A Comparison of Monte Carlo, Lattice Rules and Other Low-Discrepancy Point Sets

Christiane Lemieux* and Pierre L'Ecuyer**

Département d'Informatique et de Recherche Opérationnelle
Université de Montréal, C.P. 6128, Succ. Centre-Ville
Montréal, H3C 3J7, CANADA
lemieux@iro.umontreal.ca, lecuyer@iro.umontreal.ca

Abstract. We explore how lattice rules can reduce the variance of the estimators for simulation problems, in comparison with the Monte Carlo method. To do this, we compare these two methods on option valuation problems in finance, along with two types of (t, s)-sequences. We also look at the effect of combining variance reduction techniques with the preceding approaches. Our numerical results seem to indicate that lattice rules are less affected by the "curse of dimensionality" than the other types of quasi-Monte Carlo methods and provide more precise estimators than Monte Carlo does.

1. Introduction

To evaluate

$$If = \int_{[0,1)^s} f(\boldsymbol{x}) d\boldsymbol{x} \quad , \tag{1}$$

where $f : [0,1)^s \to \mathbb{R}$, the usual method is to take a point set $P = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ and use

$$Qf = \sum_{i=1}^{N} f(\boldsymbol{x}_i)/N \tag{2}$$

as an estimate of (1). To choose the set P when s exceeds 3 or 4, the two main methods are Monte Carlo (MC) and quasi-Monte Carlo (QMC) [31,23]. MC uses an i.i.d. sample of N points randomly and uniformly distributed in $[0,1)^s$. In practice, this is simulated by taking a good pseudorandom generator. QMC methods try to outperform this by taking points which are more regularly distributed than random points and are not independent. Two major approaches have been developed to construct such *low-discrepancy point sets*: lattice rules and (t, m, s)-nets (or (t, s)-sequences). One of the property of these point sets is that inequalities of the form

* Research supported by NSERC-Canada and FCAR-Québec scholarships
** Research supported by NSERC-Canada grants No. ODGP0110050 and SMF0169893.

$$|If - Qf| \leq D(P)V(F),\ f \in E \tag{3}$$

hold for their integration error, for certain classes E of sufficiently smooth functions. In (3), the discrepancy $D(P)$ measures the regularity of P and $V(f)$ measures the variability of f. P is called a low-discrepancy point set if $D(P)$ is small. One way to make this statement more precise is to use a particular definition of $D(P)$, for example the *star-discrepancy* $D^*(P)$ [23], and ask for $D^*(P)$ to be in $O(N^{-1}(\log N)^s)$. The most useful property of this kind of upper bound is not to give an idea of the integration error, because the bound is usually too loose, but to derive criteria for choosing good sets P by providing a quantity to minimize, namely $D(P)$.

In this paper, we explore how lattice rules behave on problems typically solved by simulation. More particularly, we are interested in seeing how they can reduce the variance compared to the MC method. We give numerical examples comparing lattice rules, Monte Carlo, and two types of (t,s)-sequences, combined with some variance reduction techniques and dimension reduction techniques. In Sect. 2, we explain how lattice rules are constructed and how their associated integration error can be estimated. We give in Sect. 3 three examples of selection criteria to choose lattice rules. Section 4 discusses some general concepts about improving the efficiency of the estimators in a quasi-random context and the following section explains the finance problem (the same as in [1]) used for the numerical illustrations of Sect. 6. A conclusion is made in the last section.

2. Lattice Rules

We first recall some standard definitions that can be found in [14,28].

- An s-dimensional **integration lattice**, L, is a discrete subset of $\mathbb{R}^s$ that is closed under addition and substraction and which contains the integer vectors $\mathbb{Z}^s$ as a subset.
- A **shifted lattice** with shift $\Delta \in \mathbb{R}^s$ is the set $L + \Delta = \{\boldsymbol{z} + \Delta : \boldsymbol{z} \in L\}$ for some lattice L.
- The **node set** for a shifted integration lattice, $L + \Delta$, is the set of points in the lattice that fall inside the unit cube, that is, $P = (L + \Delta) \cap [0,1)^s$.
- The **dual lattice** of L is defined as $L^\perp = \{\boldsymbol{k} \in \mathbb{R}^s : \boldsymbol{k} \cdot \boldsymbol{z} \in \mathbb{Z}$ for all $\boldsymbol{z} \in L\}$.
- A **rank-1 lattice** is a lattice whose node set may be expressed as

  $$P = \{\{i\boldsymbol{h}/N\} : i = 0, \ldots, N-1\}\ ,$$

  for some generating vector $\boldsymbol{h} \in \mathbb{Z}^s$, where $\{\boldsymbol{x}\} = \boldsymbol{x} - \lfloor \boldsymbol{x} \rfloor = \boldsymbol{x} \bmod 1$.

In fact, the *rank* of a lattice is the smallest number of vectors that are needed to generate all the points in L. The lattice rules that we use in the numerical examples of Sect. 6 all have rank 1.

When the vector $\boldsymbol{h}$ defining a rank-1 lattice is of the form $\boldsymbol{h} = (1, a, \ldots, a^{s-1}) \bmod N$ and N is prime, the rule is called a *Korobov-type rule*. As mentioned in [9,22], this type of rule is closely related to linear congruential generators (LCGs), because, for these rules, P is equal to the set of all successive overlapping s-tuples generated by an LCG with modulo N and multiplier a, from all possible initial seeds. When a is a primitive element modulo N, the corresponding LCG has a maximal period of $N-1$, so the point set P can be generated simply and quickly by choosing any nonzero seed and taking the first $N-1$ successive s-tuples produced by the LCG, and adding the origin.

