

UCLA
COMPUTATIONAL AND APPLIED MATHEMATICS

**Modified Monte Carlo Methods Using
Quasi-Random Sequences**

Russel E. Caflisch
Bradley Moskowitz

December 1994
CAM Report 94-41

Department of Mathematics
University of California, Los Angeles
Los Angeles, CA. 90024-1555

Modified Monte Carlo Methods Using Quasi-Random Sequences

Russel E. Caflisch *
Bradley Moskowit †
Mathematics Department
University of California
Los Angeles, CA 90095-1555.

Abstract

Computational experiments have shown that Monte Carlo methods using quasi-random sequences lose some of their effectiveness for integration problems in which the dimension is large or the integrand is not smooth. In this paper, two modified Monte Carlo methods are developed, which regain an enhanced convergence rate. The standard rejection method involves discontinuities, corresponding to the decision to accept or reject. In place of this, a smoothed rejection method is formulated and found to be very effective when used with quasi-random sequences. Monte Carlo evaluation of Feynman-Kac path integrals involves high dimension, one dimension for each discrete time interval. Through an alternative discretization, the effective dimension of the integration domain is drastically reduced, so that quasi-random sequences are effective.

1 Introduction

Quasi-random sequences are a deterministic alternative to random or pseudo-random sequences for use in Monte Carlo methods. Whereas Monte Carlo integration using random or pseudo-random sequences has error of size $O(N^{-1/2})$, the theoretical error bounds using quasi-random points are of size $O(N^{-1} \log^d N)$ in dimension d [8].

*caflisch@math.ucla.edu. Research of both authors supported in part by the Army Research Office under grant number DAAH04-95-1-0155.

†moskowit@bettis.gov. Current address: Westinghouse Bettis Atomic Power Lab, Reactor Methods and Analysis, P.O. Box 79, West Mifflin, PA 15122-0079

In practice, however, this improved performance for quasi-random sequences is often not observed. The loss of accuracy for integration problems was found by Morokoff and Caflisch [2, 3, 4] to be due to two causes: high dimension of the integration domain and lack of smoothness of the integrand. These results are reviewed in Section 2.

The purpose of this paper is to show how standard Monte Carlo methods can be modified to increase the effectiveness of quasi-random sequences, through reduction of effective dimension and through smoothing. Two examples will be presented. First, in Section 3 the effective dimension for the evaluation of a Feynman-Kac integral will be reduced by modification of the simulation procedure using the Brownian bridge. The second example, presented in Section 4, is a smoothed version of the acceptance-rejection method. Additional details on some of these problems are contained in [7]

Some conclusions and prospects for further work in this direction are discussed in the final section.

2 Quasi-Random Sequences, Dimension and Smoothness

The principal error bound for Monte Carlo integration using quasi-random sequences is the Koksma-Hlawka inequality, which states that for Monte Carlo integration of a function f on the unit cube in R^d using N points, the integration error $E_N(f)$ is bounded by

$$E_N(f) \leq D_N V(f) \tag{1}$$

in which D_N is the *discrepancy* of the sequence and $V(f)$ is the variation of f in the Hardy-Krause sense [8]. Moreover, for quasi-random sequences the discrepancy D_N is of size $O(N^{-1} \log^d N)$. Roughly speaking $V(f)$ is finite if the integrand is d times differentiable in d dimensions.

The dependence of quasi-Monte Carlo performance on dimension, is most directly seen in calculations of the discrepancy for quasi-random sequences [3]. At any dimension, the discrepancy behaves like $O(N^{-1} \log^d N)$ for sufficiently large N . On the other hand, for large values of d and moderate values of N , the discrepancy of a quasi-random sequence is seen to be almost exactly that of a random sequence.

The effect of smoothness on integration error is more difficult to assess. First, note that the variation, which appears in the Koksma-Hlawka inequality is infinite unless the integrand is very smooth. On the other hand, we expect that this bound is a conservative over estimate and that less smoothness will suffice. For example, Wozniakowski [12] (cf. [3] for a simplified proof) showed that for the Brownian sheet, which is only Hölder with exponent approximately 1/2, the average integration error is equal to the L^2 discrepancy. Note that this differs from the L^∞ discrepancy D_N and that the factor $V(f)$ in (1) is known to optimal even for some C^∞ functions [8]. Nevertheless, since $V(f)$ is infinite for almost all Brownian sheet functions, the result of [12] suggests that there could be improvements in the Koksma-Hlawka inequality.

On the other hand, computational experiments [4] show that for integrands that are discontinuous, the enhanced convergence rate of quasi-Monte Carlo integration is reduced.

3 Reduction of Effective Dimension for Feynman-Kac Integrals

In this section we show how the difficulties with high dimension can be overcome in a particular example. For a Feynman-Kac integral, the dimension is the number of time steps included in simulating the underlying Brownian motion. Although this may be quite large, a simple modification of the discretization of the Brownian path allows one to concentrate most of the variance into the first few dimensions, which significantly reduces the effective dimension and improves the accuracy of quasi-Monte Carlo integration.

