Valuation of mortgage-backed securities using Brownian bridges to reduce effective dimension

Russel E. Caflisch, William Morokoff, and Art Owen

The quasi-Monte Carlo method for financial valuation and other integration problems has error bounds of size $O((\log N)^k N^{-1})$, or even $O((\log N)^k N^{-3/2})$, which suggests significantly better performance than the error size $O(N^{-1/2})$ for standard Monte Carlo. But in high-dimensional problems, this benefit might not appear at feasible sample sizes. Substantial improvements from quasi-Monte Carlo integration have, however, been reported for problems such as the valuation of mortgage-backed securities, in dimensions as high as 360. The authors believe that this is due to a lower effective dimension of the integrand in those cases. This paper defines the effective dimension and shows in examples how the effective dimension may be reduced by using a Brownian bridge representation.

1. INTRODUCTION

Simulation is often the only effective numerical method for the accurate valuation of securities whose value depends on the whole trajectory of interest rates or other variables. Standard Monte Carlo simulation using pseudo-random sequences can be quite slow, however, because its convergence rate is only $O(N^{-1/2})$ for N sample paths. Quasi-Monte Carlo simulation, using deterministic sequences that are more uniform than random ones, holds out the promise of much greater accuracy, close to $O(N^{-1})$ in optimal cases. Randomized versions of quasi-Monte Carlo simulation can in some cases bring the typical error close to $O(N^{-3/2})$.

This dramatic improvement in convergence rate has the potential for significant gains both in computational time and in range of application of simulation methods for finance problems. An optimistic reading of the results suggests an effective squaring or even cubing of the sample size N. Large improvements have in fact been found in a number of earlier studies [1, 11, 18], which were all motivated by the results of Paskov [17] on mortgage-backed securities.

Quasi-Monte Carlo simulation is not a magic bullet, however. The asymptotic error magnitudes are the ones it is 'close to' above, multiplied by $(\log N)^k$, where k depends on the dimension s of the simulation. In high dimensions these powers of $\log N$ do not become negligible at any computationally possible sample size. This loss of effectiveness has been documented for a series of test problems in [6–8]. When simulations are cast as integration problems, the resulting integral is often of very high dimension (e.g. dimension 360 for a mortgage of length 30 years), so any loss of effectiveness at high dimensionality can affect them.

Our first goal in this paper is to reconcile two apparently conflicting truths. The first is that quasi-Monte Carlo is not much better than Monte Carlo in high dimensions with practical sample sizes. The second is that quasi-Monte Carlo has been seen to far surpass

Monte Carlo in some high-dimensional examples. It is our view that success in high-dimensional problems tells us more about the integrand than about the method of integration. Some high-dimensional integrands are indeed amenable to quasi-Monte Carlo simulation. Integrands of low 'effective dimension', which we define in two ways below, are of this type. Our second goal is to give an example of a financial simulation, in which one can reduce the effective dimension of an integrand, thereby making quasi-Monte Carlo much more effective.

The outline of this paper as follows. Section 2 gives a brief introduction to quasi-random sequences and their properties, including the Koksma–Hlawka inequality which is the basic estimate on integration error for quasi-Monte Carlo. The dependence on dimension and the character of two-dimensional projections of quasi-random sequences is also discussed. Section 3 introduces some useful decompositions of integrands, and uses them to define two notions of the effective dimension of an integrand. The mortgage-backed security problem is formulated in Section 4. Our main technical tool for formulating the problem with reduced effective dimension is the Brownian bridge representation of a random walk, which is described in Section 5. Computational results for the mortgage-backed security problem are presented in Section 6. Conclusions are discussed in Section 7.

2. QUASI-RANDOM SEQUENCES: DISCREPANCY AND INTEGRATION ERROR

2.1 Basic Properties

The origin of the improved accuracy of quasi-Monte Carlo methods is the improved uniformity of quasi-random sequences. Figure 1 shows two plots, each of 4096 points in two dimensions. The top is a pseudo-random sequence and the bottom is a quasi-random (Sobol') sequence. In the pseudo-random sequence there is clumping of points, which limits their uniformity. The cause of this clumping is that, since points in a pseudo-random sequence are (nearly) independent, they have a certain chance of landing very near to each other. The constructions used for points in a quasi-random sequence, on the other hand, prevent them from clumping together.

The uniformity of a sequence of points in the s-dimensional unit cube $I^s = [0, 1]^s$ can be measured in terms of its *discrepancy*. This is defined by considering the number of points in rectangular subsets of the cube. For a set $J \subseteq I^s$ and a sequence of N points $(x_n)_{n=1}^N$ in I^s, define

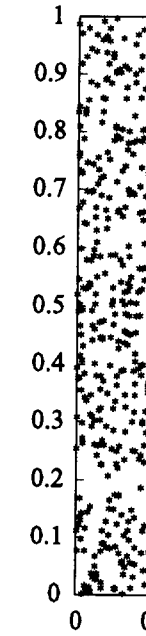
$$R_N(J) = \frac{1}{N}\sum_{n=1}^{N} \mathcal{X}_J(x_n) - m(J).$$

Here $\mathcal{X}_J$ is the characteristic function of the set J, and $m(J)$ is its volume. If E^* is the set of subrectangles with one corner at $(0, \ldots, 0)$, then the star discrepancy is defined as

$$D_N^* = \sup_{J \in E^*} |R_N(J)|. \tag{2.1}$$

Some other discrepancies do not treat the origin differently from the other corners of I^s, and there are L^2 counterparts to the L^∞ definition given above. See Niederreiter [10] for some of these other definitions and Hickernell [4] for some more recent generalizations.

The importance of discrepancy can be seen from the Koksma–Hlawka inequality for



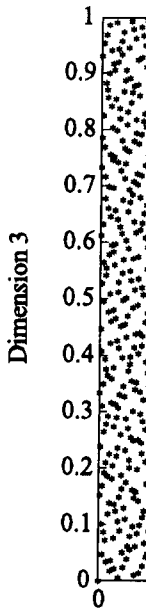


FIGURE 1. Two

success in high-
ethod of integra-
asi-Monte Carlo
ways below, are
on, in which one
asi-Monte Carlo

to quasi-random
which is the basic
imension and the
s also discussed.
s them to define
-backed security
ting the problem
f a random walk,
-backed security
7.

ND

is the improved
of 4096 points in
s a quasi-random
ints, which limits
pseudo-random
very near to each
n the other hand,

$I^s = [0, 1]^s$ can be
mber of points in
ints $(x_n)_{n=1}^N$ in I^s,

If E^* is the set of
efined as

(2.1)

