

UCLA
COMPUTATIONAL AND APPLIED MATHEMATICS

**A Quasi-Monte Carlo Approach to Particle Simulation
of the Heat Equation**

William J. Morokoff
Russel E. Caflisch

July 1991

CAM Report 91-13

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

A Quasi-Monte Carlo Approach to Particle Simulation of the Heat Equation*

William J. Morokoff and Russel E. Caflisch†
Mathematics Department, UCLA

December 1, 1992

Abstract

The convergence of the Monte Carlo method for numerical integration can often be improved by replacing random numbers with more uniformly distributed numbers known as quasi-random. In this paper the convergence of Monte Carlo particle simulation is studied when these quasi-random sequences are used. For the one dimensional heat equation discretized in both space and time, convergence is proved for a quasi-random simulation using reordering of the particles according to their position. Experimental results are presented for the spatially continuous heat equation in one and two dimensions. The results indicate that a significant improvement in both magnitude of error and convergence rate can be achieved over standard Monte Carlo simulations for certain low-dimensional problems.

1 Introduction

1.1 Integration and Discrepancy

If $f(x_1, \dots, x_s)$ is a function of s variables over the s -dimensional unit cube I^s , then the integral of f over this domain may be approximated by averaging f

*AMS Classification: 65C05, 35K05, 82A70

†Research of both authors supported in part by the Air Force Office of Scientific Research under grant number AFOSR 90-0003.

at randomly selected points in the cube. This is the basic standard Monte Carlo method. The error ϵ is then given by

$$\epsilon = \int_{I^s} f(\mathbf{x})d\mathbf{x} - \frac{1}{N} \sum_{i=1}^N f(\mathbf{x}_i),$$

where $\{\mathbf{x}_i\}$ is a sequence of N random points in the cube. It can be shown that

$$E(\epsilon^2) = \frac{\sigma^2(f)}{N},$$

where $E(\cdot)$ is the expectation with respect to the random variables and $\sigma^2(f)$ is the variance of f . This leads to the well known results that the Monte Carlo method converges like $N^{-1/2}$ [6]. The random points used here are assumed to be independent and uniformly distributed over I^s . It is the uniformity of any set of N random points that makes this method work; for integration the independence is of secondary importance. Thus it seems plausible that if the random sequence is replaced by a more uniform sequence, smaller error may result. This is known as Quasi-Monte Carlo integration.

Initially it may appear that a grid would provide optimal uniformity. However, grids suffer from several difficulties. First, in high dimension, the number of points required to create even a course mesh is exponentially large as a function of dimension s . Also, grids have rather high discrepancy, a quantity which measures the uniformity of a set of points. This is defined and discussed below. Finally, the standard method for increasing accuracy of a grid is to halve the mesh size, which requires adding 2^s times the current number of points. Although it is desirable to be able to increase the number of points used without adding such an extremely large number, it is unclear how to place additional points on a grid to maintain uniformity unless the mesh is halved.

The solution to this problem is to use infinite sequences of points known as quasi-random or low-discrepancy sequences such that for every N , the first N terms of the sequence are uniformly distributed throughout the cube. In order to quantify this, the discrepancy of a set of N points is defined as follows. If Q is a rectangle contained in I^s , and $m(Q)$ is its volume, then the discrepancy D_N of the sequence $\{\mathbf{x}_i\}$ of N points is

$$D_N = \sup_{Q \in I^s} \left| \frac{\# \text{ of points in } Q}{N} - m(Q) \right|.$$

By the law of iterated logarithms, the discrepancy of a random sequence is bounded by $(\log \log N)N^{-1/2}$. There are many quasi-random sequences known for which the discrepancy is bounded by a constant times $(\log N)^s/N$, which suggests greater uniformity than a random sequence. Examples of such sequences are the Halton, Sobol' and Faure sequences, which have been analyzed in several papers [5],[17],[18],[3],[13],[14]. A common one dimensional sequence, which has structure similar to many quasi-random sequences, is the van der Corput sequence. This sequence can be written

$$\left\{ \frac{1}{2}, \frac{1}{4}, \frac{3}{4}, \frac{1}{8}, \frac{5}{8}, \frac{3}{8}, \frac{7}{8}, \frac{1}{16}, \frac{9}{16}, \frac{5}{16}, \frac{13}{16} \dots \right\}$$

