subseteq \mathcal{S}} \sigma_{\mathbf{u}}^2$ where $\sigma_{\mathbf{u}}^2 = \text{Var}[f_{\mathbf{u}}(\mathbf{U})]$ (with $f_{\emptyset} = \mu$). If

$$\sum_{\mathbf{u} \in \mathcal{J}} \sigma_{\mathbf{u}}^2 \geq \rho \sigma^2 \quad (3)$$

for a class \mathcal{J} of small subsets of \mathcal{S} and some ρ close to 1, and if we can construct the RQMC point set so that the projection $P_n(\mathbf{u})$ of P_n over the subspace determined by \mathbf{u} is highly uniform for all $\mathbf{u} \in \mathcal{J}$, then the RQMC variance can be much smaller than the MC variance for this function f . In particular, if (3) holds for $\mathcal{J} = \{\mathbf{u} : |\mathbf{u}| \leq d\}$ for some small d , we say that f has *low effective dimension in the superposition sense* (Owen 1998). If it holds for $\mathcal{J} = \{\mathbf{u} \subseteq \{1, \dots, d\}\}$ for some small d , we say that f has low effective dimension in the *truncation sense* (Caflish, Morokoff, and Owen 1997). The effective dimension in the truncation sense can often be reduced by redefining f without changing the expectation μ , via a change of variables, in a way that the first few uniforms account for most of the variance in f (Acworth, Broadie, and Glasserman 1998, Avramidis and L'Ecuyer 2006, Caflish, Morokoff, and Owen 1997, Glasserman 2004, Imai and Tan 2006, L'Ecuyer 2004, Moskowitz and Caflish 1996, Morokoff 1998, Wang and Sloan 2007). In other words, we change the way the uniforms are used to generate the estimator X in the simulation.

3 SIMULATING LÉVY PROCESSES

A Lévy process $\{Y(t), t \geq 0\}$ is a continuous-time stochastic process with stationary and independent increments, and with $Y(0) = 0$ (Bertoin 1996, Asmussen and Glynn 2007). That is, for arbitrary $t_s > t_{s-1} > \dots > t_1 > t_0 = 0$, the increments $Y(t_j) - Y(t_{j-1})$, $j = 1, \dots, s$, are independent random variables and the distribution of $Y(t_j) - Y(t_{j-1})$ depends only on the length $t_j - t_{j-1}$. Lévy processes are *infinitely divisible*, which means that for any fixed $t > 0$, $Y(t) - Y(0)$ can be written as a sum of n i.i.d. random variables for any positive integer n (arbitrarily large). Conversely, every process having this property is a Lévy process. The stationary Poisson process, the Brownian motion, the inverse Gaussian process, and the gamma process are prominent examples of Lévy processes.

A natural way to generate the trajectory of a Lévy process at the discrete monitoring times $0 = t_0 < t_1 < \dots < t_s$ is by generating the independent increments $Y(t_j) - Y(t_{j-1})$ successively, for $j = 1, \dots, s$. This is the *sequential sampling* or *random walk* approach (Glasserman 2004). For a stationary Brownian motion, for example, these increments are independent normal random variables whose mean and variance are given explicitly by the parameters of the process and are proportional to $t_j - t_{j-1}$. They are easy to generate. Generating random variates from the distribution of the increments is not easy for all Lévy processes, but it is often at least feasible. In this paper, we consider only Lévy processes whose increments can be generated by inversion.

Suppose that we want to estimate an integral of the form $\mu = \mathbb{E}[g(\mathbf{Y})]$ for some function $g : \mathbb{R}^s \rightarrow \mathbb{R}$, where $\mathbf{Y} = (Y(t_1), \dots, Y(t_s))$, by randomized quasi-Monte Carlo (RQMC). We assume that the increments $Y(t_j) - Y(t_{j-1})$ are generated by inversion from independent $U(0,1)$ random variables U_1, \dots, U_s . One can also write $\mu = \mathbb{E}[f(U_1, \dots, U_s)] = \int_{[0,1]^s} f(\mathbf{u}) d\mathbf{u}$ for some function f that incorporates all the transformations from the U_j to $g(\mathbf{Y})$. Then we can apply RQMC as outlined earlier. Here, the dimension is s and the effective dimensions in the truncation sense is likely to be large when s is large, because all increments play a non-negligible role in determining the sample path.