her corners of I^s,
ederreiter [10] for
eneralizations.
ka inequality for

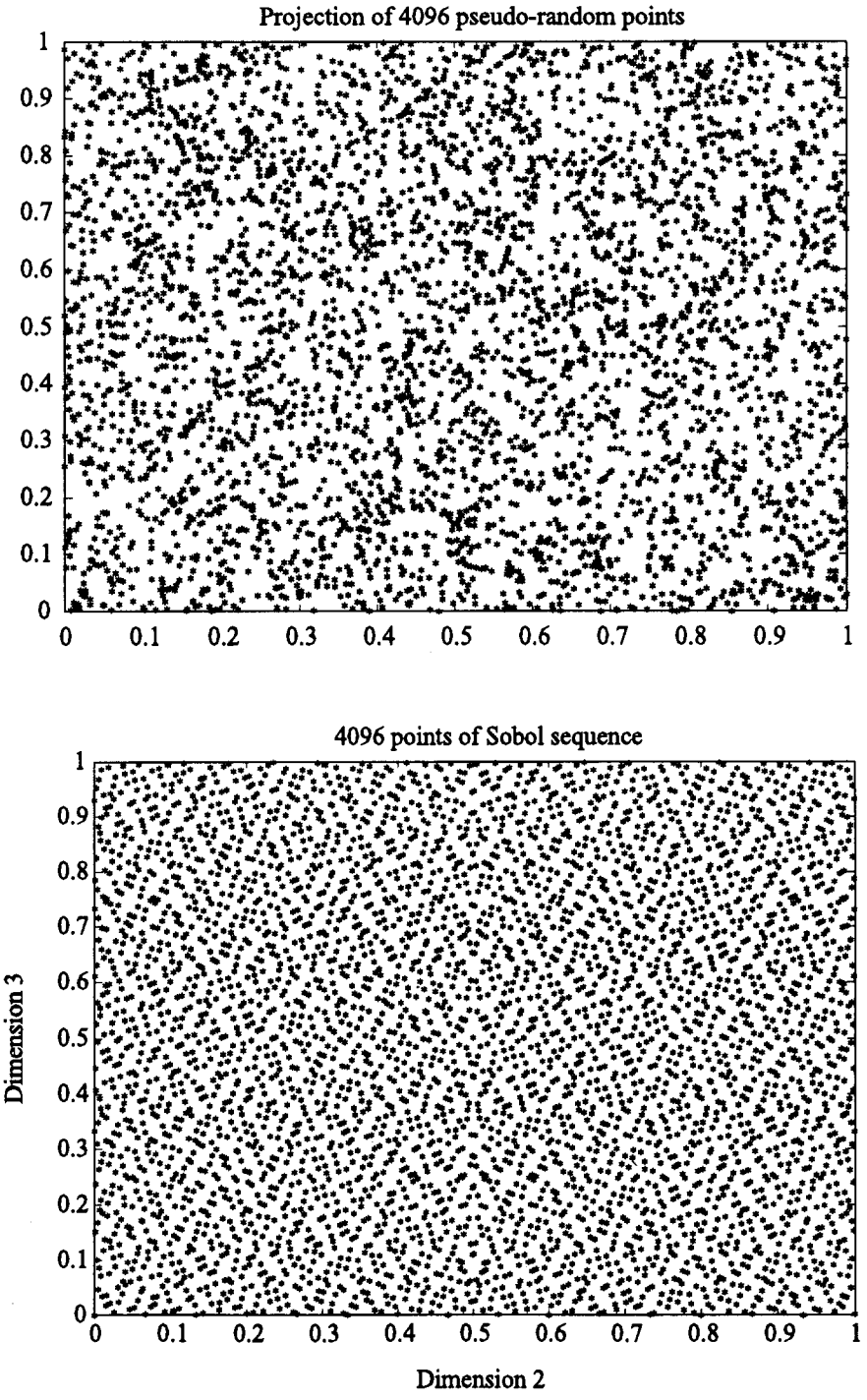


FIGURE 1. Two-dimensional projection of a pseudo-random sequence (top) and a Sobol' sequence (bottom).

integration error. For the integral of a function f on the s-dimensional unit cube, the simulation-based estimate of the integral is

$$I_N(f) = \frac{1}{N}\sum_{n=1}^{N} f(x_n) \tag{2.2}$$

and the integration error is

$$e_N(f) = \frac{1}{N}\sum_{n=1}^{N} f(x_n) - \int_{I^s} f(x)\,\mathrm{d}x. \tag{2.3}$$

The Koksma–Hlawka inequality says that

$$|e_N(f)| \leqslant V(f) D_N^*, \tag{2.4}$$

where $V(f)$ is the variation of f.

In one dimension, the variation is $V(f) = \int_0^1 |\mathrm{d}f|$. The definition in higher dimension is more complicated. Define for all $k \leqslant s$ and all sets of k integers $1 \leqslant i_1 < \cdots < i_k \leqslant s$ the quantity

$$V^{(k)}(f; i_1, \ldots, i_k) = \int_{I^k} \left| \frac{\partial^k f}{\partial t_{i_1} \ldots \partial t_{i_k}} \right|_{t_j = 1, j \neq i_1, \ldots, i_k} \mathrm{d}t_{i_1} \ldots \mathrm{d}t_{i_k}.$$

The variation of f (in the sense of Hardy and Krause) is defined as

$$V(f) = \sum_{k=1}^{s} \sum_{1 \leqslant i_1 < \cdots < i_k \leqslant s} V^{(k)}(f; i_1, \ldots, i_k).$$

The Koksma–Hlawka inequality (2.4) should be compared with the formula for root-mean-square error of Monte Carlo integration using a random sequence. If the sequence (x_n) is uniformly distributed on I^s, then

$$\mathsf{E}[e_N(f)^2]^{1/2} = \sigma(f) N^{-1/2}, \tag{2.5}$$

in which E is the usual expectation and $\sigma(f)$ is the square root of the variance of f given by

$$\sigma(f) = \left(\int_{I^s} [f(x) - \bar{f}]^2 \,\mathrm{d}x \right)^{1/2}, \tag{2.6}$$

with $\bar{f} = \int_{I^s} f \,\mathrm{d}x$.

The error magnitudes (2.4) and (2.5) are similar in that the bound is a product of one term depending on properties of the integrand function and a second term depending on properties of the sequence. The Koksma–Hlawka inequality is an absolute bound, which is more satisfying theoretically than (2.5), an equality in expectation which holds only probabilistically. For practical purposes the preference is reversed. Each factor in (2.4) is incredibly hard to compute, whereas the Monte Carlo variance can be estimated from the same data used to compute $\bar{f}$. Furthermore, the Koksma–Hlawka bound is an inequality that is only tight for a worst-case function f, whose fluctuations are exquisitely matched to the discrepancies in the sequence (x_n), while the Monte Carlo variance estimates the error for the actual f being sampled.

The infinite sequence $(x_n)_{n=1}^{\infty}$ is said to be *quasi-random* if

$$D_N^* \leqslant c(\log N)^k N^{-1},$$

in which the constant c and the logarithmic exponent k may depend on the dimension s. For integration by quasi-random sequences, the Koksma–Hlawka inequality says that the integration error is size $O((\log N)^k N^{-1})$, which for large N is much more accurate than standard Monte Carlo simulation.

Examples of quasi-random sequences have been constructed by Halton, Faure, Sobol', Niederreiter, and others. For a comprehensive discussion, see the monograph of Niederreiter [10].

Although the variation of f requires s derivatives of f, we have found in practice that only a minimal amount of smoothness of f is needed for effectiveness of quasi-Monte Carlo integration. For problems in which f is discontinuous, however, the improvements of quasi-Monte Carlo integration are diminished.

The effectiveness of quasi-Monte Carlo in high-dimensional problems was studied in [7, 8]. There it was shown that the faster convergence rate for quasi-Monte Carlo is generally lost for problems of high dimension. The simplest evidence for this conclusion is seen in the discrepancy as a function of N for various values of the dimension s. For small dimensions, the discrepancy appears to be $O(N^{-1})$, ignoring logarithmic factors, for all N. For large dimensions, the discrepancy behaves initially like $O(N^{-1/2})$ as for a random sequence, taking on an apparent $O(N^{-1})$ rate only for very large values of N.

The transition value of N appears to grow exponentially with the dimension. This has not been rigorously proved, but for quasi-random sequences based on nets as described below, it is in keeping with the definition of nets. In high dimensions, unless one uses a very large number of points, quasi-random sequences are no more uniform than random sequences. Thus the Koksma–Hlawka bound does not imply any advantage for quasi-Monte Carlo with moderate values of N and large dimension s. On the other hand, we have found almost no problems for which quasi-Monte Carlo is worse than standard Monte Carlo.

It is difficult to evaluate the uniformity of a sequence in a high-dimensional space. A necessary but not sufficient condition for uniformity is uniformity of low-dimensional coordinate projections of the sequence. The Sobol' sequence used in the computations below (based on the sequence generated by FINDER) has excellent one-dimensional projections, and in the 360-dimensional case we test, many, but not all, of its two-dimensional projections are very uniform. In Section 3 we show how such partial uniformity is sufficient for the task of integrating functions of low effective dimension.

The one- and two-dimensional projections of (x_n) are easily graphed. Three-dimensional projections can be investigated, with some difficulty, with dynamic graphics tools, such as XGobi [21]. The graph at the bottom of Figure 1 shows a 'good' pairing of dimensions using Sobol's second and third dimensions with his recommended starting values. A 'bad' pairing of dimensions is presented in Figure 2, which shows two higher dimensions (following Sobol's convention for associating dimension with generating polynomial) based on the polynomials $x^7 + x^5 + x^4 + x^3 + 1$ and $x^7 + x^5 + x^4 + x^3 + x^2 + x + 1$ and the starting values (1, 3, 5, 11, 3, 3, 35) and (1, 1, 7, 5, 11, 59, 113), respectively. (The polynomials correspond to dimensions 27 and 32 respectively in FINDER, but the starting values are different.) Although this nonuniformity could go away if these starting values were changed, we have found that this type of nonuniformity is fairly typical of some of the two-dimensional projections of high-dimensional Sobol' and Halton sequences.