The Feynman-Kac Formula provides a connection between linear parabolic differential equations and stochastic path integrals, which is similar to the method of characteristics for solving hyperbolic differential equations. For the initial value problem

$$\frac{\partial u}{\partial t} = \frac{1}{2} \frac{\partial^2 u}{\partial x^2}(x, t) + v(x, t) u(x, t) , \quad t \geq 0 \quad (2)$$

with initial condition $u(x, 0) = f(x)$, the solution is given by the Feynman-Kac formula as

$$u(x, t) = E_{x,0} \left[f(\xi(t)) e^{\int_0^t v(\xi(\tau), t-\tau) d\tau} \right] \quad (3)$$

where ξ represents a Wiener path starting at $\xi(0) = x$.

This solution can be approximated by discretizing time, generating Wiener process sample paths, and averaging over the set of sample paths to obtain Monte Carlo estimates of the expectation above. The integral in the exponent is itself approximated for each sample path using the Trapezoidal rule.

3.1 Standard Discretization

The **Standard Discretization** of the Wiener process is defined as follows:

1. Choose (x, t) at which the solution is to be approximated.
2. Choose the number of equal time steps m and set $\Delta t = \frac{t}{m}$ and $t_i = i \Delta t$ for $i = 0, \dots, m$.
3. Generate Gaussian independent variables, $\Delta W^{(i)}$ from the distribution $\mathcal{N}(0, \Delta t)$ (the normal distribution with mean 0 and variance Δt), for $i = 1, \dots, m$.
4. The Wiener path is sampled exactly at each t_i as the following sum:

$$\xi^{(i)} = x + \sum_{j=1}^i \Delta W^{(j)} \quad (4)$$

The sample paths are exact at the discrete times t_i , but an approximation is involved when the path integral of v is estimated. For example, using the Trapezoidal rule the Wiener path integral is approximated as

$$\int_0^t v(\xi(r), t-r) dr \approx \frac{\Delta t}{2} \left[v(x, t) + 2 \sum_{i=1}^{m-1} v(\xi^{(i)}, t-t_i) + v(\xi^{(m)}, 0) \right] = \tilde{I} \quad (5)$$

The nondifferentiability of the paths leads to an error (bias) of order $O(\sqrt{\Delta t})$ for this quadrature rule. In the computations presented below, however, Δt is sufficiently small so that discretization errors are negligible in comparison to statistical errors.

Using the sample paths above, the Monte Carlo estimate of the Feynman-Kac path integral is

$$\hat{u}_N(x, t) = \frac{1}{N} \sum_{k=1}^N f(\xi_k^{(m)}) e^{\tilde{I}_k} \quad (6)$$

in which \tilde{I}_k and $\xi_k^{(m)}$ for $i \leq k \leq N$ are N realizations of \tilde{I} and $\xi^{(m)}$. If the $\xi_k^{(m)}$ are normally distributed, as described above, then

$$E(\hat{u}_N(x, t)) = u(x, t) + \text{bias error}$$

in which the bias error comes from the quadrature rules and is insignificant.

For application of quasi-random Monte Carlo, this average is expressed as a multidimensional integral. First, each normal variable $\Delta W^{(j)}$ can be represented as a transformation G of a uniformly distributed variable x_j by $\Delta W^{(j)} = G(x_j)$. Then define

$$\begin{aligned} \tilde{f}(x_1, x_2, \dots, x_m) &= f(\xi^{(m)}) \\ \tilde{I}(x_1, x_2, \dots, x_m) &= \tilde{I} \end{aligned}$$

in which $\xi^{(i)} = x + \sum_{j=1}^i G(x_j)$, and x_j are uniformly distributed, e.g. pseudorandom or quasi-random points. The expectation of the Monte Carlo estimate in (6) can then be written as the following integral:

$$E(\hat{u}_N(x, t)) = \int_0^1 \dots \int_0^1 \tilde{f}(x_1, \dots, x_m) e^{\tilde{I}(x_1, \dots, x_m)} dx_1 \dots dx_m$$

The Monte Carlo estimate of this integral is

$$\tilde{u}_N(x, t) = \frac{1}{N} \sum_{k=1}^N \tilde{f}(x_1^k, \dots, x_m^k) e^{\tilde{I}(x_1^k, \dots, x_m^k)} \quad (7)$$

The transformation G used here is that of Marsaglia [1], which is continuous and monotonic. By contrast, Box-Muller, which is discontinuous, gives poorer results for quasi-random sequences [2].

3.2 Alternative Discretization Method

When the number of dimensions, m , is moderately large, an alternative discretization method leads to significant improvements in the quasi-random integration estimates by concentrating most of the variance into the lowest dimensions.

Assume that m is a power of two, and define the alternative discretization as follows:
Alternative Discretization:

1. Choose (x, t) at which the solution is to be approximated.
2. Choose the number of equal time steps $m = 2^p$. Then let $\Delta t = \frac{t}{m}$ and $t_i = i \Delta t$ for $i = 0, \dots, m$.
3. Generate Gaussian variables, $\Delta W^{(0)}$ distributed according to $\mathcal{N}(0, t)$ and $\Delta W^{(i)}$ distributed according to $\mathcal{N}(0, t/\alpha)$, for $i = 1, \dots, m$, in which $\alpha = 4 \cdot 2^{int(\log_2 i)}$.
4. The Wiener path $\xi^{(i)}$ is exactly sampled at each t_i as follows:

$$\begin{aligned}
 \xi^{(0)} &= x \\
 \xi^{(m)} &= \xi^{(0)} + \Delta W^{(0)} \\
 \xi^{(\frac{m}{2})} &= \frac{\xi^{(0)} + \xi^{(m)}}{2} + \Delta W^{(1)} \\
 \xi^{(\frac{m}{4})} &= \frac{\xi^{(0)} + \xi^{(\frac{m}{2})}}{2} + \Delta W^{(2)} \\
 \xi^{(\frac{3m}{4})} &= \frac{\xi^{(\frac{m}{2})} + \xi^{(m)}}{2} + \Delta W^{(3)} \\
 &\vdots
 \end{aligned} \tag{8}$$