An advantage of Korobov-type rules is that if one is interested in the regularity properties of the projections P_u of P on lower-dimensional spaces, where $u \subseteq S = \{1, 2, \ldots, s\}$ has cardinality $|u|$ and $P_u = \{(x_{j_1}, \ldots, x_{j_{|u|}}) : \boldsymbol{x} \in P, u = (j_1, \ldots, j_{|u|})\}$, then one does not need to analyze all 2^s projections P_u, since $P_u = P_v$ if $|u| = |v|$ and $u = v + (k, \ldots, k)$, where $k \in \mathbb{Z}$ i.e., only the difference between successive elements of u are relevant, implying that it is sufficient to look only at projections P_u for $u = (1, j_2, \ldots, j_{|u|})$. In the particular case where one is only interested in projections with successive indices, it is sufficient to look only at the subsets u of the form $u = \{1, 2, \ldots, t\}$, for $1 \leq t \leq s$.

As mentioned in the introduction, the upper bounds on the integration error of the form (3) are often very loose and hard to compute. To have an idea of the error, one should rather try to estimate it, as it is usually done for the MC method. For QMC methods, the usual way of achieving this is to randomize the point set P so that its regularity properties are preserved. One approach is the *Cranley-Patterson randomization technique* [8], originally proposed for lattice rules. The idea of this method is to generate m i.i.d. vectors $\Delta_1, \ldots, \Delta_m$, randomly and uniformly over $[0, 1)^s$, and use them to randomly shift the point set P, modulo 1. Then, compute

$$
\begin{aligned}
X_i &= Qf(\{P + \Delta_i\}), \ i = 1, \ldots, m \\
\bar{X} &= \frac{1}{m} \sum_{i=1}^{m} X_i \\
\hat{\sigma}^2 &= \frac{\sum_{i=1}^{m} (X_i - \bar{X})^2}{m(m-1)}
\end{aligned}
\tag{4}
$$

and use $\hat{\sigma}$ as an estimate of the standard deviation of $\bar{X}$. In [33], it is shown that for any low-discrepancy point set P, if X_i is defined as in (4), then $\mathrm{E}(X_i) = If$ and $\mathrm{Var}\ (X_i) \in O(N^{-2} (\log N)^{2s})$, which for fixed s, is a better asymptotic rate than the usual $O(N^{-1})$ associated with the Monte Carlo method. Though other methods specially designed for (t, m, s)-nets have been proposed (the *scrambled (t,m,s)-nets* in [24]), one can use the Cranley-Patterson technique for them as well.

3. Selection Criteria Derived from Deterministic Bounds

As mentioned in the introduction, point sets P can be chosen by first defining a class E of functions and then selecting the set that minimizes the discrepancy term $D(P)$ in the deterministic upper bound (3). In the case where P is derived from a lattice ($P = L \cap [0,1)^s$), a straightforward way to be brought in this setting is to use the fact that if f has an absolutely convergent Fourier series representation, then one has [28]:

$$Qf - If = \sum_{\mathbf{h} \in L^\perp \setminus \{\mathbf{0}\}} \hat{f}(\boldsymbol{h}) \ , \tag{5}$$

where $\hat{f}(\boldsymbol{h})$ is the Fourier coefficient of f evaluated at $\boldsymbol{h} \in \mathbb{Z}^s$. If we take E as $E_\alpha = \{f : |\hat{f}(\boldsymbol{h})| \le c\|h\|^{-\alpha},\ \forall \boldsymbol{h} \in \mathbb{Z}^s\}$ for a certain norm $\|\cdot\|$ and constants $c > 0$, $\alpha > 0$, then we can define $D(P)$ as $D(P) = \sum_{\mathbf{h} \in L^\perp \setminus \{\mathbf{0}\}} \|\boldsymbol{h}\|^{-\alpha}$ and use this as a criterion to be minimized. For the particular case where we use the product norm $\|\boldsymbol{h}\|_\pi = \prod_{j=1}^s \max(1, |h_j|)$ in the definition of E_α and $D(P)$, we find the well-known

$$\begin{aligned} D(P) = P_\alpha &= \sum_{\mathbf{h} \in L^\perp \setminus \{\mathbf{0}\}} \|\boldsymbol{h}\|_\pi^{-\alpha} \\ &= -1 + \frac{1}{N} \sum_{\mathbf{z} \in P} \prod_{j=1}^{s} \left[1 - \frac{(-1)^{\alpha/2}(2\pi)^\alpha}{\alpha} B_\alpha(z_j)\right] (\alpha \text{ even}) \ , \end{aligned} \tag{6}$$

where $B_\alpha(\cdot)$ is the Bernoulli polynomial of degree α. Hence, P_α can be computed in $O(Ns)$ and has been often used as a figure of merit to choose lattice rules [29,28], usually with $\alpha = 2$.

Another possibility stemming from equation (6) is to find lattice rules minimizing the dominant quantity in this expression or, equivalently, that maximize

$$\rho = \min_{\mathbf{h} \in L^\perp \setminus \{\mathbf{0}\}} \|\boldsymbol{h}\|_\pi \ , \tag{7}$$

the so-called *Zaremba index* [28]. This quantity has also been used as a criterion to select lattice rules, but seems harder to compute than P_α. For this reason, searches based on ρ rarely extend to dimensions greater than 8 or 10.