The bad behavior seen in the second plot of Figure 2 can be explained in terms of filling in holes. If 8192 (2^{13}) points are used, the plot looks almost identical to what is shown for

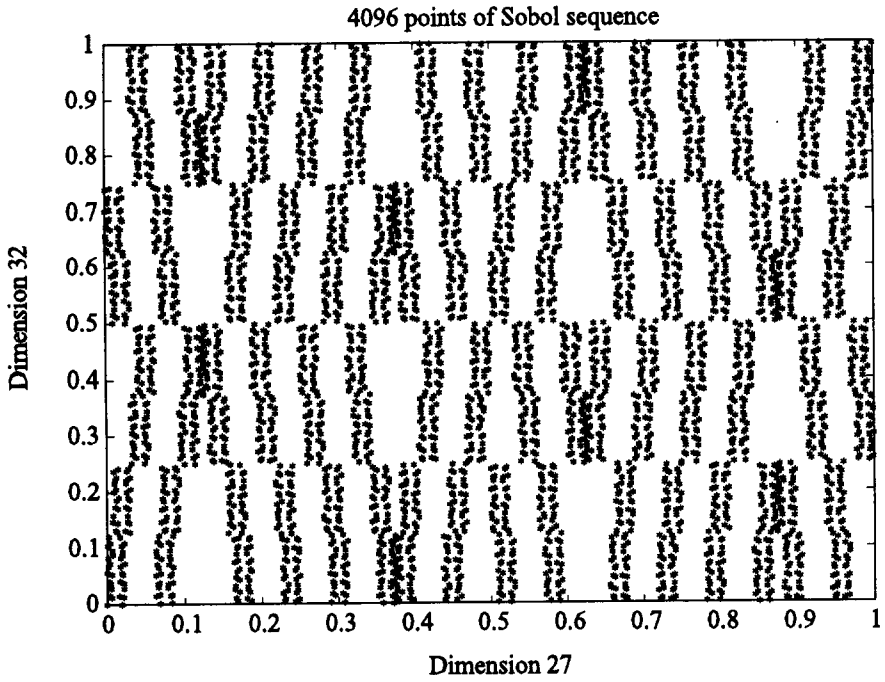


Figure 2. Two-dimensional projection of a Sobol' sequence.

4096. However, the next 8192 points fall only where the gaps appear. Thus by $N = 16{,}384$, the projection plot is almost perfectly uniform. The problem is that the cycle for filling in such holes can be too long.

2.2 (t, m, s)-nets and (t, s)-sequences

The quasi-random points we consider are, or are based on, (t, m, s)-nets and (t, s)-sequences. The definitive reference on this topic is Chapter 4 of the monograph by Niederreiter [10].

An elementary interval in base b is a set of the form

$$J = \prod_{j=1}^{s} \left[\frac{a_j}{b^{k_j}}, \frac{a_j + 1}{b^{k_j}} \right) \subseteq I^s$$

for nonnegative integers a_j and k_j, with $a_j < b^{k_j}$. For $t \geq 0$, a sequence of $N = b^m$ points x_n is a (t, m, s)-net in base b if every elementary interval J in base b of volume b^{t-m} has $R_N(J) = 0$. Given m, s, and b, the smaller t is, the better.

An infinite sequence $(x_n)_{n=1}^{\infty}$ is a (t, s)-sequence in base b if each finite sequence $(x_n)_{n=Ab^m+1}^{(A+1)b^m}$, with $A \geq 0$, is a (t, m, s)-net in base b for all $m \geq 0$.

The net property becomes relevant for $m > t$, that is, $N \geq b^{t+1}$. Below this value of N, any sequence of points in I^s, even identical points, are consistent with the net property. The smallest N at which the net property constrains some fully s-dimensional elementary

subinterval (one with all $k_j > 0$) is b^{t+s}. The asymptotic rate for the discrepancy of nets should therefore start at around $N = b^{t+s}$.

The most widely used constructions of (t, s)-sequences are due to Sobol', Faure, and Niederreiter. The Sobol' sequences are (t, s)-sequences in base $b = 2$. For $s = 360$ the value of t is quite large for Sobol' sequences, in the thousands, according to Niederreiter (personal communication, 1996). The Faure sequences are $(0, s)$-sequences in base b where $b \geqslant s$ is a prime number. For $s = 360$, the smallest value of b is 361.

For Faure sequences the net property starts to be relevant at $N = 361$, but the asymptotics are not relevant until $N = 361^{360} > 10^{920}$. For Sobol' sequences the net property starts to be relevant at $N = b^t$ and the asymptotics are relevant at $N = b^{t+s}$. Even if t is as small as 1000, these two values are larger than 10^{300} and 10^{409} respectively.

Recent constructions of Niederreiter and Xing (1996) hold the promise of (t, s)-sequences in base 2 with $t \doteq s$, a significantly smaller value than previously known possible. Even here though, this suggests that the net property becomes relevant at $N = 2^{361} > 10^{108}$ and the asymptotics become relevant at $N = 2^{720} > 10^{216}$.

Despite the impractically large values of N mentioned above, quasi-Monte Carlo works well on some high-dimensional integrands, using realistic sample sizes. The reason is that low-dimensional views of the quasi-Monte Carlo points can have small discrepancy and some integrands are dominated by low-dimensional structure, as described in Section 3. For example, the points in Figure 1 appear to be a $(t, m, 2)$-net with a much smaller value of t than holds for all 360 dimensions of the points.

2.3 *Scrambled (t, m, s)-nets and (t, s)-sequences*

A scrambled net is a hybrid of quasi-Monte Carlo and Monte Carlo methods, in which a sequence (x_n) is randomized. These are defined and analyzed in a sequence of papers by Owen [14–16]; see also Hickernell [3]. The randomization is carefully constructed in the base b so that (t, m, s)-nets map onto (t, m, s)-nets, (t, s)-sequences onto (t, s)-sequences, and each x_n individually has the uniform distribution on I^s. As a result $I_N(f)$ becomes a random variable with mean $\bar{f}$ and at least quasi-random accuracy.

By replicating the randomization, one can estimate the sampling variance statistically from the same data used to estimate the integral. The variance of $I_N(f)$ is $o(N^{-1})$ for any f with $\int f(x)^2 dx < \infty$.

The precise improvement over Monte Carlo depends on properties of f. If f is well approximated by sums of characteristic functions of elementary intervals of large volume, then $V(I_N(f))$ decreases rapidly. In particular, if f is sufficiently smooth that $h(x) = \partial^s f(x)/\partial x^s$ is Lipschitz-continuous, $|h(x) - h(x')| \leqslant B\|x - x'\|^\beta$ for some $B \geqslant 0$, and $\beta \in (0, 1]$, then $V(I_N(f)) = O(N^{-3}(\log N)^{s-1})$ for scrambled nets on N points. This provides for errors of $O(N^{-3/2}(\log N)^{(s-1)/2})$ in probability, a better rate than that achieved by unscrambled nets. The extra $N^{-1/2}$ factor of accuracy may be attributed to cancellation of some integration errors, which does not happen with deterministic sequences.

The asymptotic rate can be expected to commence at $N = b^{t+s}$, as for unscrambled nets, though with realistic sample sizes, significant benefits can be seen on integrands of lower effective dimension.

3. INTEGRAL DECOMPOSITIONS AND EFFECTIVE DIMENSION

Suppose that we can write

$$f(x) = \sum_{m=0}^{M} f_m(x). \tag{3.1}$$

Then $\bar{f} = \sum_{m=0}^{M} \bar{f}_m$ and $I_N(f) = \sum_{m=0}^{M} I_N(f_m)$, and $e_N(f) = \sum_{m=0}^{M} e_N(f_m)$ in obvious notation.

In decompositions like equation (3.1) we will customarily arrange that $f_0(x) = \bar{f}$ is constant in x and that $\bar{f}_m = 0$ for $m \geqslant 1$. Then

$$e_N(f) = \sum_{m=1}^{M} e_N(f_m), \tag{3.2}$$

because $I_N(f_0) = \bar{f}_0$. If $\int f_m(x) f_k(x)\,dx = 0$ for $m \neq k$, the decomposition (3.1) is said to be orthogonal.