For this discretization method, the first step is directly from 0 to t . Then the intermediate steps are filled in by taking successive subdivisions of the time intervals into halves. Each new intermediate path position is determined using the following rule:

$$\xi^{(j)} = \frac{\xi^{(j_1)} + \xi^{(j_2)}}{2} + \Delta W, \quad \Delta W \in \mathcal{N}(0, \frac{t_{j_2} - t_{j_1}}{4}) \tag{9}$$

where j_1, j_2 are the indices of the nearest prior and later time steps, respectively, for which the positions have already been determined. This representation of the Wiener path in terms of past and future positions is referred to as the Brownian bridge (also called a tied-down Wiener process).

The representation (8) can be conveniently rewritten as follows:

$$\begin{aligned}
 \xi^{(0)} &= x \\
 \xi^{(m)} &= x + \Delta W^{(0)} \\
 \xi^{(\frac{m}{2})} &= x + \frac{\Delta W^{(0)}}{2} + \Delta W^{(1)} \\
 \xi^{(\frac{m}{4})} &= x + \frac{\Delta W^{(0)}}{4} + \frac{\Delta W^{(1)}}{2} + \Delta W^{(2)} \\
 \xi^{(\frac{3m}{4})} &= x + \frac{3\Delta W^{(0)}}{4} + \frac{\Delta W^{(1)}}{2} + \Delta W^{(3)} \\
 &\vdots
 \end{aligned} \tag{10}$$

	Variance	
	Standard	Alternative
x_1	$1.0000t$	$3.1875t$
x_2	$0.8750t$	$1.6875t$
x_3	$0.7500t$	$.1875t$
x_4	$0.6250t$	$.1875t$
x_5	$0.5000t$	$.0625t$
x_6	$0.3750t$	$.0625t$
x_7	$0.2500t$	$.0625t$
x_8	$0.1250t$	$.0625t$

Table 1: Variances for Standard and Alternative Discretization Schemes, $m = 8$

The variance contributed by each dimension to the representation of the Wiener sample paths is compared for the Standard and the Alternative Discretizations are presented in Table 1 for the case $m = 8$. Note that variance due to x_k slowly decreases for the Standard Discretization, since early values of the Wiener process affect more parts of the path than later values, but that this decays is much more rapid for the Alternative Discretization.

3.3 Computational Example for the Feynman-Kac Formula

This example demonstrates the accuracy of Monte Carlo estimates of the solution of a simple linear parabolic differential equation using (6) at selected values of x and a fixed (small) time t . A comparison will be made of pseudorandom and quasi-random sequences with either the Standard discretization (4), or the Alternate discretization (9).

Example 1 Consider the following linear parabolic differential equation:

$$\frac{\partial f}{\partial t}(x, t) = \frac{1}{2} \frac{\partial^2 f}{\partial x^2}(x, t) + \left(\frac{1}{t+1} + \frac{1}{x^2+1} - \frac{4x^2}{(x^2+1)^2} \right) \frac{\partial f}{\partial x}(x, t)$$

with initial condition $f(x, 0) = \frac{1}{x^2+1}$. The exact solution is solution: $f(x, t) = \frac{t+1}{x^2+1}$.

Estimates are computed at a fixed (small) time, T , and at eight equally spaced positions between $x = -3$ and $x = 3$ using a discretization of time into m equal steps and a sample size of N . The estimates are then compared with the exact solution using the following L^2 measure of error:

$$L2e = \sqrt{\frac{1}{8} \sum_{k=0}^7 [f(x_k, T) - \hat{f}_N(x_k, T)]^2}$$

where $x_k = -3 + 6k/7$.

T is selected to be very small so that discretization errors are insignificant relative to the Monte Carlo errors.