If we use the euclidean norm $\|\boldsymbol{h}\|_e = \left(\sum_{j=1}^s |h_j|^2\right)^{1/2}$ instead of the product norm $\|\cdot\|_\pi$ in the definition of ρ (as suggested in [9]), we get

$$d_s^{-1} = \min_{\mathbf{h} \in L^\perp \setminus \{\mathbf{0}\}} \|\boldsymbol{h}\|_e \ ,$$

where d_s corresponds to the distance between the farthest family of hyperplanes that cover L, and is the quantity measured by the *spectral test*, a figure of merit commonly used for LCGs ([18,19,16]). See [9] for the connections between ρ and d_s, and arguments supporting the use of d_s.

What is interesting about d_s is that it can be computed rather quickly and a tight lower bound d_s^* is known, so we can define

$$S_s = \frac{d_s^*}{d_s} \in (0, 1] \ .$$

Being able to rescale d_s to a value between 0 and 1 allows one to define the figure of merit

$$M_s = \min_{2 \le t \le s} S_t \ ,$$

which has the nice feature of taking account of all projections P_u with $u = \{1, 2, \ldots, t\}$, $2 \le t \le s$, instead of just looking at the whole set $S = \{1, 2, \ldots, s\}$. In the examples of Sect. 6, the lattice rules used have been chosen as the best LCGs of maximal period with respect to M_8 and can be found in [19], along with many other "good" LCGs.

To conclude this section, we just want to point out that for each of ρ^{-1} and d_s, there corresponds some class of functions, different from E_α, for which each of these two figures corresponds to the $D(P)$ in the upper bound (3). Also, the three criteria presented here are just a few examples among many others available, for general point sets P. See [14] for more on this.

4. Probabilistic Efficiency Improvement Viewpoint

In the preceding section, we have explained how the deterministic upper bound (3) can be used to derive criteria for choosing P. However, when it comes to the analysis of the integration error, the probabilistic approach seems more convenient and for this purpose, we have explained in Sect. 2 one way to randomize a low-discrepancy point set in order to estimate the error.

This randomization yields an unbiased estimator of If taken as the sample mean of an i.i.d. sample of the random variable Qf. From there, the usual approach to compare different estimators is to use the notion of *efficiency*, which for an unbiased estimator $\hat{\mu}$ of μ is defined as

$$\mathrm{Eff}(\hat{\mu}) = (C(\hat{\mu}) \ \mathrm{Var}\ (\hat{\mu}))^{-1} \ .$$

$C(\hat{\mu})$ is the expected cost to calculate $\hat{\mu}$, usually estimated by the CPU time needed to compute it. $\hat{\mu}_1$ is preferred to $\hat{\mu}_2$ if $\mathrm{Eff}(\hat{\mu}_1) > \mathrm{Eff}(\hat{\mu}_2)$.

For specific problems, one can make computer experiments to estimate the efficiency of different estimators, in order to compare them. However, in our case, we would prefer to have general results proving that a given estimator Qf of If has smaller variance than the MC method, assuming that the computing cost is approximately the same for both of them. In simulation, this fits into the framework of *variance reduction techniques (v.r.t)* (see [17,11,5] and other references given there). Since the problems typically solved by simulation can be seen as particular cases of (1), as we shortly explain, one can "translate" this v.r.t. setting to the context of multidimensional integration and, conversely, explain where QMC methods stand in the former setting.

In the context of simulation, one usually wants to estimate $\mathrm{E}(h(\boldsymbol{X}))$, where $\boldsymbol{X}$ is a vector of random variables, for example, the arrival and service times in a queueing system or the successive prices of an asset that evolves randomly, as in the next section. If the number s of variables in $\boldsymbol{X}$ is fixed and if we use inversion to generate this random vector, as is the case in this paper, then, by a simple change of variable, we may write

$$\mathrm{E}(h(\boldsymbol{X})) = \mathrm{E}(f(\boldsymbol{U})) = \int_{[0,1)^s} f(\boldsymbol{U}) d\boldsymbol{U} = If \ ,$$

where F is the distribution function of $\boldsymbol{X}$, $f = h \circ F^{-1}$, and $\boldsymbol{U}$ is the vector of $U(0,1)$ variates used to generate $\boldsymbol{X}$. This brings us to (1), which can be treated by either MC or QMC methods.

The MC method corresponds to pseudorandomly generating the components of $\boldsymbol{U}$ so that they can be analyzed as if they were i.i.d. By making N replications, one can define the estimator

$$X_{MC} = \sum_{i=1}^{N} f(\boldsymbol{U}_i)/N$$

with estimated variance $\sum_{i=1}^{N}(f(\boldsymbol{U}_i) - X_{MC})^2/N(N-1)$.

Applying a v.r.t. amounts to replace $f(\cdot)$ by a function $\tilde{f} : [0,1)^s \to \mathbb{R}$ such that usually, $\mathrm{E}(\tilde{f}(\boldsymbol{U})) = \mathrm{E}(f(\boldsymbol{U}))$, but hopefully, $\mathrm{Var}\ (\tilde{f}(\boldsymbol{U})) \leq \mathrm{Var}\ (f(\boldsymbol{U}))$. For example, using $\tilde{f}(\boldsymbol{U}) = 0.5(f(\boldsymbol{U}) + f(\boldsymbol{1} - \boldsymbol{U}))$ guarantees that $\mathrm{Var}\ (\tilde{f}(\boldsymbol{U})) \leq 0.5\ \mathrm{Var}\ (f(\boldsymbol{U}))$, whenever f is a monotone function with respect to each of its arguments [5,11]. This method is called *antithetic variates*.