In simple Monte Carlo simulation, the x_n are taken independently from the uniform distribution on $[0, 1]^s$. For an orthogonal decomposition, $\sigma^2(f) = \sum_{m=1}^{M} \sigma^2(f_m)$. Many Monte Carlo methods have the effect of changing the sampling variance of $\bar{f}$ to

$$\frac{1}{N} \sum_{m=1}^{M} \Gamma_m \sigma^2(f_m) \tag{3.3}$$

for some 'gain' constants $\Gamma_m \geqslant 0$. Getting every $\Gamma_m < 1$ implies an improvement over simple Monte Carlo. Even when the Γ_m are not all smaller than 1, a method in which Γ_m is small whenever σ_m^2 is large can be very effective.

A case in point is antithetic sampling. Write $f = f_0 + f_1 + f_2$, where $f_0(x) = \bar{f}$, $f_1(x) = \frac{1}{2}[f(x) + f(1-x)] - \bar{f}$, and $f_2(x) = \frac{1}{2}[f(x) - f(1-x)]$. Here $1 - x$ is interpreted componentwise. This is an orthogonal decomposition into constant, even (symmetric), and odd (antisymmetric) parts. Antithetic sampling involves taking the points x_n and $x_n' = 1 - x_n$ in pairs, for $1 \leqslant n \leqslant \frac{1}{2}N$. One can show that in antithetic sampling $\Gamma_2 = 0$ and $\Gamma_1 = 2$. Antithetic sampling can be anywhere from half as good as simple Monte Carlo (when $f_2 = 0$) to infinitely better than simple Monte Carlo (when $f_1 = 0$). When it is thought that $\sigma_2^2 \gg \sigma_1^2$, as for nearly linear functions, antithetic sampling becomes attractive.

In an analysis of variance (ANOVA) decomposition, $M = 2^s - 1$, and there is one term in (3.1) for each subset of the s components of x. The empty set corresponds to f_0. The term for each subset is a function only of the components within the subset. It is convenient to replace the labels $0, \ldots, M$ by subsets $u \subseteq \{1, \ldots, s\}$. Thus $f(x) = \sum_u f_u(x)$. The ANOVA decomposition is an orthogonal one, so $\sigma^2(f) = \sum_u \sigma_u^2$, where $\sigma_u^2 = \sigma^2(f_u)$. See Owen [12] for definitions, Takemura [22] for some history of this decomposition, and Hickernell [4] for some recent generalizations.

Let $g_t(x) = \sum_{|u|=t} f_u(x)$ for $0 \leqslant t \leqslant s$. Then g_t describes the part of f that is exactly t-dimensional and $\sum_{d=1}^{t} g_d$ describes the part that is at most t-dimensional. The variance of g_t is $\sigma^2(g_t) = \sum_{|u|=t} \sigma_u^2$. If, for example, $\sigma^2(g_1) = 0.99\sigma^2(f)$, then we can conclude that 99% of the variance of f is due to the components of X taken one at a time. In such a case we might say that f is effectively one-dimensional. Likewise, if $\sigma^2(g_1) + \sigma^2(g_2) + \sigma^2(g_3)$ is close enough to σ^2 we might consider f to be effectively three-dimensional. For such an f, a set of

points with good uniformity in every triple of variables will produce an accurate integral estimate, even if the points are not particularly uniform in some quadruples of variables.

Definition 3.1. The effective dimension of f, in the superposition sense, is the smallest integer d_S such that $\sum_{0<|u|\leqslant d_S} \sigma^2(f_u) \geqslant 0.99\sigma^2(f)$.

This notion of effective dimension differs from the one implicitly used in [17], which we state formally as:

Definition 3.2. The effective dimension of f, in the truncation sense, is the smallest integer d_T such that $\sum_{u\subseteq\{1,\ldots,d_T\}} \sigma^2(f_u) \geqslant 0.99\sigma^2(f)$.

The threshold 0.99 is an arbitrary choice, and one might well prefer other values in some settings. For each $d = 1, \ldots, s$, we can define how d-dimensional f is, in the senses above, by the ratios $\sum_{0<|u|\leqslant d} \sigma_u^2/\sigma^2$ and $\sum_{u\subseteq\{1,\ldots,d\}} \sigma_u^2/\sigma^2$. Clearly the fraction of variance that is d-dimensional is higher in the superposition sense than in the truncation sense.

To distinguish the two definitions, consider the function $f = \sum_{i=1}^{s} 2^i x_i$. For this function $d_S(f) = 1$ and $d_T(f) = s$. In general, arranging for all d_S-dimensional projections of $(x_n)_{n=1}^N$ to have low discrepancy ensures an accurate simulation. Alternatively, arranging for the first d_T components of $(x_n)_{n=1}^N$ to have low discrepancy will suffice. The dimension d_S does not depend on the order in which the input variables are indexed, while d_T does.

If f is nearly one-dimensional in the superposition sense, then Latin hypercube sampling will be an extremely effective simulation technique. In Latin hypercube sampling [5], the jth component of x_n is $x_{nj} = [\pi_j(n) - u_{nj}]/N$, where π_j is a (uniform) random permutation of the integers 1 through N and the u_{nj} are independent $U[0, 1]$ random variables. Latin hypercube sampling stratifies each individual dimension, but imposes no higher-dimensional stratification. In fact, a Latin hypercube sample is a scrambled $(0, 1, s)$-net in base N. Stein [20] shows that Latin hypercube sampling is essentially equivalent to having $\Gamma_u = 0$ for subsets u of cardinality $|u| = 1$ and $\Gamma_u = 1$ for $|u| > 1$. The variance contribution of each f_u with $|u| = 1$ is $o(N^{-1})$ and, if that f_u is smooth, the variance contribution is in fact $O(N^{-3})$. We use Latin hypercube sampling below to investigate the fraction of the variance that comes from one-dimensional parts of an integrand.

Some orthogonal array sampling schemes [12, 13, 23] balance all margins up to order t, randomize the higher ones, and have $\Gamma_u \doteq 1_{|u|>t}$. These should be effective on integrands that are, or are nearly, of effective dimension t or less, in the superposition sense.

For nonrandomized quasi-Monte Carlo methods, the decomposition (3.1) does not lead to a variance interpretation, but we may still write

$$|e_N(f)| \leqslant \sum_{|u|\geqslant 1}^{M} |e_N(f_u)| \leqslant \sum_{|u|\geqslant 1}^{M} D^*_{N,u} V_u(f_u), \tag{3.4}$$

where $D^*_{N,u}$ is the discrepancy of the $|u|$-dimensional points obtained by keeping only those components $(x_n)_{n=1}^N$ in u, and $V_u(f_u)$ is the variation of f_u taken as a $|u|$-dimensional function. We believe that many successes of quasi-Monte Carlo methods on high-dimensional problems can be attributed to a low effective dimension of the integrand, in one or both of the senses above. In such cases, arranging for small values of $D^*_{N,u}$ to coincide with the few large values of σ_u^2 and large values of $D^*_{N,u}$, if any, to coincide with small values of σ_u^2 (which ordinarily implies small $V(f_u)$) should produce accurate results.

As N increases, more and more of the terms in equation (3.4) should switch from the

Monte Carlo rate $N^{-1/2}$ to a quasi-Monte Carlo rate $N^{-1}(\log N)^k$. While this is taking place, there may be an appearance of an asymptotic rate better than $N^{-1/2}$. If, however, the very highest dimensional term is nonzero, then one can expect that the errors are bounded below by a small multiple of $N^{-1/2}$ until N reaches the sometimes astronomical sample sizes b^{t+s} as described in Section 2.2.

Scrambled net sampling can be analyzed in terms of a square wavelet decomposition applied to each term in the ANOVA decomposition [15]. The result is that the contribution of f_u decreases very rapidly after $N = b^{t+|u|}$. This explains how the scrambled net variance is $o(N^{-1})$. Large gains can be expected if f has sufficiently low dimension in the superposition sense.