The n^{th} term of the sequence is generated by taking the dot product of the vector $(1/2, 1/4, 1/8, 1/16, \dots)$ with the vector (a_1, a_2, a_3, \dots) , where the binary expansion of n is $a_m \dots a_2 a_1$, and all a_k with $k > m$ are set to zero so that the dot product is a finite sum. Note that as n increases and fluctuates between being odd or even, a_1 fluctuates from 0 to 1, so that the terms of the sequence switch between being less than $1/2$ and greater than $1/2$. The computational cost of generating this and other quasi-random sequences is not significantly greater than the cost for pseudo-random sequences.

The discrepancy of a sequence can be related to integration error as an upper bound through the Koksma-Hlawka inequality; it is also possible to formulate an L_2 version of discrepancy, which has recently been shown to be equivalent to average case integration error for continuous functions with respect to a Wiener sheet measure [19]. In [9] we present a discussion of bounds for discrepancy and compare computations of discrepancy with theoretical results. We have also performed a computational study of the convergence properties of quasi-random sequences for moderate values of N , the results of which appear in [10]. A number of other papers also discuss the properties of Quasi-Monte Carlo integration [11],[12], [2],[4],[8],[16],[15].

1.2 Simulation

Monte Carlo methods are frequently used to solve integral and differential equations. As with simple integration, the goal is to sample points randomly from a distribution function, which are then used as integration nodes. However, in the case of integral and differential equations, the distribution function is not known explicitly, but only as the solution to the equation. One method of performing this sampling is particle simulation. Here "particles" are sampled from some known

initial or source distribution, and then are allowed to move and interact in phase space according to the dynamics described in the equation. The particles at some later time or position are then the sampled points from the distribution function solution of the equation. An example of this is the Direct Simulation Monte Carlo method used to solve the Boltzmann equation [1]. In this method the particles represent gas molecules, and the distribution function is density in physical and velocity space. The particles evolve by convection and collisions; random numbers are used to determine which particles collide and the results of the collision.

It may be hoped that the improvement achieved by using quasi-random sequences in place of random numbers in integration problems can also be attained in particle simulations. This presents a more difficult problem of how to effectively apply these sequences and preserve convergence. In particle simulations the independence of the random numbers again becomes important, particularly in non-linear or time dependent problems where particles interact or are manipulated over several time steps. If care is not taken in how the quasi-random sequences are used, the correlations may destroy the method. In [7] Lécot studies this problem for the spatially homogeneous Boltzmann equation and provides a convergence proof when quasi-random sequences are used. However, the proof suggests convergence slower than the random methods, and numerical results are inconclusive as to whether any improvement has been achieved.

In section 2 of this paper a simpler problem is considered, that of a particle simulation of the one dimensional, space and time discrete heat equation. A convergence proof is given for quasi-random numbers, which sheds light on the nature of using low discrepancy sequences in simulations. Numerical results verify the predicted convergence. In section 3 the method is extended to the spatially continuous case in both one and two dimensions. Conclusions are stated in the final section.

2 Discrete One Dimensional (Quasi)-Random Walk

The one dimensional heat equation with periodic boundary conditions

$$\begin{aligned} \frac{\partial u}{\partial t} &= \frac{\partial^2 u}{\partial x^2} \\ u(x, 0) &= u_0(x) \\ u(x+1, t) &= u(x, t) \end{aligned}$$

can be approximated by a set of difference equations obtained by replacing the spatial derivative with a centered difference and the time derivative with a forward difference. If $\Delta x = 1/M$ is the spatial grid spacing, Δt is the time step, and u_i^n is the approximation to $u(i\Delta x, n\Delta t)$, then these equations can be expressed as

$$\begin{aligned} u_i^{n+1} &= (1 - 2\lambda)u_i^n + \lambda(u_{i+1}^n + u_{i-1}^n) \\ u_i^0 &= u_0(i\Delta x) \\ u_M^n &= u_0^n \\ u_{M-1}^n &= u_{-1}^n \\ u_{M+1}^n &= u_1^n \end{aligned}$$