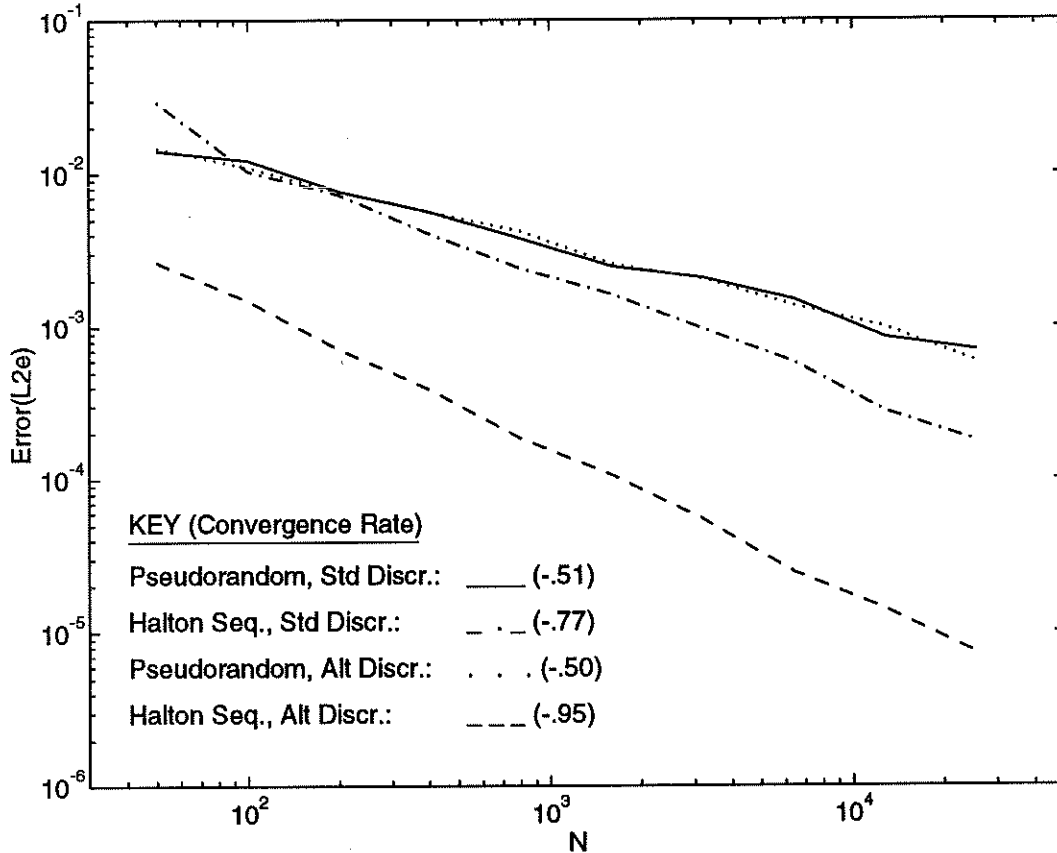


Figure 1: Log-Log Plot for Example 1, 75 runs, $T = 0.08$, $m = 32$.

A measure of the rate of decrease of the values of $L2e$ is obtained by a least squares fit of the function $cN^{-\alpha}$ to the calculated values of $L2e$. The resulting values of α for pseudorandom and quasi-random (Halton was used here) points used in both the standard and alternative discretizations are presented along with log-log plots of the data in Figures 1 for $T = .08$ ($m = 32$). The results show that the convergence rate for pseudorandom is independent of the discretization scheme. On the other hand, the performance of quasi-random Monte Carlo degrades in the standard discretization scheme, but is maintained at α nearly equal to 1 in the alternative discretization. It is important to notice that the quasi-random improvement from the alternate scheme grows as the dimension increases. This is expected, as is the fact that at 32 dimensions standard quasi-random is just barely better than pseudorandom for the values of N used here.

For standard Monte Carlo using pseudorandom sequences there is no difference between the two discretizations, since the total variance of the two methods is the same. On the other hand, the Alternative discretization method provides significantly more accurate results than the Standard discretization, when using quasi-random sequences in the Feynman-Kac formula. This improvement is due to reduction of the effective dimension of the problem.

4 Smoothing of the Rejection Method

The rejection method is one of the most commonly used sampling procedures in Monte Carlo methods. In its standard form it involves a decision to either accept or reject trial sample points. Because of the resulting discontinuity in the integrand, it is not well adapted for quasi-random sequences. In this section a smoothed rejection method, which overcomes this difficulty, will be formulated.

4.1 Importance Sampling Using the Rejection Method

One of the most common uses of the rejection method is for variance reduction through importance sampling, and this will be used here to measure the effectiveness of the rejection method. For the integral $A = \int_D f(\mathbf{x})d\mathbf{x}$, introduce an importance function h which mimics the behavior of f over D but is either integrable analytically or easily numerically integrated. The sampling procedure is then altered to generate points distributed according to h instead of points which are uniformly distributed. Then, instead of evaluating $f(\mathbf{x})$ at each sample point, one evaluates $\frac{f(\mathbf{x})}{h(\mathbf{x})}$, which provides an unbiased estimate of the integral, since

$$E_h \left(\frac{f}{h} \right) = \int_D \frac{f(\mathbf{x})}{h(\mathbf{x})} \cdot h(\mathbf{x}) d\mathbf{x} = \int_D f(\mathbf{x}) d\mathbf{x} = A$$

The Importance Sampled Monte Carlo estimate can be written as follows:

$$\tilde{A}_N = \frac{1}{N} \sum_{i=1}^N \frac{f(\mathbf{x}_i)}{h(\mathbf{x}_i)} \quad , \quad \mathbf{x}_i \sim h(\mathbf{x}) \quad (11)$$

where $\mathbf{x}_i \sim h(\mathbf{x})$ indicates that the sample points, $\mathbf{x}_1, \dots, \mathbf{x}_N$ have a probability density given by h (note that h should be normalized so that $\int_D h d\mathbf{x} = 1$). The variance of this estimate is substantially reduced if f/h has a smaller variance than f .

For a limited class of density functions h , the sample points may be generated by a transformation of the uniform distribution. For more general h , some other generation procedure must be used, such as the rejection method.

4.2 Standard Rejection Method

Here is the basic algorithm for the **Rejection Method** when the original sample is uniformly distributed in the d -dimensional unit cube. Note that one additional dimension is added for the extra variable y which acts as a “decision-maker”:

1. Select $\gamma \geq \sup_{\mathbf{x} \in D} h(\mathbf{x})$.

2. Repeat until N points have been accepted:

- (a) Sample (\mathbf{x}_t, y_t) from $U([0, 1]^{d+1})$.
- (b) If $y_t < \gamma^{-1} h(\mathbf{x}_t)$, accept trial point \mathbf{x}_t .
Otherwise, reject the trial point.