Classical Monte Carlo techniques often replace the integrand f by another one with the same integral and a smaller variance. The ANOVA decomposition above suggests two more ways to improve integration: reducing the effective dimension in either of the two senses above. In high-dimensional problems, reducing both sorts of effective dimension pays off. It is, for example, easier to arrange low discrepancy for all three-dimensional projections of 16 variables than for all three-dimensional projections of 360 variables. We believe that the Brownian bridge discretization in Section 5 succeeds for these reasons on the mortgage-backed securities problem introduced in Section 4.

4. MORTGAGE-BACKED SECURITIES

Consider a security backed by mortgages of length M months with fixed interest rate i_0, which is the current interest rate at the beginning of the mortgage. The present value of the security is then

$$\mathrm{PV} = \mathsf{E}[v] = \mathsf{E}\left[\sum_{k=1}^{M} u_k m_k\right], \tag{4.1}$$

in which E is the expectation over the random variables involved in the interest rate fluctuations. The variables in the problem are as follows.

u_k = discount factor for month k,

m_k = cash flow for month k,

i_k = interest rate for month k,

w_k = fraction of remaining mortgages prepaying in month k,

r_k = fraction of remaining mortgages at month k,

c_k = (remaining annuity at month k)/c,

c = monthly payment,

ξ_k = an $\mathrm{N}(0, \sigma)$ random variable.

This notation follows that of Paskov [7], except that our c_k corresponds to his a_{M-k+1}. Several of these variables are easily defined:

$$u_k = \prod_{j=0}^{k-1}(1+i_j)^{-1},$$
$$m_k = cr_k[(1-w_k)+w_k c_k],$$
$$r_k = \prod_{j=1}^{k-1}(1-w_j),$$
$$c_k = \sum_{j=0}^{M-k}(1+i_0)^{-j}.$$

Following Paskov, we use models for the interest rate fluctuations and the prepayment rate given by

$$\begin{aligned} i_k &= K_0\, e^{\xi_k}\, i_{k-1} \\ &= K_0^k\, e^{\xi_1+\cdots+\xi_k}\, i_0, \\ w_k &= K_1 + K_2 \arctan(K_3 i_k + K_4), \end{aligned} \tag{4.2}$$

in which K_1, K_2, K_3, K_4 are constants of the model. The constant $K_0 = e^{-\sigma^2/2}$ is chosen to normalize the log-normal distribution, i.e. so that $E[i_k] = i_0$. The initial interest rate i_0 is an additional constant that must be specified.

In this study we do not divide the cash flow of the security among a group of tranches, as in [17], but only consider the total cash flow. Nevertheless, the results should be indicative of a more general computation involving a number of tranches.

The expectation PV can be written as an integral over $\mathbb{R}^M$ with Gaussian weights

$$g(\xi) = (2\pi\sigma^2)^{-1/2} e^{-\xi^2/2\sigma^2}. \tag{4.3}$$

This is transformed into an unweighted integral by a mapping $\xi = G(x)$ with $G'(x) = g(\xi)$, which takes a uniformly distributed variable x to an $N(0, \sigma)$ variable ξ. The formula for PV is

$$\begin{aligned} \mathrm{PV} &= \int_{\mathbb{R}^M} v(\xi_1, \ldots, \xi_M) g(\xi_1) \ldots g(\xi_M)\, d\xi_1 \ldots d\xi_M \\ &= \int_{[0,1]^M} v\big(G(x_1), \ldots, G(x_M)\big)\, dx_1 \ldots dx_M. \end{aligned} \tag{4.4}$$

Note that, in quasi-Monte Carlo evaluation of an expectation involving a stochastic process with M time steps, the resulting integral is M-dimensional.

The parameter values i_0, K_1, K_2, K_3, K_4, σ used in [17] are proprietary, and not known to us. In the numerical study below, we have used two sets of values for them. The first set, chosen to be plausible to us, turned out by inspection to be very nearly a linear function of the Gaussian increment random variables ξ_k. We refer to this set of parameters as the 'nearly linear example'. For this case, the parameters are

$$(i_0, K_1, K_2, K_3, K_4, \sigma^2) = (.007, .01, -.005, 10, .5, .0004). \tag{4.5}$$

In the second example, the 'nonlinear example', the parameters are

$$(i_0, K_1, K_2, K_3, K_4, \sigma^2) = (.007, .04, .0222, -1500.0, 7.0, .0004). \tag{4.6}$$

This example also has a large linear component, but the effect is less extreme. While perhaps less plausible, it represents a more challenging integrand. The monthly interest rate i_0 corresponds to a yearly rate of 8.4%. The variance in interest rate increments σ^2 leads to yearly fluctuations of size 0.5%. In the nearly linear example, the prepayment rate is nearly linear in the interest rate, in the range of interest; whereas for the nonlinear example, the prepayment rate has a step increase when the interest rate falls much below i_0. In both examples the length of the loans is taken to be 30 years ($M = 360$).

Monte
place, t
the ve
bound
sampl

Scr
appli
f_u de
$o(N^-$
sens

C
sam
wa
ab
is,
va
B
b

5. BROWNIAN BRIDGE

Since Brownian motion is a Markov process, it is most natural to generate its value $b(t + \Delta t_1)$ as a random jump from a past value $b(t)$ as

$$b(t + \Delta t_1) = b(t) + \sqrt{\Delta t_1}\, \nu, \tag{5.1}$$

in which ν is an N(0, 1) random variable. On the other hand, the value $b(t + \Delta t_1)$ can also be generated from knowledge of both a past value $b(t)$ and a future value $b(T = t + \Delta t_1 + \Delta t_2)$, with $0 \leqslant \Delta t_i$, according to the *Brownian bridge* formula

$$b(t + \Delta t_1) = ab(t) + (1 - a)b(T) + c\nu, \tag{5.2}$$

in which

$$a = \Delta t_2/(\Delta t_1 + \Delta t_2), \qquad c = \sqrt{a\Delta t_1}. \tag{5.3}$$

Note that $c \leqslant \sqrt{\Delta t_1}$, so that the variance of the random part of the Brownian bridge formula (5.2) is less than that in (5.1).

The standard method of generating a random walk $y_k = \sigma b(k\Delta t)$ is based on the updating formula (5.1). The initial value is $y_0 = 0$. Each subsequent value y_{k+1} is generated from the previous value y_k using formula (5.1), with independent normal variables ν_k.

Another method, which we refer to as the *Brownian bridge discretization* can be based on (5.2). Suppose we wish to determine the path $y_0, \ldots, y_M$, and for convenience assume that M is a power of 2. The initial value is $y_0 = 0$. The next value generated is $y_M = \sigma\sqrt{M\Delta t}\,\nu_0$. Then the value at the midpoint $y_{M/2}$ is determined from the Brownian bridge formula (5.2). Subsequent values are found at the successive midpoints, i.e. $y_{M/4}, y_{3M/4}, y_{M/8}, \ldots$. The procedure is easily generalized to general values of M.

Although the total variance in this representation is the same as in the standard discretization, much more of the variance is contained in the first few steps of the Brownian bridge discretization due to the reduction in variance in the Brownian bridge formula. This reduces the effective dimension of the random walk simulation, which increases the accuracy of quasi-Monte Carlo. Moskowitz and Caflisch [5] applied this method to the evaluation of Feynman–Kac integrals and showed the error to be substantially reduced when the number of time steps, which is equal to the dimension of the corresponding integral, is large. Since the mortgage-backed securities problem described above depends on a random walk, and can be written as a discretization of a Feynman–Kac integral, we were naturally led to apply the Brownian bridge discretization to this problem.

ξ_k = an N(0, σ) random ...

6. N

6.1 N

The va
mortga
Carlo v
that the
41.84 =
$\sigma_2^2 = 4$
99.98%

Simil
that rou
structu
nearly
varianc
numeri
ables. A
slight d
that the

We n
functio
square
tions a
number
comput
they are
differen
quadrat
Gaussia
terms o

In the
antitheti
ations. I
evaluati
dominan
efficientl

First,
Figure 3
points, w
dramatic
the code
50 dime
rest pseu
random
sense, th

The th

6. NUMERICAL RESULTS

6.1 Nearly Linear Example

The value PV for this example was calculated to be 131.78706. The mean length of a mortgage in this case is 100.9 months and the median length is 93 months. The Monte Carlo variance of PV is 41.84 and the variance in antithetic sampling is 0.014. This suggests that the function is very nearly an odd function of the Gaussian increments. In fact, solving $41.84 = \sigma_1^2 + \sigma_2^2$, with $0.014 = 2\sigma_1^2$, provides the rough estimates $\sigma_1^2 \doteq 0.007$ and $\sigma_2^2 = 41.833$, so that the odd or antisymmetric part of this integrand provides about 99.98% of the variation.