Here $\lambda = \Delta t/\Delta x^2$, and by the CFL condition, $\lambda \leq 1/2$. The simulation was constructed to approximate the solution to these difference equations, and, in the error analysis, the convergence is based on the exact solution of the difference equations, and not on the heat equation itself. The initial data is chosen so that $\sum_{i=1}^M u_i^0 = 1$. It is easy to show that $\sum_{i=1}^M u_i^{n+1} = \sum_{i=1}^M u_i^n$, so that the total amount of “heat” in the system is conserved. For the particle simulation, this means that no particles enter or leave the domain. The approximation of the continuous heat equation by this discretized version is a well-studied subject in numerical analysis and will not be addressed further here.

The simulation consisted of using N equally weighted “particles of heat”. Initially $[Nu_i^0]$ particles are placed at each grid point x_i . Here $[z]$ is the nearest integer to z . The approximation to the solution at time Δt is then obtained by moving each particle according to the dynamics described in the difference equation. With probability $1 - 2\lambda$, a particle at x_i remains at the same position. With probability λ , a particle at x_i moves one grid space to x_{i-1} ; and with probability λ , a particle at x_i moves one grid space to x_{i+1} . Thus each particle random walks around the grid, moving at most one grid space per time step. After each particle has had a chance to move once, the approximation to u_i^1 is given by counting the number of particles at x_i and dividing by N . The particles then move again, and the solution at time $2\Delta t$ is calculated. The simulation continues on in this manner. It should be remarked that the CFL condition on λ prohibits the probability of staying at the same position from becoming negative.

In standard Monte Carlo, the random walk described above would be performed by choosing a uniform random number z between 0 and 1. If $2\lambda < z \leq 1$, the particle stays put; if $0 \leq z \leq \lambda$, the particle moves left; and if $\lambda < z \leq 2\lambda$, the particle moves right. The application of quasi-random sequences to this problem is

not quite so simple. One possibility is to assign a different one dimensional quasi-random sequence (of length n) to each particle. Without some kind of random scrambling, this will cause the particle to just repeat the same pattern over and over, and thus the method will not converge. Moreover, the number of sequences required (and thus the effective dimension of the problem) is N , which among other things is quite impractical. Another possibility is to use only a single one dimensional sequence (of length N) for all particles at a given time step. The first particle would be given the first term of the sequence, the second particle the second term, and so forth. At the next time step, a different one dimensional sequence could be used. This simulation can be described by an integral over the n dimensional cube, where n is the number of time steps. The difficulty here is that as Δt decreases or the total time of the calculation increases, the dimension of the quasi-random sequence grows, and the problems of high dimensionality arise. Thus this approach is not practical.

These difficulties may be avoided by using the a single one-dimensional sequence (of length nN) for all time steps. However, this cannot be implemented in a straightforward fashion, because of correlations. If $\lambda = 1/2$, the number of particles N is even, and the van der Corput sequence is used, then if what is originally labeled the first particle is always assigned the first term of the next N terms of the sequence, this particle will always move left. The second particle will always move right, and so on. This is an extreme case of what may happen, but even if the correlations are not as strong, convergence cannot be guaranteed. If the particles are relabeled, or reordered, according to their position at each time step, this difficulty disappears. For the one-dimensional heat equation, if $x(j)$ is the spatial position of the j^{th} particle, the reordering means relabeling the particles so that $x_1 \leq x(1) \leq x(2) \cdots \leq x(N) \leq x_M$. This has the effect of breaking correlations, and allows the same sequence to be used at the next time step.

The idea of reordering is motivated by a paper by Christian Lécot [7] in which a convergence proof for a simulation of the spatially homogeneous Boltzmann equation is described. In this paper it is the magnitude of the velocities of the particles which is used to reorder them. The proof given below for the convergence of this method for the heat equation follows the general outline of the Lécot's proof; however, in this simpler setting, the necessity of reordering and its effects on convergence become clearer. Tables 1 and 2 summarize the notation already introduced and define several other quantities used in the convergence proof.