For certain types of Lévy processes, we also know how to generate random variates from the distribution of $Y(t)$ conditional on $\{Y(t_1) = y_1, Y(t_2) = y_2\}$ for arbitrary values of y_1, y_2 , and $t_1 < t < t_2$. Then, a second approach to generate $Y(t_1), \dots, Y(t_s)$ is by the following *Lévy bridge sampling* approach. To keep the notation simple, we assume here that s is a power of 2. We first generate the final value $Y(t_s)$, then we generate $Y(t_{s/2})$ from its conditional distribution given $(Y(t_0), Y(t_s))$, and we apply the same technique recursively to generate $Y(t_{s/4})$ conditional on $(Y(t_0), Y(t_{s/2}))$, then $Y(t_{3s/4})$ conditional on $(Y(t_{s/2}), Y(t_s))$, then $Y(t_{s/8})$ conditional on $(Y(t_0), Y(t_{s/4}))$, and so on, until

all s values have been determined. This technique is convenient to approximate the trajectory of Y up to a certain accuracy, and where the required value of s is not necessarily known in advance. It also provides a powerful tool to improve the effectiveness of quasi-Monte Carlo (QMC) methods by reducing the effective dimension of the problem. The idea is that the first few random numbers that are generated have larger impact on the trajectory under this technique than with sequential sampling. This method was proposed in combination with QMC by Moskowitz and Caflisch (1996) for the case of a Brownian motion; it is then called Brownian bridge sampling. It was further studied by Caflisch, Morokoff, and Owen (1997), Glasserman (2004), and Avramidis and L'Ecuyer (2006), among others.

In the case of a Brownian motion, a much more general method to sample the vector $\mathbf{Y} = (Y(t_1), \dots, Y(t_s))$ is as follows (Devroye 1986, Glasserman 2004). Let Φ be the standard normal distribution function. Decompose the covariance matrix Σ of \mathbf{Y} as $\Sigma = \mathbf{A}\mathbf{A}^t$ for some matrix \mathbf{A} (where t means “transposed”), generate $\mathbf{Z} = (Z_1, \dots, Z_s)^t$ where the $Z_j = \Phi^{-1}(U_j)$ are independent standard normal random variables, and return $\mathbf{Y} = \mathbf{A}\mathbf{Z}$. The Z_j 's are easily generated by inversion (Devroye 1986, L'Ecuyer 2008b). The decomposition $\Sigma = \mathbf{A}\mathbf{A}^t$ is not unique; there are in fact (in general) an infinite number of matrices \mathbf{A} that satisfy this condition. A first possibility, the *Cholesky factorization*, takes \mathbf{A} to be lower triangular and is equivalent to sequential sampling. Brownian bridge sampling corresponds to a second way of decomposing Σ . A third possibility takes $\mathbf{A} = \mathbf{P}\mathbf{D}^{1/2}$ where \mathbf{D} is a diagonal matrix that contains the eigenvalues of Σ in decreasing order and \mathbf{P} is an orthogonal matrix whose columns are the corresponding unit-length eigenvectors. This is the classical eigen-decomposition used in standard *principal component analysis* (PCA). It selects \mathbf{A} so that the maximum amount of variance of \mathbf{Y} comes from Z_1 , then the maximum amount of variance conditional on Z_1 comes from Z_2 , and so on. In other words, this *PCA sampling scheme* concentrates the variance in the first coordinates of \mathbf{Z} as much as possible, i.e., in the first uniform random numbers if the components of \mathbf{Z} are generated by inversion. Its use for reducing the effective dimension in the context of QMC was first proposed by Acworth, Broadie, and Glasserman (1998).