This algorithm produces a sequence of accepted points in d dimensions which are distributed according to h , as required for Importance Sampling.

The sum (11), in which the points \mathbf{x}_i are chosen using the rejection method, can be reinterpreted as a Monte Carlo evaluation of the following integral

$$A = \gamma \int_D \int_0^1 \frac{f(\mathbf{x})}{h(\mathbf{x})} \chi(y < \gamma^{-1} h(\mathbf{x})) dy d\mathbf{x}. \quad (12)$$

The direct Monte Carlo estimation of this integral is

$$\tilde{A}_N = \frac{\gamma}{N} \sum_{i=1}^N \chi \left\{ y_i < \frac{h(\mathbf{x}_i)}{\gamma} \right\} \cdot \frac{f(\mathbf{x}_i)}{h(\mathbf{x}_i)} \quad (13)$$

The variance of this integral is reduced by replacing the total number of points N by the number of accepted points to get the following Monte Carlo estimator

$$\tilde{A}_N = \frac{1}{\sum_{i=1}^N \chi \left\{ y_i < \frac{h(\mathbf{x}_i)}{\gamma} \right\}} \sum_{i=1}^N \chi \left\{ y_i < \frac{h(\mathbf{x}_i)}{\gamma} \right\} \cdot \frac{f(\mathbf{x}_i)}{h(\mathbf{x}_i)} \quad (14)$$

One difference between the sum (14) and a straightforward Monte Carlo evaluation of (12), is that in (14) the number N is taken to be the number of accepted points rather than the total number of trial points. This accounts for the factor γ in (13).

Quasi-Monte Carlo cannot be effectively applied to the integral (12) because the integrand contains a characteristic function, corresponding to the decision to accept or reject. This will be demonstrated in computational examples below.

4.3 Smoothed Rejection

The Rejection Method is discontinuous as a result of the inherently discontinuous nature of the binary decision: accept or reject. This can be eliminated by allowing sample points to have associated ‘‘acceptance’’ weights as described below. A **Smoothed Rejection** method that retains the flexibility and advantages of the Rejection method but does not involve a discontinuous integrand will now be formulated. First, replace the integral in (12) by an equivalent smooth integral, as in

$$A = \gamma \int_D \int_0^1 \frac{f(\mathbf{x})}{h(\mathbf{x})} \chi_\delta(y, \gamma^{-1} h(\mathbf{x})) dy d\mathbf{x}. \quad (15)$$

in which the smooth function χ_δ satisfies

$$\int_0^1 \chi_\delta(y, \gamma^{-1} h(\mathbf{x})) dy = \gamma^{-1} h(\mathbf{x}). \quad (16)$$

The function χ_δ , which will also be referred to as the weight w , will be chosen to be piecewise linear below.

The corresponding Monte Carlo approximation corresponds to the following Smoothed Rejection procedure:

1. Select $\gamma \geq \sup_{\mathbf{x} \in D} h(\mathbf{x})$ and $0 < \delta \ll 1$.
2. Repeat until weight of accepted points is within one unit of N :
 - (a) Sample (\mathbf{x}_t, y_t) from $U([0, 1]^{d+1})$.
 - (b) If $y_t < \frac{h(\mathbf{x}_t)}{\gamma} - \frac{1}{2}\delta$, then acceptance weight is $w = 1$.
Else if $y_t > \frac{h(\mathbf{x}_t)}{\gamma} + \frac{1}{2}\delta$, then $w = 0$.
Else $w = \frac{1}{\delta} \left(\frac{h(\mathbf{x}_t)}{\gamma} + \frac{1}{2}\delta - y_t \right)$.

The density of accepted points \mathbf{x} is $f_{\text{accept}}(\mathbf{x})$, given by (with $w = \chi_\delta$)

$$\begin{aligned} f_{\text{accept}}(\mathbf{x}) &= \frac{1 \cdot \int_0^1 w(\mathbf{x}, y) dy}{\int_{I^d} [1 \cdot \int_0^1 w(\boldsymbol{\xi}, y) dy] d\boldsymbol{\xi}} \\ &= \frac{h(\mathbf{x})/\gamma}{1/\gamma} \\ f_{\text{accept}}(\mathbf{x}) &= h(\mathbf{x}) \end{aligned}$$

which shows that the density function h is correctly sampled.

There is some extra work required by Smoothed Rejection as compared to ordinary Rejection. First, there is the work associated with assigning each sample point a weight, and storing and using these weights. This is typically small enough to be considered insignificant. Second, there is additional work which comes from the acceptance of points with weights of less than 1; i.e., in order to reach a total acceptance weight of size N , more than N evaluations of f are required. This extra work can be minimized by setting the constant δ sufficiently small. On the other hand, if δ is made too small, the advantages of continuity will be effectively lost.

4.4 Weighted Uniform Sampling

Another alternative to the Rejection method can be formulated by eliminating the accept/reject decision entirely, and instead assigning each sample point a weight equal to its acceptance probability given by $\gamma^{-1}h(\mathbf{x}_i)$.