The proof also makes use of a Hammersley sequence. This is an s dimensional finite sequence of length N , $s \geq 2$, formed by pairing N terms of an

$s - 1$ dimensional sequence in the unit cube with the one dimensional sequence $\{1/2N, 3/2N, \dots, (2n - 1)/2N\}$. It is possible to show that the addition of this one dimensional sequence does not change the discrepancy bound; that is, the $s - 1$ dimensional sequence and the s dimensional sequence have the same discrepancy bound. Thus one dimension is gained for free. The trade off is that this only works for a fixed N , which means that a Hammersley sequence of length N cannot be enlarged by adding on more points. For particle simulations this means that if a larger N is required, the simulation must be completely redone, and the results for the smaller N cannot be averaged in. However, this is frequently the procedure followed anyway, especially when there are particle interactions. The idea of the Hammersley sequence and the above notation are used to state the result in the following lemma.

Lemma 1 *Let $\{b\}$ be a one dimensional sequence in $[0, 1]$, such that the two dimensional Hammersley sequence formed from any contiguous subsequence of $\{b\}$ of length N has discrepancy bounded by D_N . If $\{b\}$ is used in the simulation of the discrete periodic one dimensional heat equation described above with particle reordering, then the weak measure of error at time step n , \mathcal{D}_N^n , satisfies*

$$\mathcal{D}_N^n \leq 5n D_N + \mathcal{D}_N^0,$$

where \mathcal{D}_N^0 is the error in representing the initial data by the particle approximation.

As is usual with particle method convergence estimates, convergence is shown here only in a weak sense. This fits well with the idea of the discrepancy of a sequence in that discrepancy measures how well a sequence integrates a class of test functions - namely, the rectangles. The first step of the proof is to sum the difference equations against a test function over the grid points. This establishes a weak formulation of the equation. The idea is then to define a weak measure of error which tells how well the particle distribution approximates the exact solution with reference to the test function at time t^{n+1} . This measure is related to the error at time t^n plus a term to account for the error introduced during the particle movement over one time step. The particle movement is shown to be a Monte Carlo evaluation of an integral over the unit square, and the error is bounded by the discrepancy of the quasi-random sequence.

The upper bound in Lemma 1 grows linearly with the number of time steps n , but this growth with t is not observed in computational experiments. Computational experiments also show that discrepancy is a good indicator of the error for

N	Number of Particles
M	Number of Grid Points
n	Number of Time Steps
j	Identifies Particle j : $1 \leq j \leq N$
i	Identifies Grid Point i : $1 \leq i \leq M$
x_i	$i\Delta x$: Coordinate of i^{th} Grid Point
X_j^n	Position of j^{th} Particle at Time $t^n = n\Delta t$
$u(x_i, t^n)$	Exact Solution of Difference Equations
$[u]_t(x_i, t^n)$	$(u(x_i, t^n + \Delta t) - u(x_i, t^n))/\Delta t$
λ	$\Delta t/(\Delta x)^2$
$\{\bar{z}_k\}$	A sequence in the unit square

Table 1: Basic Notation Used in Convergence Proof

$\phi(x)$	Piece-wise Continuous Bounded Periodic Function
$\phi_{s,r}(x)$	Characteristic Function of Interval $[s, r]$, $0 \leq s \leq r \leq 1$
$d_N^n(s, r)$	$(1/N) \sum_{j=1}^N \phi_{s,r}(X_j^n) - \sum_{i=1}^M \phi_{s,r}(x_i) u(x_i, t^n)$
D_N^n	$\sup_{\{(s,r): 0 \leq s \leq r \leq 1\}} d_N^n(s, r) $
$\Psi_j(c)$	Characteristic Function of $((j-1)/N, j/N]$
$\Theta(X_j^n, b)$	$\begin{cases} X_j^n - \Delta x, & 0 \leq b \leq \lambda \\ X_j^n + \Delta x, & \lambda < b \leq 2\lambda \\ X_j^n, & 2\lambda < b \leq 1 \end{cases}$
$K^{n+1}\phi(b, c)$	$\sum_{j=1}^N \phi(\Theta(X_j^n, b)) \Psi_j(c)$
$e_N^n(s, r)$	$\lambda \left[(1/N) \sum_{j=1}^N \phi_{s,r}(X_j^n + \Delta x) - \sum_{i=1}^M u(x_i, t^n) \phi_{s,r}(x_i + \Delta x) \right]$ $+ \lambda \left[(1/N) \sum_{j=1}^N \phi_{s,r}(X_j^n - \Delta x) - \sum_{i=1}^M u(x_i, t^n) \phi_{s,r}(x_i - \Delta x) \right]$
$\delta_N^n(s, r)$	$(1/N) \sum_{j=1}^N K^{n+1}\phi_{s,r}(\bar{z}_j) - \int_{I^2} K^{n+1}\phi_{s,r}(\bar{z}) d\bar{z}$