It should be underlined that PCA sampling does not take into account the function g . Even with PCA, one can construct functions g for which $g(\mathbf{Y})$ depends more on Z_d than on Z_1 , for example. Perhaps a better formulation of the problem is to find a decomposition $\mathbf{A}\mathbf{A}^t$ that maximizes the fraction of $\text{Var}[g(\mathbf{Y})]$ coming from Z_1 , then maximize the fraction that comes from Z_2 given Z_1 , and so on. For nonlinear functions g , this is a difficult problem. Imai and Tan (2002), Imai and Tan (2004), Imai and Tan (2006) propose an approximate solution via a linear approximation of g obtained by a first-order Taylor expansion around an

arbitrary point in the unit cube, to compute the j th column of \mathbf{A} so that the corresponding Z_j accounts for the maximal amount of residual variance of the linear approximation. This technique is generally difficult to implement (especially when g is highly nonlinear) and may involve high overhead. We will not use it in this paper.

To simulate a Lévy process at times t_1, \dots, t_s by PCA sampling, we first simulate a Brownian process $\{W(t), t \geq 0\}$ with mean zero and variance parameter σ^2 (so $W(t)$ is normal with mean 0 and variance $\sigma^2 t$), at the observations times $0 = \tau_0 < \tau_1 < \dots < \tau_d$ by PCA sampling, as described earlier. Let $\mathbf{W} = (W(\tau_1), \dots, W(\tau_s))^t$. We then transform the independent increments $W(\tau_j) - W(\tau_{j-1})$ of this process into independent $U(0, 1)$ random variates V_j via

$$V_j = \Phi\left(\frac{W(\tau_j) - W(\tau_{j-1})}{\sigma\sqrt{\tau_j - \tau_{j-1}}}\right), \quad j = 1, \dots, s.$$

Finally, we compute the increments of the Lévy process as $Y(t_j) - Y(t_{j-1}) = G_j^{-1}(V_j)$, where G_j is the distribution function of the j th increment, for $j = 1, \dots, s$. The rationale is to have more of the variance of $(Y(t_1), \dots, Y(t_s))$ coming from the first uniforms U_j , to help RQMC. We call this *PCA sampling with sequential transformation* (PCAS).

Another way of constructing the Lévy process trajectory $(Y(t_1), \dots, Y(t_s))$ from the Brownian process trajectory $(W(\tau_1), \dots, W(\tau_s))$ is as follows. It works under the assumption that we know how to generate $Y(t)$ conditional on $\{Y(t_a) = y_a, Y(t_b) = y_b\}$ by inversion, for arbitrary y_a, y_b , and $t_a < t < t_b$. Let $G_{t_a, y_a, t_b, y_b, t}$ denote the distribution function of $Y(t)$ conditional on $\{Y(t_a) = y_a, Y(t_b) = y_b\}$, let $\Psi_{\tau_a, w_a, \tau_b, w_b, \tau}$ denote the distribution function of $W(\tau)$ conditional on $\{W(\tau_a) = w_a, W(\tau_b) = w_b\}$, and let G_s be the (unconditional) distribution function of $Y(t_s)$. Start by defining

$$Y(t_s) = G_s^{-1}(\Phi(W(\tau_s)/(\sigma\tau_s^{1/2}))),$$

then let

$$Y(t_{s/2}) = G_{t_0, 0, t_s, Y(t_s), t_{s/2}}^{-1}(\Psi_{\tau_0, 0, \tau_s, W(\tau_s), \tau_{s/2}}(W(\tau_{s/2}))),$$

$$Y(t_{s/4}) = G_{t_0, 0, t_{s/2}, Y(t_{s/2}), t_{s/4}}^{-1}(\Psi_{\tau_0, 0, \tau_{s/2}, W(\tau_{s/2}), \tau_{s/4}}(W(\tau_{s/4}))),$$