Now, instead of using i.i.d. $\boldsymbol{U}_i$, one can take a randomized low-discrepancy point set $P = \{\boldsymbol{V}_1, \ldots, \boldsymbol{V}_N\}$ and define

$$X_{QMC} = \sum_{i=1}^{N} f(\boldsymbol{V}_i)/N \ .$$

From this point of view, using a QMC method may be seen as a v.r.t., which works under some conditions and could be categorized as a *correlation induction* strategy [2]. In that context, the type of result we would like to have is that for a certain class of functions f,

$$\mathrm{Var}\ (X_{QMC}) \leq c\ \mathrm{Var}\ (X_{MC}) \ , \tag{8}$$

where c is a constant with respect to N, preferably not depending on s and less than 1. When P is a scrambled (t, m, s)-net, Owen has proved in [25] that (8) holds, for $f \in \mathcal{L}^2$, but with c increasing with s when $t > 0$. When P is the node set of a lattice, no result of this kind has been proven, at least to our knowledge. However, we can give four short statements that may help to figure out for what conditions a result like (8) might hold for lattice rules.

1. Suppose $\{\boldsymbol{V}_1, \ldots, \boldsymbol{V}_N\}$ is the node set of a randomly shifted lattice $L + \Delta$. If f is periodic and has "frequency" $\boldsymbol{h}$ for some $\boldsymbol{h}$ in $L^\perp$, then $f(\boldsymbol{V}_1) = \ldots = f(\boldsymbol{V}_N)$ and therefore Var $(X_{QMC}) = N$ Var (X_{MC}), which is very bad. If $s = 1$, just take f with period $1/N$ for this result to hold. This means that we must impose conditions on the periodicity of f in order to have (8).
2. When $s = 1$ and $f \in \mathcal{L}^2$ is monotone and such that $\mathrm{Cov}(f(V), f(V + \Delta))$ is convex and symmetric about $\Delta = 1/2$, it has been shown in [12] that Var $(X_{LR}) \leq$ Var (X_{MC}), where $X_{LR} = \sum_{i=1}^{N} f(\{V_i + \Delta\})/N$, $V_i = (i-1)/N$. Their result is even stronger than that, as they have shown that under these conditions on f, among all $P = \{W_1, \ldots, W_N\}$, Var $\left(\sum_{i=1}^{N} f(\{W_i + \Delta\})/N\right)$ is minimized by taking $W_i = V_i, 1 \leq i \leq N$.
3. Since $P = \{\boldsymbol{V}_1, \ldots, \boldsymbol{V}_N\}$ comes from a lattice, we may partition $[0, 1)^s$ in N cells $\boldsymbol{C}_1, \ldots, \boldsymbol{C}_N$ of equal volume, each containing exactly one point. This can be done so that each cell is just a translation of the cell that has the origin in it, modulo 1, i.e., $\boldsymbol{C}_i = \{\boldsymbol{C}_1 + \boldsymbol{V}_i\}, i = 1, \ldots, N$, where $\mathbf{0} \in \boldsymbol{C}_1$. Then, instead of using the same shift Δ for all points in the node set P, we could use N independent shifts Δ_i, $i = 1, \ldots, N$, each uniformly distributed over $\boldsymbol{C}_1$, and define $\boldsymbol{Y}_i = \{\boldsymbol{V}_i + \Delta_i\}$, so that $\boldsymbol{Y}_i \in \boldsymbol{C}_i, i = 1, \ldots, N$. Then, use the estimator
$$\tilde{X}_{LR} = \sum_{i=1}^{N} f(\boldsymbol{Y}_i)/N \ .$$
Using results on the *stratification method* [7], one can show that Var $(\tilde{X}_{LR})$ $\leq$ Var (X_{MC}). Of course, the computation cost required by this modified method is higher than if we use the same shift for all points, but it has the same asymptotic order than for the MC method.
4. When $f \in \mathcal{L}^2$, it can be shown [20] that if X_{QMC} is constructed from a randomly shifted lattice rule $P = (L + \Delta) \cap [0, 1)^s$, then
$$\mathrm{Var}\ (X_{QMC}) = \sum_{\mathbf{h} \in L^\perp \setminus \{\mathbf{0}\}} |\hat{f}(\boldsymbol{h})|^2 \ , \tag{9}$$
whereas
$$\mathrm{Var}\ (X_{MC}) = (1/N) \sum_{\mathbf{h} \in \mathbb{Z}^s \setminus \{\mathbf{0}\}} |\hat{f}(\boldsymbol{h})|^2 \ . \tag{10}$$

This means that X_{QMC} has a smaller variance than X_{MC} if the "average" of the squared norm of the Fourier coefficients over the dual lattice $L^{\perp}$ is smaller than the "average" over all integer vectors $\boldsymbol{h} \in \mathbb{Z}^s$. This is true because the $1/N$ factor in (10) is cancelled by the fact that $L^{\perp}$ has N times less points than $\mathbb{Z}^s$. For the choice of P, this is a result that goes in the same direction as (5), but using a property of the variance of the estimator instead of the deterministic integration error.