Similarly the variance in Latin hypercube sampling is about 0.0155, from which we find that roughly $(41.84 - .0155)/41.84 \doteq 99.96\%$ of the variation comes from one-dimensional structure. This function is effectively one-dimensional in the superposition sense, and it is nearly antisymmetric. The percentages quoted above are based on ratios of sampling variances and may not be exact, but both of these findings agree with what we found by numerical inspection: this function is very nearly linear in the Gaussian increment variables. Application of Latin hypercube sampling to the antithetic integrand leads to only a slight decrease in variance compared to the nonantithetic case, which we interpret to mean that the multidimensional part of the integrand is not predominantly odd.

We now describe the accuracy of various integration methods for this problem as a function of N, the number of paths. For each of these results, we present the root-mean-square of the error among 25 computations. For Monte Carlo methods the 25 computations are statistically independent, at least to the level possible using pseudo-random numbers. The Sobol' calculations for each of the 25 runs and for each value of N was computed using different nonoverlapping sub-sequences of the Sobol' sequence. Because they are not a sample from any population, the root-mean-square error presented is the difference between the values obtained and a 'gold standard' obtained from using the quadratic terms of a Taylor series expansion of the integrand about the origin in the Gaussian coordinates as a control variate and $N \approx 3.2$ million. The results are plotted in terms of error versus N, both in log base 10.

In these plots, for the antithetic computations, N refers to the number of times the antithetic integrand $\frac{1}{2}[f(x) + f(1-x)]$ is evaluated. This corresponds to $2N$ function evaluations. Plotting the antithetic runs versus $2N$ would be more appropriate when function evaluation is the dominant cost and plotting versus N when generating the x_n is the dominant cost. We don't attempt to plot CPU time on the axis, as this can depend on how efficiently a method is implemented.

First, we perform straightforward Monte Carlo evaluation, with results plotted in Figure 3. The top curve shows results from Monte Carlo using standard pseudo-random points, with the error decreasing at the expected rate of $N^{-1/2}$. The second curve shows a dramatic improvement using the 360-dimensional Sobol' sequence (generated with part of the code FINDER). In a separate calculation (not plotted), it was found that if only the first 50 dimensions (using the standard discretization) were taken to be quasi-random and the rest pseudo-random, the size of the error decreased slightly compared with the purely random case, and the apparent convergence rate remained $N^{-1/2}$. So, in the truncation sense, the dimensionality is not below 50.

The third curve in Figure 3 shows the results of combining the quasi-random sequence

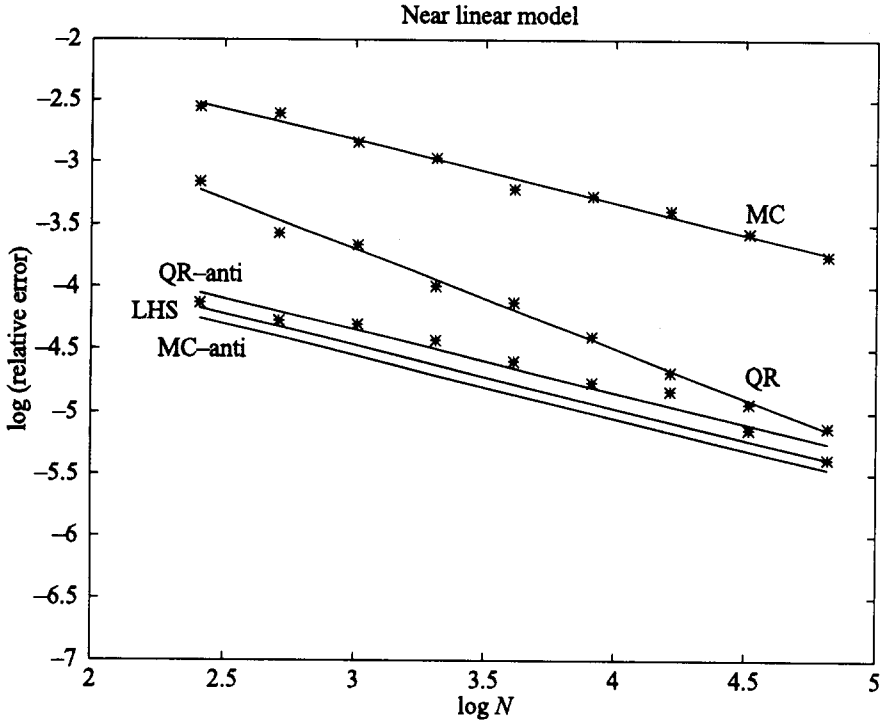


Figure 3. Error versus N (log base 10) for the nearly linear problem, in the original representation.

with antithetic sampling. This leads to a sizeable reduction in the error size as the dominant antisymmetric part has been removed. However, the improved convergence rate, characteristic of low-dimensional quasi-Monte Carlo methods, also disappears. Finally, reference lines for Latin hypercube sampling and antithetic random Monte Carlo are shown. These both effectively remove the one-dimensional linear elements of the integrand, with antithetic variates killing it off exactly, while this error decreases like $O(N^{-3/2})$ for the Latin hypercube case, leaving the remaining errors to converge at the standard $N^{-1/2}$ rate. The quasi-random error, while outperforming simple random Monte Carlo on the one-dimensional elements, still can only achieve the $O(N^{-1})$ rate in the optimal case. Both Latin hypercube and antithetic random outperform the antithetic quasi-random sequence slightly. This may be because the quasi-random sequence is performing at worse than the Monte Carlo level on the higher-dimensional part of the integrand.

A scrambled (0, 360)-sequence in base 361, with N a small multiple of 361, behaves like Latin hypercube sampling. When N nears $361^2 = 130321$, some bivariate sources of variance disappear. For intermediate multiples of 361, some of the bivariate effects are reduced by a multiple $N/361^2$. On this function, the results for the scrambled (0, 360)-sequence were essentially that same as for Latin hypercube sampling, and thus they are not plotted in Figure 3. When applied to the antithetic integrand, the scrambled sequence showed a slight improvement over the antithetic Latin hypercube sampling; however, no significant gains

are achieved in either case with antithetic sampling, because of the increased computation time required.

Our results are consistent with the results of Paskov [17, 18] and Ninomiya and Tezuka [11]. They contradict the observation that the effectiveness of quasi-Monte Carlo is lost in high dimensions. However, we argue below that the improvement is almost entirely due to improved integration of the one-dimensional parts of the integrand.

Next we consider the Brownian bridge version of the integrand. The reformulated integrand has the same mean and the same variance as the original, but more of the structure is packed into the first few dimensions. This should help the Sobol' sequence because it would only need to have small discrepancy among the first few dimensions. This encoding decreases the variance in Latin hypercube sampling from 0.0155 to 0.00963, suggesting that the Brownian bridge encoding has made the integrand even more inherently one-dimensional.