$$Y(t_{3s/4}) = G_{t_{s/2}, Y(t_{s/2}), t_s, Y(t_s), t_{3s/4}}^{-1}(\Psi_{\tau_{s/2}, W(\tau_{s/2}), \tau_s, W(\tau_s), \tau_{3s/4}}(W(\tau_{3s/4}))),$$

and so on, in the same order as for the Lévy bridge sampling. We call this second construction *PCA sampling with bridge transformation* (PCAB). It can be more appropriate than PCAS in the case where computing the inverse conditional distribution is less expensive than computing the inverse

distribution of an increment. This happens, for example, for a gamma process sampled at equidistant monitoring times (Avramidis and L'Ecuyer 2006, L'Ecuyer and Simard 2006).

For both PCAS and PCAB, we have the choice of the observation times τ_1, \dots, τ_s and of σ^2 . Their choice determines the covariance matrix Σ of \mathbf{W} . The τ_j do not have to be the same as the t_j . We note that multiplying all the τ_j by some factor κ is equivalent to multiplying σ^2 by κ , which means that without loss of generality we could restrict ourselves to $\sigma^2 = 1$.

Another important observation is that for a general Lévy process whose increment over a given time interval has finite and nonzero variance, the variance of the increment is always proportional to the length of the interval: $\text{Var}[Y(t)] = \nu t$ for some positive constant $\nu < \infty$. This follows from the fact that $\text{Var}[Y(\kappa t)] = \kappa \text{Var}[Y(t)]$ for any constant $\kappa > 0$, due to the stationary and independent increments. Moreover, for $0 \leq t_i < t_j$, we always have $\text{Cov}[Y(t_i), Y(t_j)] = \text{Var}[Y(t_i)] = \nu t_i$.

It seems natural, then, to take $\sigma^2 = \nu$ and $\tau_j = t_j$ for all j , so that the covariance matrix Σ of $(W(\tau_1), \dots, W(\tau_s))$ matches exactly the covariance matrix of the vector $(Y(t_1), \dots, Y(t_s))$. That is, we define our Brownian motion $\{W(t), t \geq 0\}$ with the same volatility parameter as the Lévy process, $\sigma^2 = \nu$, and we take the same observation times. We shall adopt this heuristic for our numerical examples.

4 EXAMPLES

4.1 A Gamma Process

A *gamma process* $\{G(t), t \geq 0\}$ with drift parameter $\mu = \alpha/\lambda$ and volatility (or variance) parameter $\sigma^2 = \nu = \alpha/\lambda^2$ is a Lévy process whose increment over a time interval of length t has a gamma distribution with parameters $(t\alpha, \lambda) = (t\mu^2/\nu, \mu/\nu)$, i.e., with mean $t\mu$ and variance $t\nu$. (Note that μ here has a different meaning than earlier.) For $t_a < t < t_b$, the distribution of $(G(t) - G(t_a))/(G(t_a), G(t_b))$ conditional on $(G(t_a), G(t_b))$ is a beta distribution with parameters $((t - t_a)\alpha, (t_b - t)\alpha)$ (Avramidis and L'Ecuyer 2006).

The gamma process G can be simulated by both PCAS and PCAB, with $\sigma^2 = \nu$ and $\tau_j = t_j$ as explained earlier. When the observation times are equally spaced, PCAB runs faster than PCAS (by a factor of about 3 or 4) because it can exploit the fast inversion algorithm of L'Ecuyer and Simard (2006) for the symmetric beta distribution, whereas PCAS requires inversion of the gamma distribution, which is much slower. For comparison, we also simulate the gamma process via sequential sampling and gamma bridge sampling, using inversion in both cases. Again, gamma bridge sampling runs faster than sequential sampling (by

a factor of 3 or 4), thanks to the availability of the fast beta inversion. We insist on always using inversion for compatibility with RQMC.