Before concluding this section, we want to point out that for QMC methods, there is an additional way to reduce the variance. As mentioned at the end of Sect. 2, if Qf is constructed from a randomly shifted low-discrepancy point set P, then Var $(Qf) \in O(N^{-2}(\log N)^{2s})$ and this upper bound increases with s. This suggests that replacing f by another function g with a smaller *effective dimension* [14,27] might reduce the variance. Roughly, the effective dimension of f can be thought of as the smallest index $t \leq s$ such that the subsets of variables with indices included in $\{1, 2, \ldots, t\}$ contribute to more than $k_1\%$ of the variance of f, where k_1 can be taken as 95 or 99 for example. When this value is known, one can practically replace the dimension s by t when analyzing the asymptotic behavior of the error or variance. Note that for the MC method, reducing the effective dimension is not useful, because the asymptotic order of convergence of the variance is independent of s.

When simulating the path of a Brownian motion $B(\cdot)$ between times 0 and T, at least two methods exist to reduce the effective dimension. The standard method in this case is to associate each $U(0,1)$ variate to the successive increments of $B(\cdot)$. The first alternative is to use the *Brownian bridge* property and then, first generate $B(T)$, then $B(T/2)$, then $B(T/4)$ and $B(3T/4)$ and so on. See [6,1] for details. The second alternative is to use a principal components decomposition. Roughly, this implies that the subset of t variables explaining most of the variability is the one with indices $\{1, 2, \ldots, t\}$. For (t, m, s)-nets, these two alternative methods have the additional advantage of putting more importance on the first variables, which is good because (t, m, s)-nets are usually better equidistributed in the first dimensions. See [1] for more on this idea.

5. A Finance Problem

Finance gives rise to several interesting problems involving the computation of multidimensional integrals. An example is the pricing of *contingent claims*, which are assets whose price depends on the value of other assets, called *underlying assets*. An *asian option* is an example of a contingent claim; its price depends on the mean value of an underlying asset during a certain period of time. More precisely, let

$S(t)$ = value of the underlying asset at time t,

$$\begin{aligned} s &= \text{number of instants where } S(\cdot) \text{ is sampled,} \\ T &= \text{expiration date of the option,} \\ T - T_1 &= \text{length of the time interval that is divided into } s \text{ subintervals of} \\ &\quad\ \text{equal length containing each one sampling point of } S(\cdot), \\ t_j &= T_1 + j(T - T_1)/s,\ j = 1, \ldots, s, \\ K &= \text{strike price of the option.} \end{aligned}$$

If $C_G(t)$ denotes the value of the asian call option based on the *geometric* average at time t, then by definition its final value is

$$C_G(T) = \max\left(0, \left(\prod_{j=1}^{s} S(t_j)\right)^{1/s} - K\right) \ ,$$

which is the difference between the geometric average of the underlying asset sampled at s points equally spaced over the period (T_1, T) and the strike price K, provided that this average exceeds K, otherwise the option has no value.

To model the evolution of the underlying asset, we use the model of Black and Scholes [3], which says that

$$dS(t) = \mu S(t)dt + \sigma S(t)dB(t) \ ,$$

where μ is the mean return parameter of the asset, σ is its volatility parameter and $B(\cdot)$ is a standard Brownian motion. This model involves other assumptions about the market, implying the existence of a unique risk-neutral measure under which

$$dS(t) = rS(t)dt + \sigma S(t)d\tilde{B}(t) \ ,$$

where r is the risk-free interest rate and $\tilde{B}(\cdot)$ is a standard Brownian motion under the risk-neutral measure. The value at time 0 of the asian option is then given by

$$C_G(0) = \tilde{\mathrm{E}}(e^{-rT} C_G(T)) \ , \tag{11}$$

where $\tilde{\mathrm{E}}(\cdot)$ is the expectation under the risk-neutral measure.

The option on the geometric average is convenient from a theoretical point of view, because it can be priced exactly in the Black-Scholes setting. This follows from the fact that the product of lognormal random variables is also a lognormal. However, in practice, it is the option on the arithmetic average that is sold. The final value of this option is

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

and even with the Black-Scholes model, one cannot compute $C_A(0)$ analytically (this would amount to compute the distribution of a sum of lognormal random variables). So one would rely on MC or QMC methods to estimate the s-dimensional integral $C_A(0) = \tilde{\mathrm{E}}(e^{-rT}C_A(T))$. To compute each replicate of $C_G(T)$ or $C_A(T)$, a point $\boldsymbol{x} \in [0,1)^s$ is transformed into a vector of normal random variables which are used to generate the values of the Brownian motion at the observation points t_j. The paths of $S(\cdot)$ are simulated as follows:

$$S(t_j) = S(0)\exp\left((r - 0.5\sigma^2)t_j + \sigma\tilde{B}(t_j)\right)$$

for $j = 1, \ldots, s$, where $\tilde{B}(t) \sim N(0,t)$, a Gaussian random variable with mean 0 and variance t. To reduce the variance of the estimator of $C_A(0)$, $C_G(0)$ can be used as a control variable, as explained in [15]. The geometric average is always smaller than the arithmetic average, but these two quantities are highly correlated and so are the corresponding option prices. As the numerical results in [21] show, this control variable brings a dramatic variance reduction. Antithetic variates may also be used in order to give a small additional reduction of the variance.