A new Monte Carlo estimate is obtained as follows:

$$\begin{aligned} \check{A}_N &= \frac{\sum_{i=1}^N \frac{f(\mathbf{x}_i)}{h(\mathbf{x}_i)} \cdot \frac{h(\mathbf{x}_i)}{\gamma}}{\sum_{i=1}^N \frac{h(\mathbf{x}_i)}{\gamma}} \\ &= \frac{\sum_{i=1}^N f(\mathbf{x}_i)}{\sum_{i=1}^N h(\mathbf{x}_i)} \end{aligned}$$

This sum is essentially the ratio of two Crude Monte Carlo integration estimates – an estimate of the original function, f , in the numerator, and an estimate of the importance function, h , in the denominator. The resulting estimate is biased. Nevertheless, Powell and Swann [10] and Spanier and Maize [11] have shown that this bias is negligible in comparison with the rms-error as $N \rightarrow \infty$. Moreover the positive correlation between f and h , when h has been well chosen to closely mimic the behavior of f , provides substantial variance reduction.

The bias and rms-error are given as follows:

$$\text{bias}(\check{A}_N) = \frac{A \text{var}(h)}{N} - \frac{\text{cov}(f, g)}{N} + O(N^{-3/2}) \quad (17)$$

$$\text{rmse}(\check{A}_N) = \frac{\sqrt{\text{var}(f) + A^2 \text{var}(h) - 2A \text{cov}(f, h)}}{\sqrt{N}} + O(N^{-3/4}) \quad (18)$$

One advantage of Weighted Uniform Sampling, as emphasized in [10], is that sample points need not be generated for the density h . A second advantage is that the weighted uniform sampling estimate is continuous (assuming that f and h are continuous) so that quasi-random sequences may be effectively used in the sum. This will be demonstrated in computational examples below.

On the other hand, a disadvantage of Weighted Uniform Sampling is that for problems with large regions of low importance many more function evaluations will be performed within such regions than if Importance Sampling were used instead. This is reflected in a greater amount of variance reduction in the latter case, and for such problems Smoothed Rejection may be preferable to Weighted Uniform Sampling.

4.5 Computational Example

The Rejection, Smoothed Rejection and Weighted Uniform Sampling methods will now be compared an example. Consider the following integral:

$$A = \int_{I^d} f(\mathbf{x}) d\mathbf{x}$$

where I^d is the unit cube in d dimensions, and evaluate the following four estimates of this integral:

$$\begin{aligned} \text{Crude: } \hat{A}_N^{(1)} &= \frac{1}{N} \sum_{i=1}^N f(\mathbf{x}_i), \quad \mathbf{x}_i \sim U(I^d) \\ \text{W.U.S.: } \hat{A}_N^{(2)} &= \frac{\sum_{i=1}^N f(\mathbf{x}_i)}{\sum_{i=1}^N h(\mathbf{x}_i)}, \quad \mathbf{x}_i \sim U(I^d) \\ \text{Rej Meth: } \hat{A}_N^{(3)} &= \frac{1}{N} \sum_{i=1}^N \frac{f(\mathbf{x}_i)}{h(\mathbf{x}_i)}, \quad \mathbf{x}_i \sim h(\mathbf{x}), \text{ accepted point.} \\ \text{Smooth Rej: } \hat{A}_N^{(4)} &= \frac{1}{N} \sum_{i=1}^{N^*} w_i \frac{f(\mathbf{x}_i)}{h(\mathbf{x}_i)}, \quad (\mathbf{x}_i, w_i) \sim h(\mathbf{x}) \end{aligned}$$

For the last estimate, (\mathbf{x}_i, w_i) represents a weighted sample point with acceptance weight w_i , and N^* is chosen such that the sum of the acceptance weights is within one unit of N .

For a given value of N , take M samples of each of these estimates, denoted by $\hat{A}_{N,k}^{(j)}$ for $1 \leq k \leq M$. The average integration error will be measured through the empirical root mean square error, defined as

$$rmse(\hat{A}_N^{(j)}) \approx \sqrt{\frac{1}{M} \sum_{k=1}^M (\hat{A}_{N,k}^{(j)} - A)^2}, \quad j = 1, \dots, 4 \quad (19)$$

Note that the formula for *rmse* error uses the exact integral A , which is known for the example below.

Since successive estimates $\hat{A}_{N,k}^{(j)}$, $k = 1, \dots, M$ are not independent for quasi-Monte Carlo, there is no theoretical basis for these error expressions. This is in contrast to standard Monte Carlo for which the Central Limit Theorem implies that the $\hat{A}_{N,k}^{(j)}$ come approximately from a Gaussian distribution with variance given by the *rmse*.

Finally, the bias of Weighted Uniform Sampling proved to be insignificant, as expected, when using either pseudorandom or quasi-random sequences of points.

Example 2 Consider Monte Carlo integration over $I^7 = [0, 1]^7$ of the function

$$f_3(x_1, \dots, x_7) = e^{1 - (\sin^2(\frac{\pi}{2}x_1) + \sin^2(\frac{\pi}{2}x_2) + \sin^2(\frac{\pi}{2}x_3))} \arcsin\left(\sin(1) + \frac{x_1 + \dots + x_7}{200}\right)$$

using the positive definite importance function:

$$h_3(x_1, \dots, x_7) = \frac{1}{\eta} e^{1 - (\sin^2(\frac{\pi}{2}x_1) + \sin^2(\frac{\pi}{2}x_2) + \sin^2(\frac{\pi}{2}x_3))}$$

where η is

$$\eta = \int_{I^7} h_3(x_1, \dots, x_7) dx_1 \dots dx_7 = e \cdot \left(\int_0^1 e^{-\sin^2(\frac{\pi}{2}x)} dx \right)^3$$

which is easily approximated to high accuracy as a one-dimensional integral.