For a numerical illustration, we take $t_j = j/s$ for $j = 1, \dots, s$, with $s = 32$, $\mu = 1$, and we vary ν : we try $\nu = 0.1, 0.01, 0.001$ and 0.0001 . Smaller values of ν give increments whose distribution is closer to normal. We start with the rather simplistic cost function g defined by $g(\mathbf{Y}) = (G(t_1) + \dots + G(t_s))/s$. The exact value of the expectation is easy to compute in this case: $\mathbb{E}[g(\mathbf{Y})] = \sum_{j=1}^s \mu t_j/s = \mu(s+1)/(2s)$.

For RQMC, we take a Sobol' net with $n = 2^{16}$ points, randomized by a left matrix scramble followed by a random digital shift in base 2 (Owen 2003, L'Ecuyer 2008b, L'Ecuyer 2008a). The following sampling methods are considered for RQMC: sequential sampling (Seq), gamma bridge sampling (Bridge), PCAS, and PCAB. Table 1 gives the variance reduction factors of RQMC compared with MC, defined as the MC variance divided by the RQMC variance for the same number n of simulation runs. To estimate the variance, we made 300 independent replications of the RQMC estimator for all examples. These variance estimators are noisy; the standard errors on the reported variance reduction factors can be 20 percent or more.

We observe that RQMC provides a huge variance reduction (by a factor of around 600,000) with all four sampling methods when ν is small. This is good news. On the other hand, the sequential and bridge sampling methods are doing better than the PCA methods when ν is large. The good performance of sequential sampling may seem surprising. It can be explained by the fact that the performance measure $g(\mathbf{Y})$ in this particular example can be written as a sum of one-dimensional functions:

$$g(\mathbf{Y}) = G(t_0) + \sum_{j=1}^s (G(t_j) - G(t_{j-1}))(s - j + 1)/s,$$

and the sequential method turns out to be equivalent to integrating each term of this sum by a one-dimensional RQMC rule (with inversion) and summing up. For more complicated (nonlinear) functions g , this simplification no longer happens in general (we will see an example of this in Section 4.3). With the bridge sampling, the performance is even better and the explanation is similar, with the difference that the first random numbers have a more important role. We also recall that the bridge and PCAB methods run faster than the other two for this example.

The poor performance of PCA (especially PCAS) for large ν may be linked to the fact that the normal distribution is a very poor approximation of the gamma distribution (the distribution of the increments of the gamma process) when ν is large.

We also tried $s = 64$ and $s = 128$, and the results were similar, except that the performance of PCAS and Seq deteriorates when we increase s , especially for large ν .

PCAS and PCAB were also getting a little better than Seq for small ν .

Table 1: Variance Reduction Factors for the Simulation of a Gamma Process with RQMC vs MC, with a randomized Sobol' net with $n = 2^{16}$.

ν	0.1	0.01	0.001	0.0001
Seq	20000	140000	450000	600000
Bridge	109000	420000	600000	650000
PCAS	200	14000	250000	570000
PCAB	1900	160000	600000	670000

4.2 A Variance-Gamma Process

A *variance-gamma* (VG) process $\{Y(t), t \geq 0\}$ can be defined by

$$Y(t) = W(G(t)),$$

where W is a Brownian motion with drift and variance parameters θ and σ^2 , G is a gamma process with drift and variance parameters 1 and ν , and W and G are independent (Madan, Carr, and Chang 1998, Avramidis, L'Ecuyer, and Tremblay 2003). Madan, Carr, and Chang (1998) argue that replacing the Brownian motion by a VG process in the Black-Scholes option pricing model improves realism.

One way to simulate the VG process by PCA is to first simulate the gamma process by PCA at the s given observation times, then simulate the Brownian motion at the s times specified by the gamma process, again by PCA. If done by inversion, this requires $2s$ uniform random variates; the first s are used for the gamma process and the next s for the Brownian motion. A major drawback of this approach is that the second PCA decomposition must be redone for each simulation run, because the observation times of the Brownian motion are always different.