6. Numerical Experiments

Following what has been done in [1], we first want to compare the standard method with the two methods mentioned above (Brownian bridge (BB) and principal components (PC)) to reduce the effective dimension. What is new here is that we use these in combination with lattice rules, and not only MC and (t,m,s)-nets. We use Sobol [30] and generalized Faure (GFaure) [10,32] sequences to construct the (t,m,s)-nets, as provided by the software package FINDER [26,27]. Since Sobol sequences are developed in base 2, we took care of using N equal to a power of 2 for them, so that the equidistribution property holds. It is often recommended (as a heuristic) to skip a certain number of points at the beginning of a (t,s)-sequence [13,4]. We skipped the first N points for Sobol and the first 200000 points for GFaure (as in [1]).

Instead of randomizing the low-discrepancy point sets to estimate the integration error and compare the different sets on that basis, here we use the same procedure as in [1]: we take a function f for which If can be computed exactly, calculate the error of our estimate Qf, and repeat this a few times with i.i.d. instances of the problem, which are obtained by generating randomly some of the parameters describing the function. Then, the measure of performance of a method is taken as the root mean square (RMS) relative error obtained that way. The function we use corresponds to the price of an asian option on the geometric average. Its parameters K, r and σ are chosen randomly as in [1] and we take 250 i.i.d. instances with $T_1 = 0$, $T = 1$ year and $S(0) = 100$.

More precisely, for a given method, let $\hat{e}(N,s)$ denote the RMS relative error produced by using N points specified by this method to estimate the price $C_G(0)$ of an option based on the geometric average over s days equally spaced over one year and whose underlying asset has value $S(0) = 100$ at time 0. Then $\hat{e}(N, s)$ is computed as follows:

1. Generate an i.i.d. sample of the parameters $\{(K_v, r_v, \sigma_v), v = 1, \dots, 250\}$ of the option, where each (K_v, r_v, σ_v) is uniformly distributed over the set $[80, 120] \times [0.0, 0.1] \times [0.1, 0.6]$.
2. For $v = 1, \dots, 250$, compute

$$Qf_v = \frac{1}{N} \sum_{i=1}^{N} f_v(\boldsymbol{x}_{i,v})$$

   where $f_v(\boldsymbol{x})$ is such that $C_{G,v}(0) = \int_{[0,1)^s} f_v(\boldsymbol{x}) d\boldsymbol{x}$ and $C_{G,v}(0)$ denotes the price at time 0 of the option with parameters (K_v, r_v, σ_v). The set $\{\boldsymbol{x}_{i,v}, i = 1, \dots, N\}$ of points in $[0,1)^s$ is specified by the method.
3. Compute

$$\hat{e}(N, s) = \sqrt{\sum_{v=1}^{250} \left(\frac{Qf_v - C_{G,v}(0)}{C_{G,v}(0)} \right)^2}. \tag{12}$$

Table 1. RMS relative errors in percent ($100\,\hat{e}(N, s)$) from 250 i.i.d. instances of the problem for the asian option on the geometric average. N is the number of points used for each of the 250 estimators and s is the dimension of the problem.

		Standard				BB			PC		
s	N	MC	Sobol	GFaure	LR	Sobol	GFaure	LR	Sobol	GFaure	LR
10	1024	5.60	**1.01**	1.62	2.00	**0.58**	0.99	1.12	0.56	0.85	**0.52**
	4096	2.54	**0.35**	0.52	1.08	**0.17**	0.24	0.53	**0.16**	0.17	0.19
	16384	1.40	**0.13**	0.25	0.31	**0.06**	0.07	0.21	**0.05**	**0.05**	0.06
50	1024	6.16	**2.44**	3.75	2.46	**0.59**	1.20	1.12	**0.56**	0.77	0.61
	4096	2.80	**0.95**	1.24	1.19	**0.20**	0.41	0.54	**0.16**	0.20	0.21
	16384	1.52	0.40	**0.37**	0.53	**0.06**	0.19	0.25	0.05	**0.03**	0.06
100	1024	5.44	4.75	3.84	**3.36**	**0.61**	1.56	0.96	**0.56**	1.01	0.66
	4096	2.93	**1.31**	2.18	1.33	**0.20**	1.80	0.53	**0.16**	0.58	0.20
150	1024	5.57	6.50	5.92	**3.50**	**0.62**	2.35	1.23	0.56	1.02	**0.55**
	4096	2.98	2.01	**1.35**	1.86	**0.19**	0.32	0.71	**0.16**	0.53	0.21

In Table 1, each block of two or three lines corresponds to a dimension $s = 10, 50, 100, 150$ and on each of these lines, we use either $N = 1024$, $N = 4096$ or $N = 16384$ points for each estimate (in fact, N is taken as the greatest prime integer smaller than those N for all the methods but Sobol).

Table 2. Estimated standard deviations and variance ratios (compared to MC) for the asian option on the arithmetic average. $N = 512$ and $s = 90$.