The resulting *rmse* errors for Example 2 using pseudorandom and quasi-random points are presented in Figures 2 and 3, respectively.

These computational examples show that quasi-Monte Carlo works well for the Smoothed Rejection Method, but even better for Weighted Uniform Sampling.

The results confirm that Smoothed Rejection is an improvement over Ordinary Rejection when using quasi-random sequences. Furthermore, Weighted Uniform Sampling produces results that are far superior to those obtained using Importance Sampling with the Rejection Method, even with smoothing. We believe the reason for this to be that although the Smoothed Rejection method involves a continuous integrand, there is still a relatively sharp slope in acceptance weights in the transition from acceptance to rejection regions. This leads to higher variation and poorer quasi-random performance, in general, than for Weighted Uniform Sampling, where the sharp transition is completely eliminated. Another, admittedly minor, advantage to Weighted Uniform Sampling is that the extra ‘decision’ variable y is no longer needed, reducing the effective dimension by one.

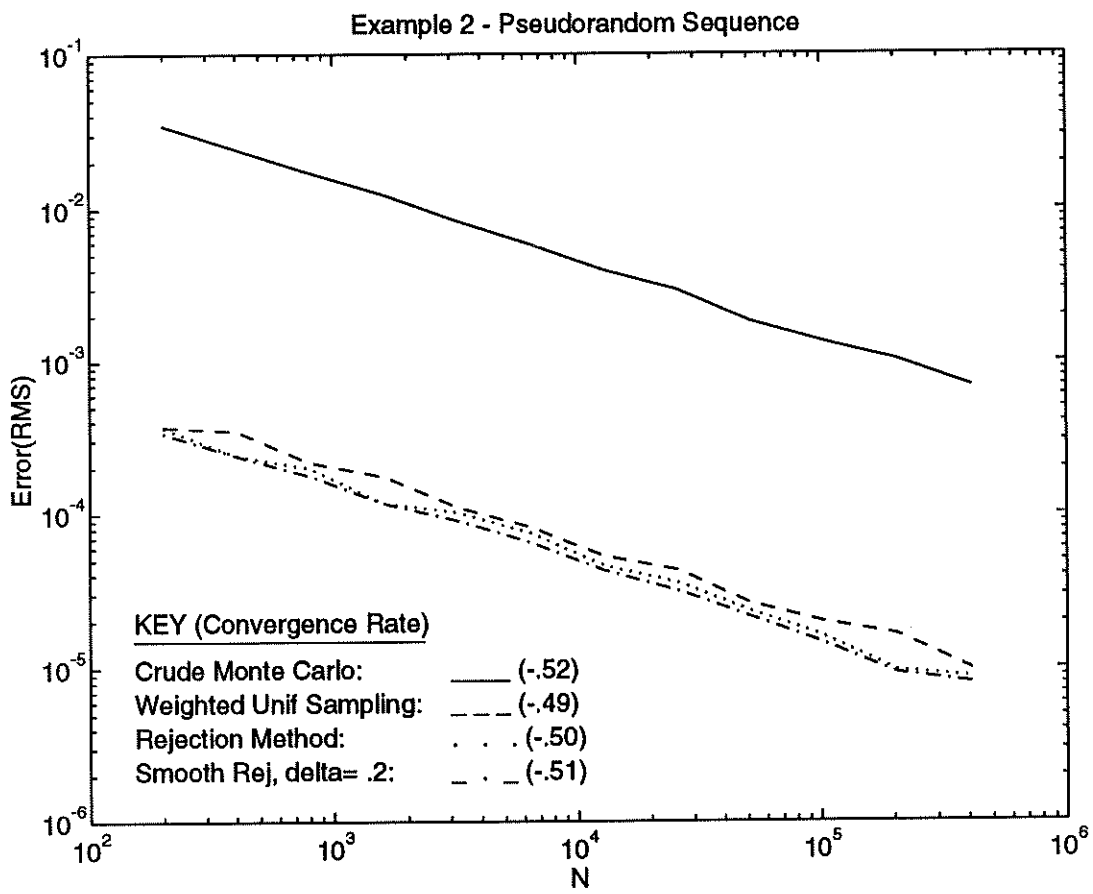


Figure 2: Pseudorandom Results, 75 runs, Example 2.