A second (faster) approach exploits the fact that the VG process can be written as the difference of two independent gamma processes (Madan, Carr, and Chang 1998, Avramidis and L'Ecuyer 2006, Asmussen and Glynn 2007):

$$G(t) = G^+(t) - G^-(t)$$

where G^+ and G^- are independent gamma processes with parameters (μ^+, ν^+) and (μ^-, ν^-) , respectively, with

$$\begin{aligned} \mu^+ &= (\sqrt{\theta^2 + 2\sigma^2/\nu} + \theta)/2, \\ \mu^- &= (\sqrt{\theta^2 + 2\sigma^2/\nu} - \theta)/2, \\ \nu^+ &= (\mu^+)^2 \nu, \text{ and} \\ \nu^- &= (\mu^-)^2 \nu. \end{aligned}$$

The VG process can then be simulated by simulating G^+ and G^- by PCA sampling, one after the other.

An improvement, it seems, would be to apply PCA simultaneously to the two gamma processes. We define a pair of independent Brownian processes W^+ and W^- with the same volatility parameters as G^+ and G^- , and the same observation times t_1, \dots, t_s . Because these two processes are independent, the joint covariance matrix of $(W^+(t_1), \dots, W^+(t_s), W^-(t_1), \dots, W^-(t_s))^t$ is block diagonal with two $s \times s$ blocks, so its PCA decomposition can be obtained by doing a PCA decomposition of each of the two blocks and reordering the eigenvectors by decreasing order of the eigenvalues. We will use this implementation in our experiments.

For a numerical illustration, we take again $t_j = j/s$ for $j = 1, \dots, 32$, with $s = 32$, and we vary v . As a cost function g , we simply take $g(\mathbf{Y}) = (Y(t_1) + \dots + Y(t_s))/s$. The exact value of its expectation is 0.515625 for all values of v . For RQMC, we take the same randomized Sobol' points as before. Table 2 summarizes the results of this numerical experiment. The Seq, Bridge, PCAS, and PCAB methods are defined as in the previous section. For each of them, we try each of the two approaches described above, denoted $W(G(t))$ and $G^+ - G^-$ in the table. We recall that the Seq and PCAS methods are slower than the other ones, because they require inversion of the gamma distribution, and that PCAB with $W(G(t))$ is also slow because it requires too many PCA decompositions.

The Bridge method with $G^+ - G^-$ is the best performer empirically, but PCAB is competitive and can permit RQMC to reduce the variance by a very large factor when v is small. With $W(G(t))$, the PCA methods provide more variance reduction than the Seq and Bridge methods, but this advantage is diminished by larger running times.

Table 2: Variance Reduction Factors for the Simulation of a VG Process with RQMC vs MC, with a randomized Sobol' net with $n = 2^{16}$.

v	0.1	0.01	0.001	0.0001
$W(G(t))$				
Seq	1000	6500	60000	350000
Bridge	300	5700	70000	390000
PCAS	400	22000	460000	620000
PCAB	1500	40000	470000	630000
$G^+ - G^-$				
Seq	20000	150000	520000	730000
Bridge	70000	300000	580000	650000
PCAS	130	7000	190000	490000
PCAB	1600	100000	500000	590000

4.3 Option Pricing Under a Geometric Variance-Gamma Process

We now consider an option pricing problem for an asset whose price evolves according to a geometric VG process S defined by

$$S(t) = S(0) \exp[(r + \omega)t + Y(t)],$$

where Y is a VG process with parameters θ , σ , and v . and $\omega = \ln(1 - \theta v - \sigma^2 v/2)/v$ (Madan, Carr, and Chang 1998, Avramidis and L'Ecuyer 2006). We want to estimate the price of an Asian call option, given by $\mathbb{E}[e^{-rT} \max(\bar{S} - K, 0)]$, where $\bar{S} = (1/s) \sum_{j=1}^s S(t_j)$ and $t_j = jT/s$ for $0 \leq j \leq s$. We try the following parameters: $s = 32$, $\theta = -0.2$, $\sigma = 0.3$, $v = 0.1$, $r = 0.1$, $T = 10$, $K = 101$, and $S(0) = 100$. The exact option value is $\mu \approx 29.916$ and the MC variance is $\sigma^2 \approx 1900$.