K		MC	LR	Sobol	GFaure
90	$\hat{\sigma}$	2.80e-04	2.22e-04	2.78e-04	2.64e-04
	$\hat{\sigma}^2/\hat{\sigma}^2_{MC}$	**(1.000)**	**(0.625)**	**(0.981)**	**(0.883)**
100	$\hat{\sigma}$	2.10e-04	1.74e-04	1.98e-04	1.94e-04
	$\hat{\sigma}^2/\hat{\sigma}^2_{MC}$	**(1.000)**	**(0.687)**	**(0.894)**	**(0.858)**
110	$\hat{\sigma}$	1.57e-04	1.41e-04	1.57e-04	1.64e-04
	$\hat{\sigma}^2/\hat{\sigma}^2_{MC}$	**(1.000)**	**(0.804)**	**(1.001)**	**(1.095)**

Table 3. Estimated standard deviations and variance ratios (compared to MC) for the asian option on the arithmetic average. $N = 4096$ and $s = 90$.

K		MC	LR	Sobol	GFaure
90	$\hat{\sigma}$	9.97e-05	4.87e-05	7.62e-05	7.29e-05
	$\hat{\sigma}^2/\hat{\sigma}^2_{MC}$	**(1.000)**	**(0.238)**	**(0.584)**	**(0.534)**
100	$\hat{\sigma}$	7.34e-05	4.05e-05	6.67e-05	5.57e-05
	$\hat{\sigma}^2/\hat{\sigma}^2_{MC}$	**(1.000)**	**(0.304)**	**(0.825)**	**(0.576)**
110	$\hat{\sigma}$	5.49e-05	3.94e-05	4.53e-05	4.71e-05
	$\hat{\sigma}^2/\hat{\sigma}^2_{MC}$	**(1.000)**	**(0.515)**	**(0.680)**	**(0.734)**

Each column corresponds to one of the methods of estimation (MC, LR, Sobol or GFaure) combined with either the standard method, BB or PC to generate the brownian paths. The numbers in the table give the RMS relative errors in percent, i.e., $100\ \hat{e}(N, s)$, as given by (12).

With each of BB and PC, the RMS relative error remains approximately the same for all three QMC methods when the dimension increases, as if the effective dimension was now independent of s. So, reducing the effective dimension also helps lattice rules. Though the usual way of explaining this kind of response is via ANOVA decompositions [14], looking at (5) helps to see what goes on here. BB and PC change f to a function g such that $|\hat{g}(\boldsymbol{h})| \neq 0$ only for $\boldsymbol{h}$ with few nonzero coordinates. Since P has been chosen to maximize $\|\boldsymbol{h}^*\|_e$, the length of the smallest $\boldsymbol{h}$ belonging to $L^\perp$, one can expect that a fair number of these vectors $\boldsymbol{h}$ with few nonzero coordinates are not in $L^\perp$, and therefore their associated Fourier coefficient does not contribute to the error in (5). If $|\hat{g}(\boldsymbol{h})| > |\hat{f}(\boldsymbol{h})|$ for these vectors, larger terms are thus cancelled in (5) when one uses g instead of f. Intuitively, this should help to reduce the error.

We can also see from this table that with the standard method, as s increases, the lattice rules estimator becomes better than Sobol and GFaure, implying that for this type of problems, lattice rules may be more appropriate than these two types of (t, m, s)-nets in large effective dimension. Note also that the advantage of LR is more important when N is small. The same behavior is not observed with BB and PC because with these, the effective dimension remains small even when s is large.

In Tables 2 and 3, we consider the asian option on the arithmetic average instead. We now use the Cranley-Patterson technique with 100 random shifts to estimate the error of LR, Sobol and GFaure. Antithetic variates and the control variable $C_G(0)$ are used to reduce the variance of the estimators, as in [21]. We look at three different values for the strike price K, and for each of these, we give an estimate $\hat{\sigma}$ of the standard deviation of the estimator based on 100 i.i.d. replications of the shifted point set, and below this value, the number in parentheses indicates the ratio of $\hat{\sigma}^2$ to the MC estimator's empirical variance. The other parameters of the option are fixed at $\sigma = 0.2$, $e^r = 1.09$, $S(0) = 100$, $T = 120$ days and $T - T_1 = 90$ days. The dimension s is therefore equal to 90 and we take the number of points $N = 512$ and 4096 in Tables 2 and 3, respectively. Even with the combination of s and N unfavorable to QMC methods in Table 2, LR still improves the variance by non-negligible factors compared to MC, whereas GFaure and Sobol hardly reduce it, confirming what we already observed in Table 1 about the behavior of the different methods when s is large and N is small. In Table 3, LR reduces the variance by a factor of 4 compared to MC when $K = 90$, a significant improvement.

7. Conclusion

Though there is no formal guarantee that randomly shifted lattice rules reduce the variance compared to MC, we have observed in all our numerical experiments that it was the case. The magnitude of the improvement varies depending on the dimension s, the number N of points used and the type of integrand. As expected, reducing the effective dimension of the integrand decreases the error of the lattice rules estimator.

Compared to (t, s)-sequences, it seems that lattice rules outperform these (as well as MC) when the effective dimension is large and the number of points is small, at least for the type of problem and (t, s)-sequences considered in this paper. This setting might seem "unfair" to (t, s)-sequences, since it is known that they need N to be sufficiently large in order to improve upon MC, especially when the dimension is large. But if one is interested in finding efficient estimators with a fixed time budget, as the dimension increases, less and less points are available for the estimation.