Figure 4 shows the results from Sobol' sequence integration in the Brownian bridge representation, with and without antithetic sampling. Also shown are reference lines for simple Monte Carlo, which is not affected by the change of representation, and for Latin hypercube sampling with the Brownian bridge. Again antithetic sampling does not substantially improve Latin hypercube sampling here. For the Brownian bridge Sobol' sequence without antithetic sampling, the results are essentially the same as for the

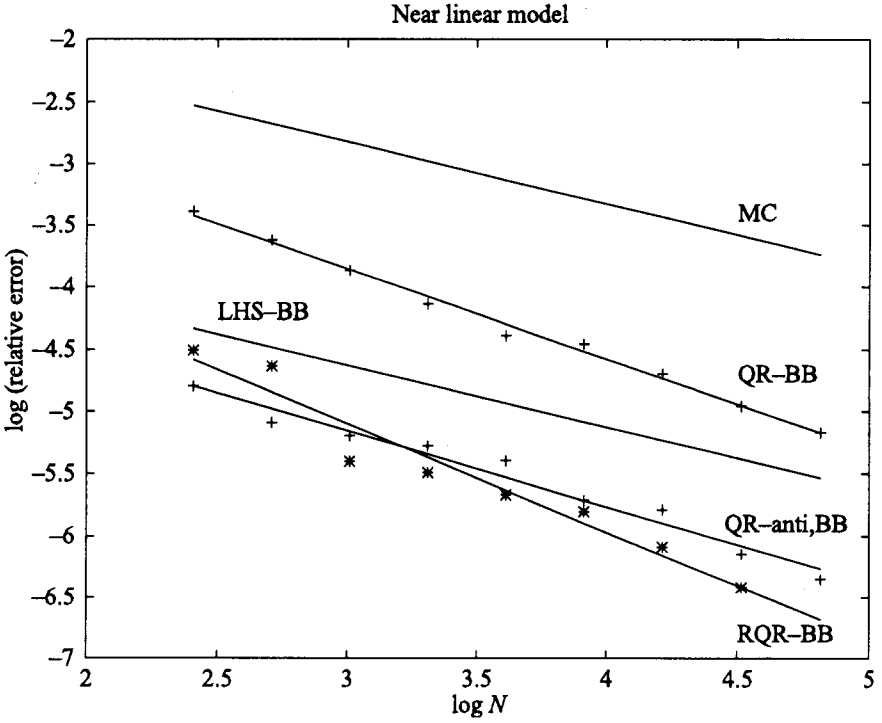


Figure 4. Error versus N (log base 10) for the nearly linear problem, in the Brownian bridge representation.

standard representation. This is because, for the dominant linear one-dimensional elements, the Brownian bridge representation simply rearranges the weights on the elements, but the sum remains constant. Because the errors associated with each one-dimensional projection of the Sobol' sequence are nearly identical, no improvement is seen.

In the Brownian bridge formulation, the Sobol' sequence with antithetic sampling is much better than either antithetic variables or Latin hypercube sampling. This suggests that it must be capturing some higher-dimensional antisymmetric structure, most probably among the first few variables.

The theory of scrambled (0, 360)-sequences in base 361 predicts that they will not be much better than Latin hypercube sampling until $N = 361^2 = 130321$, which is beyond the range we explore here. But by packing most of the structure into the first few dimensions of the integrand, we can consider methods in which scrambled nets are used on the most important dimensions and something else is used on the rest. Such methods have been proposed before: Spanier [19] describes a scheme in which quasi-Monte Carlo methods are used on the first few dimensions of an integrand and simple Monte Carlo is used on the rest, and Owen [13] considered augmenting a randomized orthogonal array with further dimensions taken from Latin hypercube samples.

We considered using a scrambled (0, 32)-sequence in base 32 for the first 32 dimensions and Latin hypercube sampling on the last 328 dimensions. For $N = 32^m$ one gets a scrambled $(0, m, 32)$-net in base 32 for the first 32 dimensions and Latin hypercube sampling for the rest. This will integrate the one-dimensional part of the function and much of the m-dimensional structure (superposition sense) of the first 32 dimensions, with the rest of the structure being integrated at the Monte Carlo rate. But (0, 32)-sequences can be stopped early or extended as necessary, whereas Latin hypercube sampling requires a prespecified number N of runs. As a compromise, we ran a scrambled (0, 32)-sequence for the first 32 dimensions and took repeated independent Latin hypercube samples of size 1024 for the last 328 dimensions. Such a simulation can be conveniently stopped at any multiple of 1024 runs. The results are shown on Figure 4, labelled as RQR-BB for randomized quasi-random with Brownian bridge. All pairs of two variables among the first 32 variables start to become balanced at sample size $N = 32^2 = 1024$, and, similarly, all triples of variables among the first 32 variables start to become balanced at sample size $N = 32^3 = 32768$. This leads to results which are similar to the antithetic Sobol' with Brownian bridge results. However, the convergence rate for the scrambled net in the Brownian bridge representation appears to be larger, leading to greater accuracy at large N.

We were able to achieve still better results than those shown here by approximating the integrand by a quadratic function about the origin in the Gaussian coordinates and using this function as a control variate with antithetic random Monte Carlo. The quadratic terms were calculated by evaluating all $360 * 361/2$ second derivatives of the integrand.

6.2 Nonlinear Example

We next consider the nonlinear example and display the same error curves. The mean value of PV for this problem is 130.712365. This value was obtained by averaging 200 independent Cranley–Patterson randomizations of 2^{19} fixed paths, generated by using the Sobol' sequence with the Brownian bridge ordering. These randomizations are due to Cranley and Patterson [2], and Tuffin [24] appears to be the first to realize their utility on nets. The certainty of this answer can be estimated by considering a six standard deviation range,

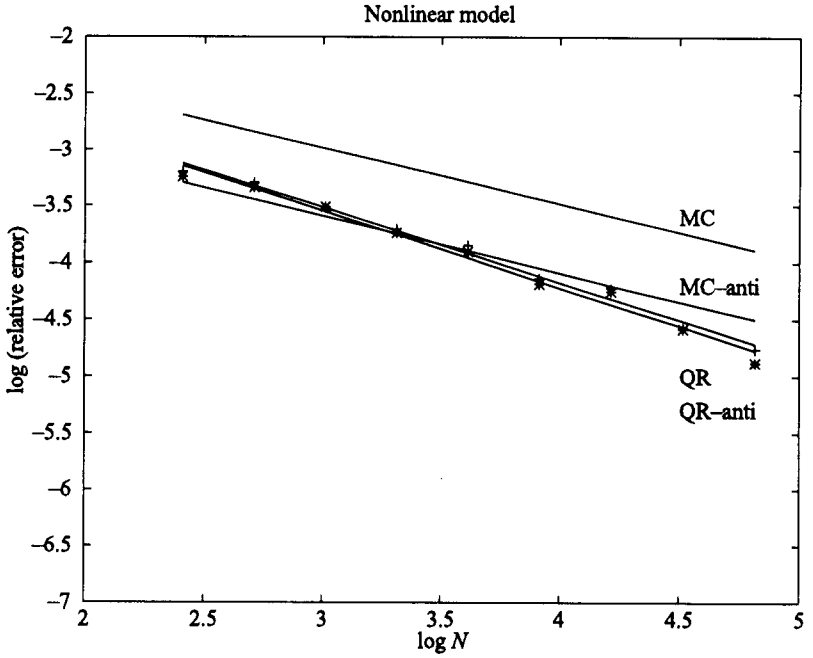


Figure 5. Error versus N (log base 10) for the nonlinear problem, in the original representation.

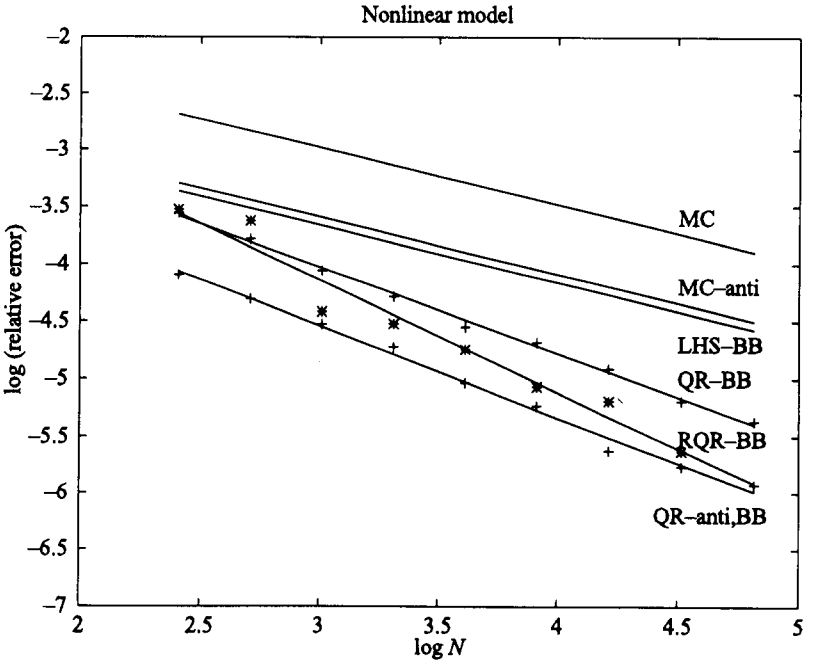


Figure 6. Error versus N (log base 10) for the nonlinear problem, in the Brownian bridge representation.

which was determined to be (130.712348, 130.712382). The error curves in Figures 5 and 6 are based on using this value as the exact solution.