Table 3: Variance Reduction Factors for the Simulation of an Asian option with a geometric VG Process, with a randomized Sobol' net with $n = 2^{16}$.

$W(G(t))$	
Seq	31
Bridge	28
PCAS	2100
PCAB	2200
$G^+ - G^-$	
Seq	50
Bridge	1600
PCAS	300
PCAB	2000

Table 3 gives the variance reduction factors of QMC compared with MC. Regarding the speed, the same comments as in the previous section apply here as well. With the $W(G(t))$ approach, the two PCA methods largely dominate the sequential and bridge methods in terms of variance reduction. With the difference of gammas ($G^+ - G^-$), PCAB and the bridge method give the best improvement with RQMC. They are also the fastest methods. We tried other experiments with smaller values of v , and PCAS was performing better (similar to PCAB) in terms of variance reduction.

5 CONCLUSION

We proposed generalizations of PCA sampling for a Brownian motion to an arbitrary Lévy process, under the assumption that the increments can be generated by inversion. We showed empirically that this PCA methodology in conjunction with RQMC can provide significant variance reductions in some cases. On the other hand, in our experiments with the gamma and variance gamma processes, the PCA methods

did not really do better than the gamma bridge sampling. Further experiments with these new PCA methods may (or may not) unveil situations where they provide better improvements than the other existing methods.

A more promising direction, it seems, would be to try to find (approximately) a matrix \mathbf{A} such that $\Sigma = \mathbf{A}\mathbf{A}^t$ and which minimizes the variance of the estimator $g(\mathbf{Y})$. This stochastic nonlinear optimization problem is hard to solve in general, but it could be solved very approximately in a first stage of a simulation experiment. Alternatively, one could use some stochastic approximation procedure to modify \mathbf{A} adaptively during the simulation. This offers ground for future research.

ACKNOWLEDGMENTS

This work has been supported by an NSERC-Canada Discovery Grant and a Canada Research Chair to the first author.