References

1. P. Acworth, M. Broadie, and P. Glasserman. A comparison of some Monte Carlo and quasi-Monte Carlo techniques for option pricing. In P. Hellekalek

and H. Niederreiter, editors, *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing*, number 127 in Lecture Notes in Statistics, pages 1–18. Springer-Verlag, 1997.
2. A. N. Avramidis and J. R. Wilson. Integrated variance reduction strategies for simulation. *Operations Research*, 44:327–346, 1996.
3. F. Black and M. Scholes. The pricing of options and corporate liabilities. *Journal of Political Economy*, 81:637–659, 1973.
4. P. Bratley, B. L. Fox, and H. Niederreiter. Implementation and tests of low-discrepancy sequences. *ACM Transactions on Modeling and Computer Simulation*, 2:195–213, 1992.
5. P. Bratley, B. L. Fox, and L. E. Schrage. *A Guide to Simulation*. Springer-Verlag, New York, second edition, 1987.
6. R. E. Caflish, W. Morokoff, and A. Owen. Valuation of mortgage-backed securities using Brownian bridges to reduce effective dimension. *The Journal of Computational Finance*, 1(1):27–46, 1997.
7. W. G. Cochran. *Sampling Techniques*. John Wiley and Sons, New York, second edition, 1977.
8. R. Cranley and T. N. L. Patterson. Randomization of number theoretic methods for multiple integration. *SIAM Journal on Numerical Analysis*, 13(6):904–914, 1976.
9. K. Entacher, P. Hellekalek, and P. L'Ecuyer. Quasi-Monte Carlo node sets from linear congruential generators. Submitted, 1998.
10. H. Faure. Discrépance des suites associées à un système de numération. *Acta Arithmetica*, 61:337–351, 1982.
11. G. S. Fishman. *Monte Carlo: Concepts, Algorithms, and Applications*. Springer Series in Operations Research. Springer-Verlag, New York, 1996.
12. G. S. Fishman and B. D. Wang. Antithetic variates revisited. *Communications of the ACM*, 26:964–971, 1983.
13. B. L. Fox. Implementation and relative efficiency of quasirandom sequence generators. *ACM Transactions on Mathematical Software*, 12:362–376, 1986.
14. F. J. Hickernell. Lattice rules: How well do they measure up? In P. Hellekalek and G. Larcher, editors, *Random and Quasi-Random Point Sets*, volume 138 of *Lecture Notes in Statistics*, pages 109–166. Springer, New York, 1998.
15. A. G. Z. Kemna and C. F. Vorst. A pricing method for options based on average asset values. *Journal of Banking and Finance*, 14:113–129, 1990.
16. D. E. Knuth. *The Art of Computer Programming, Volume 2: Seminumerical Algorithms*. Addison-Wesley, Reading, Mass., third edition, 1997.
17. P. L'Ecuyer. Efficiency improvement via variance reduction. In *Proceedings of the 1994 Winter Simulation Conference*, pages 122–132. IEEE Press, 1994.
18. P. L'Ecuyer. Good parameters and implementations for combined multiple recursive random number generators. *Operations Research*, 47(1), 1999. To appear.
19. P. L'Ecuyer. Tables of linear congruential generators of different sizes and good lattice structure. *Mathematics of Computation*, 68(225):249–260, 1999.
20. C. Lemieux. *L'utilisation de méthodes quasi-Monte Carlo randomisées pour améliorer l'efficacité des estimateurs en simulation*. PhD thesis, Université de Montréal, 1999. In preparation.
21. C. Lemieux and P. L'Ecuyer. Efficiency improvement by lattice rules for pricing asian options. In *Proceedings of the 1998 Winter Simulation Conference*, pages 579–586. IEEE Press, 1998.

22. H. Niederreiter. Multidimensional numerical integration using pseudorandom numbers. *Mathematical Programming Study*, 27:17–38, 1986.
23. H. Niederreiter. *Random Number Generation and Quasi-Monte Carlo Methods*, volume 63 of *SIAM CBMS-NSF Regional Conference Series in Applied Mathematics*. SIAM, Philadelphia, 1992.
24. A. B. Owen. Randomly permuted (t, m, s)-nets and (t, s)-sequences. In H. Niederreiter and P. J.-S. Shiue, editors, *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing*, number 106 in Lecture Notes in Statistics, pages 299–317. Springer-Verlag, 1995.
25. A. B. Owen. Scrambled net variance for integrals of smooth functions. *Annals of Statistics*, 25(4):1541–1562, 1997.
26. A. Papageorgiou and J. Traub. Beating Monte Carlo. *Risk*, 9:63–65, June 1996.
27. S. Paskov and J. Traub. Faster valuation of financial derivatives. *Journal of Portfolio Management*, 22:113–120, 1995.
28. I. H. Sloan and S. Joe. *Lattice Methods for Multiple Integration*. Clarendon Press, Oxford, 1994.
29. I. H. Sloan and L. Walsh. A computer search of rank 2 lattice rules for multidimensional quadrature. *Mathematics of Computation*, 54:281–302, 1990.
30. I. M. Sobol'. The distribution of points in a cube and the approximate evaluation of integrals. *U.S.S.R. Comput. Math. and Math. Phys.*, 7:86–112, 1967.
31. J. Spanier and E. H. Maize. Quasi-random methods for estimating integrals using relatively small samples. *SIAM Review*, 36:18–44, 1994.
32. S. Tezuka. *Uniform Random Numbers: Theory and Practice*. Kluwer Academic Publishers, Norwell, Mass., 1995.
33. B. Tuffin. On the use of low-discrepancy sequences in Monte Carlo methods. Technical Report No. 1060, I.R.I.S.A., Rennes, France, 1996.