Table 2: Further Notation Used in Convergence Proof

this problem, although this has not been the case in other integration and transport problems [9, 10].

Proof: It is first necessary to construct a weak formulation of the equations, which can be expressed as

$$[u]_t(x_i, t) = \frac{1}{(\Delta x)^2} (u(x_{i+1}, t) - 2u(x_i, t) + u(x_{i-1}, t)).$$

Multiply by $\phi(x_i)$ and summing over the grid points gives

$$\begin{aligned} \sum_{i=1}^M \phi(x_i) [u]_t(x_i, t) &= \frac{1}{(\Delta x)^2} \sum_{i=1}^M \phi(x_i) (u(x_{i+1}, t) - 2u(x_i, t) + u(x_{i-1}, t)) \\ &= \frac{1}{(\Delta x)^2} \sum_{i=1}^M u(x_i, t) (\phi(x_{i+1}) - 2\phi(x_i) + \phi(x_{i-1})). \end{aligned}$$

The last equality follows from summation by parts and the periodicity of u and ϕ . From this it follows that

$$\sum_{i=1}^M \phi(x_i) u(x_i, t^{n+1}) = (1 - 2\lambda) \sum_{i=1}^M \phi(x_i) u(x_i, t^n) + \lambda \sum_{i=1}^M u(x_i, t^n) (\phi(x_{i+1}) + \phi(x_{i-1})).$$

This is the weak formulation of the equation.

Particle Approximation: The positions of the N particles at time t^n , X_j^n , are intended to approximate the solution of the equation, $u(x_i, t^n)$, in such a way that for all test functions $\phi(x)$,

$$\sum_{i=1}^M \phi(x_i) u(x_i, t^n) \approx \frac{1}{N} \sum_{j=1}^N \phi(X_j^n).$$

The quantity $d_N^n(s, r)$ is then defined to measure how good the approximation is for characteristic functions of intervals. The corresponding sup over all intervals is taken as the weak measure of error.

Error for One Time Step: The operator $K^{n+1}\phi(b, c)$ is defined to relate $\phi(X_j^{n+1})$ to $\phi(X_j^n)$. A sequence of N terms in I^2 is constructed by pairing the one dimensional sequence used to move X_j^n to X_j^{n+1} with the sequence $(1/2N, \dots, (2N - 1)/2N)$. This creates pairs $\bar{z}_j = (b_j, c_j)$ where c_j determines which particle, and b_j describes the motion of this particle. This allows the act of assigning each particle

a number to be interpreted as an evaluation of a function on I^2 . K^{n+1} is constructed so that $\int_{I^2} K^{n+1} \phi(b, c) dbdc$ gives the exact weak solution formulation for one time step, given that X_j^n is taken as the initial data. From the definition,

$$\int_{I^2} K^{n+1} \phi(\bar{z}) d\bar{z} = \frac{1-2\lambda}{N} \sum_{j=1}^N \phi(X_j^n) + \frac{\lambda}{N} \sum_{j=1}^N [\phi(X_j^n + \Delta x) + \phi(X_j^n - \Delta x)].$$