REFERENCES

- Acworth, P., M. Broadie, and P. Glasserman. 1998. A comparison of some Monte Carlo and quasi-Monte Carlo techniques for option pricing. In *Monte Carlo and Quasi-Monte Carlo Methods 1996*, ed. P. Hellekalek, G. Larcher, H. Niederreiter, and P. Zinterhof, Volume 127 of *Lecture Notes in Statistics*, 1–18. New York: Springer-Verlag.
- Asmussen, S., and P. W. Glynn. 2007. *Stochastic simulation*. New York: Springer-Verlag.
- Avramidis, T., and P. L'Ecuyer. 2006. Efficient Monte Carlo and quasi-Monte Carlo option pricing under the variance-gamma model. *Management Science* 52 (12): 1930–1944.
- Avramidis, T., P. L'Ecuyer, and P.-A. Tremblay. 2003. Efficient simulation of gamma and variance-gamma processes. In *Proceedings of the 2003 Winter Simulation Conference*, 319–326. Piscataway, New Jersey: IEEE Press.
- Bertoin, J. 1996. *Lévy processes*. Cambridge: Cambridge University Press.
- Caffisch, 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.
- Devroye, L. 1986. *Non-uniform random variate generation*. New York, NY: Springer-Verlag.
- Glasserman, P. 2004. *Monte Carlo methods in financial engineering*. New York: Springer-Verlag.
- Henderson, S. G., and B. L. Nelson. (Eds.) 2006. *Simulation*. Handbooks in Operations Research and Management Science. Amsterdam, The Netherlands: Elsevier.
- Imai, J., and K. S. Tan. 2002. Enhanced quasi-Monte Carlo methods with dimension reduction. In *Proceedings of the 2002 Winter Simulation Conference*, ed. E. Yücesan, C. H. Chen, J. L. Snowdon, and J. M. Charnes, 1502–1510. Piscataway, New Jersey: IEEE Press.
- Imai, J., and K. S. Tan. 2004. Minimizing effective dimension using linear transformation. In *Monte Carlo and Quasi-Monte Carlo Methods 2002*, ed. H. Niederreiter, 275–292. Berlin: Springer-Verlag.
- Imai, J., and K. S. Tan. 2006. A general dimension reduction technique for derivative pricing. *Journal of Computational Finance* 10 (2): 129–155.
- L'Ecuyer, P. 2004. Polynomial integration lattices. In *Monte Carlo and Quasi-Monte Carlo Methods 2002*, ed. H. Niederreiter, 73–98. Berlin: Springer-Verlag.
- L'Ecuyer, P. 2008a. Quasi-Monte Carlo methods with applications in finance. *Finance and Stochastics*. To appear.
- L'Ecuyer, P. 2008b. *SSJ: A Java library for stochastic simulation*. Software user's guide, Available at <http://www.iro.umontreal.ca/~lecuyer>.
- 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. Szidarovszky, 419–474. Boston: Kluwer Academic.
- L'Ecuyer, P., and R. Simard. 2006. Inverting the symmetrical beta distribution. *ACM Transactions on Mathematical Software* 32 (4): 509–520.
- Madan, D. B., P. P. Carr, and E. C. Chang. 1998. The variance gamma process and option pricing. *European Finance Review* 2:79–105.
- Morokoff, W. J. 1998. Generating quasi-random paths for stochastic processes. *SIAM Review* 40 (4): 765–788.
- Moskowitz, B., and R. E. Caffisch. 1996. Smoothness and dimension reduction in quasi-Monte Carlo methods. *Journal of Mathematical and Computer Modeling* 23:37–54.
- Niederreiter, H. 1992. *Random number generation and quasi-Monte Carlo methods*, Volume 63 of *SIAM CBMS-NSF Regional Conference Series in Applied Mathematics*. Philadelphia, PA: SIAM.
- Owen, A. B. 1998. Latin supercube sampling for very high-dimensional simulations. *ACM Transactions on Modeling and Computer Simulation* 8 (1): 71–102.
- Owen, A. B. 2003. Variance with alternative scramblings of digital nets. *ACM Transactions on Modeling and Computer Simulation* 13 (4): 363–378.
- Wang, X., and I. H. Sloan. 2007. Brownian bridge and principal component analysis: Toward removing the curse of dimensionality. *IMA Journal of Numerical Analysis* 27:631–654.

AUTHOR BIOGRAPHIES

PIERRE L'ECUYER is Professor in the Département d'Informatique et de Recherche Opérationnelle, at the Université de Montréal, Canada. He holds the Canada Research Chair in Stochastic Simulation and Optimization. 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 is currently Associate/Area Editor for *ACM Transactions on Modeling and Computer Simulation*, *ACM Transactions on Mathematical Software*, *Statistical Computing*, *International Transactions in Operational Research*, *The Open Applied Mathematics Journal*, and *Cryptography and Communications*. He obtained the *E. W. R. Steacie* fellowship in 1995-97, a *Killam* fellowship in 2001-03, and became an INFORMS Fellow in 2006. His recent research articles are available on-line from his web page: <http://www.iro.umontreal.ca/~lecuyer>.

JEAN-SÉBASTIEN PARENT-CHARTIER is an M.Sc. student in computational finance. He received his bachelor degree in mathematics from Université de Montréal. His research interests include option pricing and stochastic simulation. He can be reached at parentcj@iro.umontreal.ca.

MAXIME DION is an M.Sc. student in computational finance. He holds a Ph.D. in Physics from Rutgers University. He is interested in option pricing. He can be reached at dionmaxi@iro.umontreal.ca.