The mean length of a mortgage in this case is 76.5 months and the median length is 58 months. The variance of PV in this value is 18.54 and the variance in the antithetic computation of this value is 1.127. Thus the function is about 97.7% antisymmetric. The variance under Latin hypercube sampling is 1.087, so that the function is only about 94.1% one-dimensional. This may seem like a lot of one-dimensional structure, but, compared with the previous example, the proportion of higher-dimensional structure is greatly increased.

As in the nearly linear example, antithetic sampling and Latin hypercube sampling in combination do not work better than separately. For this integrand as for that one, they each appear to remove the same source of error. In fact, for the nonlinear example, both give almost exactly the same errors.

Figure 5 shows the error reference lines from Monte Carlo and antithetic Monte Carlo sampling. Both Latin hypercube sampling and the randomized (0, 360)-sequence in base 360, as well as their antithetic counterparts, give roughly the same accuracy as the antithetic Monte Carlo sampling. For simplicity, these error curves have therefore not been included in this graph. Superimposed are errors and lines for Sobol' and those for Sobol' with antithetic sampling. In this case the Sobol' sequence is seen to catch up with the antithetic random sampling, but little is gained by combining antithetic sampling with the quasi-random sequence. The Sobol' sequence outperforms Latin hypercube sampling on this, probably because the one-dimensional parts of it are no longer so dominant.

Figure 6 shows the errors from the Brownian bridge representation of the integrand. Reference lines are shown for Monte Carlo, antithetic sampling, and Latin hypercube sampling. In this case also, Latin hypercube sampling does a bit better after the Brownian bridge transformation, suggesting that the function has become somewhat more one-dimensional. As before, combining Latin hypercube sampling with antithetic sampling does not do much good.

The Sobol' sequences perform especially well on the nonlinear function, in the Brownian bridge representation with antithetic sampling. In terms of equation (3.4), this may be due to good equidistribution among the leading Sobol dimensions, matched with their greater importance to the integrand.

7. CONCLUSIONS

Our main conclusions are as follows.

- Quasi-Monte Carlo methods provide significant improvements in accuracy and computational speed for problems of small to moderate dimension.
- While the effectiveness of quasi-Monte Carlo can be lost on problems of high dimension, this does not happen if the integrand is of low effective dimension in the superposition sense.
- Some problems that have a large nominal dimension can be reformulated to have a moderate-sized effective dimension, so that the effectiveness of quasi-Monte Carlo is increased.

- The Brownian bridge representation reduces the effective dimension for problems like the mortgage-backed security problem described here.

Instead of straightforward use of high-dimensional quasi-random sequences, our recommendations are:

- First analyze the problem, mathematically or numerically, to determine the most important input dimensions.
- Where possible, reformulate the problem to concentrate the variation in fewer dimensions.
- When a small number of dominant dimensions can be identified or induced, apply quasi-random or randomized quasi-random (when sample-based error estimates are desired) sequences to those dimensions.
- For the remaining dimensions use pseudo-random or Latin hypercube sampling.
- Consider applying classical variance reduction techniques, such as antithetic sampling, control variates, stratification, and importance sampling in conjunction with the above.

We believe that high-dimensional integration problems can range in difficulty from completely intractable to quite simple. In some cases it is possible to turn the former into the latter by carefully engineering the integrand. It is too early to say whether such manageable integrands are rare or dominant in financial applications. However, the results here indicate that, for the valuation of securities which depend on a single stochastic factor modelled as a Gaussian process with nonstochastic drift and volatility, the Brownian bridge representation may be extremely effective in reducing the dimension of the simulation.

Acknowledgements

We are grateful to Spassimir Paskov and Joseph Traub for a number of discussions and for providing us with their quasi-random number generator FINDER. Partial support for this work came from the following sources: NSF (Owen), an NSF Postdoctoral Fellowship (Morokoff), ARO grant no. DAAHO4-95-1-0155 (Caflisch and Morokoff).

REFERENCES

1. R. E. Caflisch and W. Morokoff. Quasi-Monte Carlo computation of a finance problem. In: *Proc. Workshop on Quasi-Monte Carlo Methods and Their Applications* (K. T. Fang and F. J. Hickernell, eds.), pp. 15–30, and UCLA CAM Report 96-16, 1996.

2. R. Cranley and T. N. L. Patterson. Randomization of number theoretic methods for multiple integration. *SIAM Journal of Numerical Analysis*, **13** (1976), 904–914.

3. F. J. Hickernell. The mean square discrepancy of randomized nets. Technical Report MATH-112, Department of Mathematics, Hong Kong Baptist University, 1996.

4. F. J. Hickernell. Quadrature error bounds and figures of merit for quasi-random points. Technical Report MATH-111, Department of Mathematics, Hong Kong Baptist University, 1996.

5. M. D. McKay, R. J. Beckman, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, **21**(2) (1979), 239–245.

6. W. Morokoff and R. E. Caflisch. A Quasi-Monte Carlo approach to particle simulation of the heat equation. *SIAM Journal of Numerical Analysis*, **30** (1993), 1558–1573.
7. W. Morokoff and R. E. Caflisch. Quasi-random sequences and their discrepancies. *SIAM J. Sci. Stat. Computing*, **15** (1994), 1251–1279.
8. W. Morokoff and R. E. Caflisch. Quasi-Monte Carlo integration. *J. Comp. Phys.*, **122** (1995), 218–230.
9. B. Moskowitz and R. E. Caflisch. Smoothness and dimension reduction in quasi-Monte Carlo methods. *J. Math. Comp. Modeling*, **23** (1996), 37–54.
10. H. Niederreiter. *Random Number Generation and Quasi-Monte Carlo Methods*. SIAM, Philadelphia, 1992.
11. S. Ninomiya and S. Tezuka. Toward real-time pricing of complex financial derivatives. *Appl. Math. Finance*, **3** (1996), 1–20.
12. A. B. Owen. Orthogonal arrays for computer experiments, integration and visualization. *Statistica Sinica*, **2** (1992), 439–452.
13. A. B. Owen. Lattice sampling revisited: Monte Carlo variance of means over randomized orthogonal arrays. *Annals of Statistics*, **22** (1994), 930–945.
14. A. B. Owen. Randomly permuted (t, m, s)-nets and (t, s)-sequences. In: *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing* (H. Niederreiter and P. Jau-Shyong Shiue, eds.), pp. 299–317. Springer, New York, 1995.
15. A. B. Owen. Monte Carlo variance of scrambled equidistribution quadrature. *SIAM Journal of Numerical Analysis*, **34** (1997), 1884–1910.
16. A. B. Owen. Scrambled net variance for integrals of smooth functions. *Annals of Statistics*, **25** (1997), to appear.
17. S. H. Paskov. New methodolgies for valuing derivatives. In: *Mathematics of Derivative Securities* (S. Pliska and M. Dempster, eds.). Isaac Newton Institute/Cambridge University Press, 1996.
18. S. H. Paskov and J. F. Traub. Faster valuation of financial derivatives. *Journal of Portfolio Management* (1995), 113–120.
19. J. Spanier. Quasi-Monte Carlo methods for particle transport problems. In: *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing* (H. Niederreiter and P. Jau-Shyong Shiue, eds.), pp. 121–148. Springer, New York, 1995.
20. M. Stein. Large sample properties of simulations using latin hypercube sampling. *Technometrics*, **29**(2) (1987), 143–151.
21. D. F. Swayne, D. Cook, and A. Buja. XGobi: Interactive dynamic graphics in the X window system with a link to S. *ASA Proceedings of Statistical Graphics Section* (H. Niederreiter and P. Jau-Shyong Shiue, eds.), pp. 1–8, 1991.
22. A. Takemura. Tensor analysis of ANOVA decomposition. *Journal of the American Statistical Association*, **78** (1983), 894–900.
23. B. Tang. Orthogonal array-based latin hypercubes. *Journal of the American Statistical Association*, **88** (1993), 1392–1397.
24. B. Tuffin. On the use of low discrepancy sequences in Monte Carlo methods. Technical Report Publication Interne 1060, IRISA, 1996.