The actual particle movement carried out is then regarded as N functional evaluations of the integrand to give a Monte Carlo approximation to the integral. The quantity $\delta_N^n(s, r)$ is then the Monte Carlo error for this integration. Again from the definition of K^{n+1} it follows that

$$\frac{1}{N} \sum_{j=1}^N \phi(X_j^{n+1}) = \frac{1}{N} \sum_{j=1}^N K^{n+1} \phi(\bar{z}_j),$$

where $\{\bar{z}_j\}$ is the sequence in I^2 used as integration nodes.
Analysis:

$$\begin{aligned} d_N^{n+1}(s, r) &= \frac{1}{N} \sum_{j=1}^N \phi_{s,r}(X_j^{n+1}) - \sum_{i=1}^M \phi_{s,r}(x_i) u(x_i, t^{n+1}) \\ &= \frac{1}{N} \sum_{j=1}^N \phi_{s,r}(X_j^{n+1}) - \int_{I^2} K^{n+1} \phi_{s,r}(\bar{z}) d\bar{z} \\ &\quad + \int_{I^2} K^{n+1} \phi_{s,r}(\bar{z}) d\bar{z} - \sum_{i=1}^M \phi_{s,r}(x_i) u(x_i, t^{n+1}) \\ &= \frac{1}{N} \sum_{j=1}^N K^{n+1} \phi_{s,r}(\bar{z}_j) - \int_{I^2} K^{n+1} \phi_{s,r}(\bar{z}) d\bar{z} \\ &\quad + \frac{1-2\lambda}{N} \sum_{j=1}^N \phi_{s,r}(X_j^n) + \frac{\lambda}{N} \sum_{j=1}^N [\phi_{s,r}(X_j^n + \Delta x) + \phi_{s,r}(X_j^n - \Delta x)] \\ &\quad - \left[(1-2\lambda) \sum_{i=1}^M \phi_{s,r}(x_i) u(x_i, t^n) \right. \\ &\quad \left. + \lambda \sum_{i=1}^M u(x_i, t^n) [\phi_{s,r}(x_i + \Delta x) + \phi_{s,r}(x_i - \Delta x)] \right] \\ &= \delta_N^n(s, r) + (1-2\lambda) d_N^n(s, r) + e_N^n(s, r) \end{aligned}$$

The quantity $e_N^n(s, r)$ is composed of two parts related to movement left and movement right. For the part describing a particle moving to the grid point to the right, the relationship

$$\begin{aligned} \frac{1}{N} \sum_{j=1}^N \phi_{s,r}(X_j^n + \Delta x) - \sum_{i=1}^M \phi_{s,r}(x_i)u(x_i + \Delta x, t^n) = \\ \frac{1}{N} \sum_{j=1}^N \phi_{s',r'}(X_j^n) - \sum_{i=1}^M \phi_{s',r'}(x_i)u(x_i, t^n), \end{aligned}$$

where $s' = s + \Delta x$ and $r' = r + \Delta x$, holds true. But this last quantity is just $d_N^n(s', r')$. For particles moving to the left, the same relationship can be established, except with $s'' = s - \Delta x$ and $r'' = r - \Delta x$. Thus

$$e_N^n(s, r) = \lambda(d_N^n(s', r') + d_N^n(s'', r'')).$$

Bounding the right hand side by the sup over all intervals, it follows that

$$|e_N^n(s, r)| \leq 2\lambda \mathcal{D}_N^n.$$

The quantity $\delta_N^n(s, r)$ can be written, using the definition of K^{n+1} as

$$\delta_N^n(s, r) = \frac{1}{N} \sum_{k=1}^N \sum_{j=1}^N \phi_{s,r}(\Theta(X_j^n, b_k)) \Psi_j(c_k) - \int_{I^2} \sum_{j=1}^N \phi_{s,r}(\Theta(X_j^n, b)) \Psi_j(c) dbdc.$$

Here $\bar{z}_k = (b_k, c_k)$. Because $\phi_{s,r}$ and Ψ_j are both characteristic functions of intervals, and because the particles are reordered so that $X_j^n \leq X_{j+1}^n$ for all j , the integrand is the characteristic function of a region like the one shown in Figure 1, in which $\lambda = .25$. Since discrepancy is the maximum error over rectangles with sides parallel to the coordinate axes, it can be used to bound this error δ_N^n , for integration over a characteristic function, by breaking the region up into a union of such rectangles. The interval $[s, r]$ will contain a contiguous sequence of particles X_l^n to X_m^n . The particles at the left most grid point contained in $[s, r]$ can either stay put (corresponding to b being in the interval $[2\lambda, 1]$), or move to the right (b in the interval $[\lambda, 2\lambda]$) and still be in the interval $[s, r]$, and thus counted towards the integral. These particles are associated with region A2. Particles at the right most grid point contained in $[s, r]$ can either stay put or move to the left and still be in the interval. These particles are described by region A4. All particles between the two extreme grid points will be counted no matter how they move; this