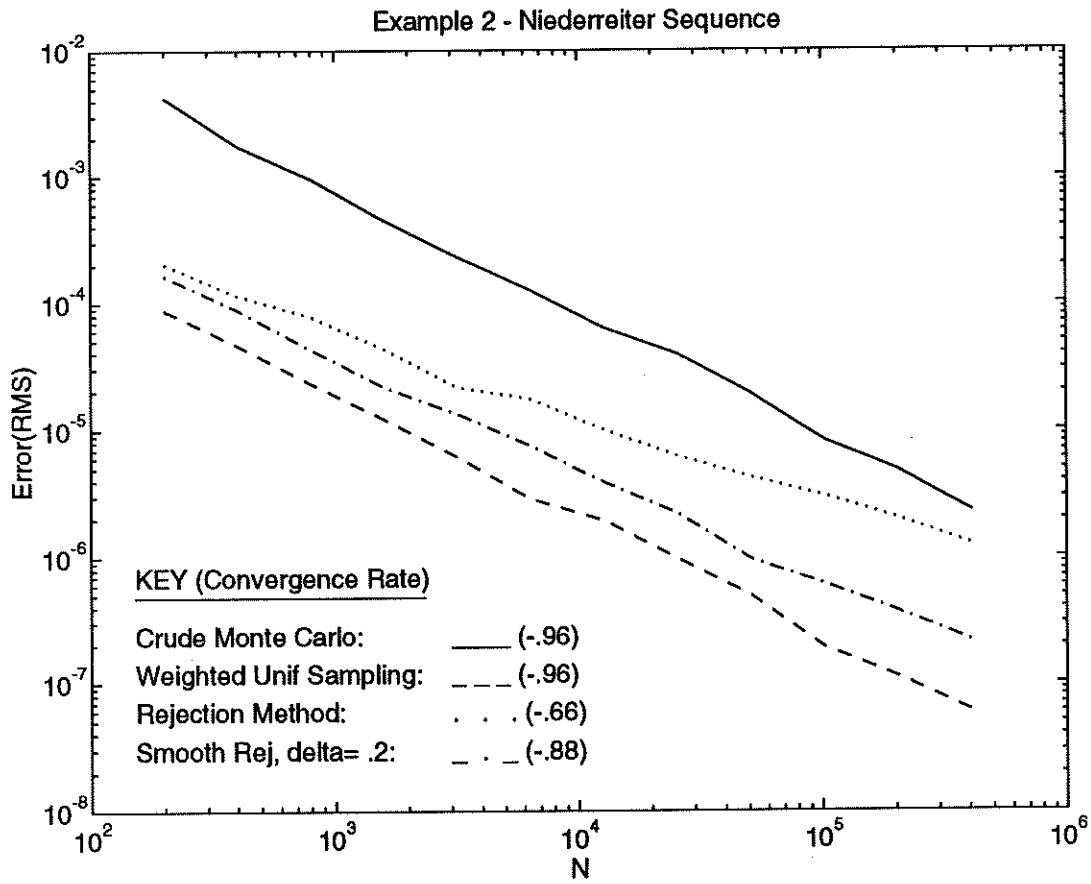


Figure 3: Quasi-Random Results, 75 runs, Example 2.

Nevertheless, Smoothed Rejection may be superior to Weighted Uniform Sampling for problems in which there are large regions of low importance.

5 Conclusions

We have presented two examples in which Monte Carlo methods could be modified to increase their effectiveness when applied with quasi-random sequences. Without these modifications, the performance of the methods with quasi-random sequences is no better than with pseudo-random sequences; after modification much of the theoretical improvement of quasi-random sequences is recovered.

There is plenty of room for further work in this direction: For example, we have not succeeded so far in developing a quasi-Monte Carlo version of the Metropolis algorithm for generation of the equilibrium of a stochastic process. We have also not tried to combine quasi-random sequences with other variance reduction techniques, such as stratification or antithetic variables.

There are several related works that deserve mention: Moskowitz [5, 6] has developed a transformation method for use of quasi-random sequences in diffusion Monte Carlo and in simulation of stochastic processes. He has applied this method to computation of the ground state energy of the hydrogen and helium atoms, as well as to other problems.

Paskov [9] has recently applied quasi-random variables to computation of integrals arising in financial derivatives. Although the integration domain has dimension 360 in his examples, the quasi-random result is significantly more accurate than the pseudo-random result.

References

- [1] G. Marsaglia. Normal (Gaussian) random variables for supercomputers. *The Journal of Supercomputing*, 5:49–55, 1991.
- [2] W. Morokoff and R.E. Caflisch. A Quasi-Monte Carlo approach to particle simulation of the heat equation. *SIAM Journal on Numerical Analysis*, 30:1558–1573, 1993.
- [3] W. Morokoff and R.E. Caflisch. Quasi-random sequences and their discrepancies. *SIAM J. Sci. Stat. Computing*, 15:1251–1279, 1994.
- [4] W. Morokoff and R.E. Caflisch. Quasi-Monte Carlo integration. *J. Comp. Phys.*, to appear, 1995.
- [5] B. Moskowitz. Improved stochastic simulation using quasi-Monte Carlo: a computational study. 1994.
- [6] B. Moskowitz. Quasirandom diffusion Monte Carlo. 1994.
- [7] B. Moskowitz and R.E. Caflisch. Smoothness and dimension reduction in quasi-Monte Carlo methods. *J. Math. Comp. Modeling*, submitted, 1994.

- [8] H. Niederreiter. *Random Number Generation and Quasi-Monte Carlo Methods*. SIAM, Philadelphia, 1992.
- [9] S.H. Paskov. Computing high dimensional integrals with applications to finance. *preprint*, 1994.
- [10] M.J.D. Powell and J. Swann. Weighted uniform sampling – a Monte Carlo technique for reducing variance. *J.Inst.Maths Applics*, 2:228–236, 1966.
- [11] J. Spanier and E.H. Maize. Quasi-random methods for estimating integrals using relatively small samples. *SIAM Review*, 36:18–44, 1994.
- [12] H. Wozniakowski. Average case complexity of multivariate integration. *Bulletin of the AMS*, 24:185–194, 1991.