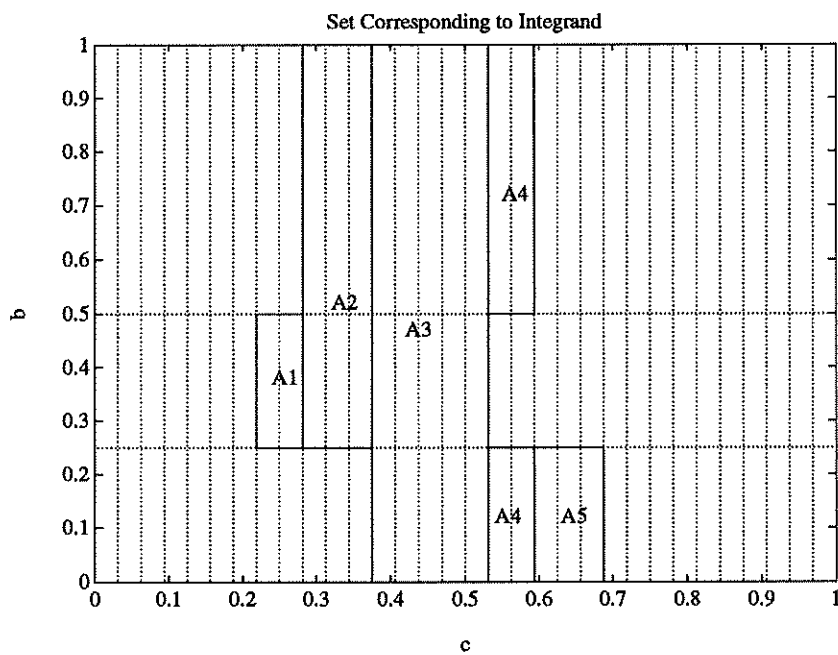


Figure 1: Typical Set Corresponding to $K^{n+1}\phi_{s,r}(b, c)$

is region $A3$. Region $A1$ refers to particles that are not in $[s, r]$, but will be if they move right, and region $A5$ is for the particles that must move left to end up in the interval.

The reordering of the particles is vital to this argument. Otherwise, Figure 1 could contain over N disjoint rectangles, as opposed to the three adjacent disjoint rectangles that now appear. Because of the possibility of wrapping around the ends due to periodicity, the maximum number of disjoint rectangles that will be formed, given that the particles have been reordered, is 5. It follows then that $\delta_N^n(s, r)$ is bounded by 5 times the error of integrating each rectangle separately. But this integration error is bounded by the discrepancy of the sequence. Thus

$$|\delta_N^n(s, r)| \leq 5D_N .$$

If these inequalities are applied to the equation derived for d_N^{n+1} , and the sup over both sides of the result is taken, the following inequality is obtained:

$$\mathcal{D}_N^{n+1} \leq 5 D_N + (1 - 2\lambda)\mathcal{D}_N^n + 2\lambda\mathcal{D}_N^n ,$$

or simply

$$\mathcal{D}_N^{n+1} \leq 5 D_N + \mathcal{D}_N^n .$$

The result given in the lemma then follows easily from repeated substitution.

If the van der Corput sequence is used to determine the particle motion, then $D_N \leq c(\log N)/N$, where c is a constant [11]. Thus convergence like $1/N$ is obtained. This has been verified computationally. Figure 2 shows the results of the simulation using a pseudo-random sequence and using the van der Corput sequence. The experiment consisted of using a grid spacing of $1/64$, $\lambda = .25$, and 40 values of N evenly spaced on a log scale between 1000 and 100,000. The initial data were taken to be a Gaussian distribution in x centered at $x = .5$ with variance $1/12$, discretized and normalized so that $\sum_{i=1}^M u(x_i, 0) = 1$. The corresponding correct percentage of the N particles was taken as the initial (deterministic) particle approximation leading to an initial error \mathcal{D}_N^0 proportional to M/N . The simulation was run up to 100 time steps, when the error was calculated. Averaging over runs to reduce scatter was found to be unnecessary, so only one run was performed. Convergence for the pseudo-random sequence is the expected $N^{-.5}$, while the quasi-random sequence achieves $1/N$. The error was calculated by averaging the square error at each grid point, and taking the square root. Though not displayed on the graph, the maximum error over all grid points was also calculated. This showed the same convergence as the square averaged error.

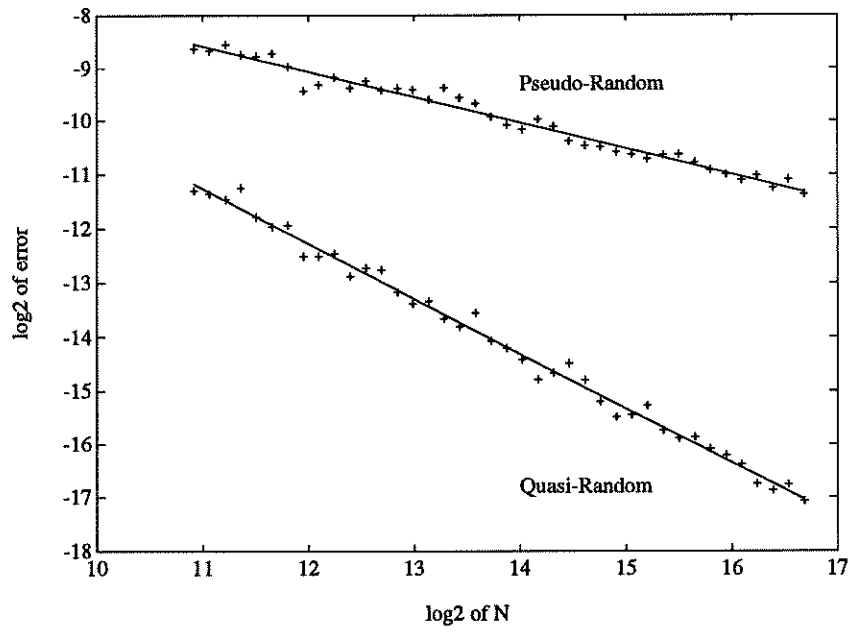


Figure 2: 1-D Discrete Periodic Heat Equation Simulation

The success of the quasi-random simulation is directly related to the fact that if there are many particles at each grid point, and the particles are reordered, then each grid point receives a large contiguous subsequence of the quasi-random sequence. Because this subsequence is very uniform over the interval $[0, 1]$, almost the exact percent of the particles will move right, left and stay put to match the difference equation, with an error of about $1/N$. If random numbers are used, this error rises to $N^{-.5}$. This suggests how quasi-random sequences may be applied in more complicated simulations. The next section discusses such possibilities, still within the context of the heat equation.

3 Gaussian Random Walk Simulation in One and Two Dimensions

A new level of complexity was added to the one dimensional heat equation simulation by removing the grid and allowing the particles to random walk to any point in the domain by taking steps sampled from a Gaussian distribution centered at the current particle position with variance related to the time discretization Δt . A quasi-random sequence was quite effective in the previous simulation because it mimicked the difference equation almost exactly once reordered. In the new simulation, the relationship between the sequence and the convergence is not as obvious. The new simulation also provides a better model problem for simulations like those used to solve the Boltzmann equation, where new particle velocities are sampled from continuous distributions related to the current velocity.

The new problem was to solve the heat equation for a one dimensional insulated rod described by the interval $[-1, 1]$, given an initial temperature distribution. The boundary conditions were

$$\frac{\partial u(-1, t)}{\partial x} = 0$$

and

$$\frac{\partial u(1, t)}{\partial x} = 0 .$$

The initial temperature profile was

$$u(x, 0) = \frac{1}{2} \left(1 + \sin \left(\frac{\pi}{2} x \right) \right) ,$$

which leads to the solution

$$u(x, t) = \frac{1}{2} + \frac{1}{2} \sin\left(\frac{\pi}{2}x\right) \exp\left(-\frac{\pi^2}{4}t\right).$$

The simulation was conducted by first sampling N particles from the initial distribution $u(x, 0)$. Several standard Monte Carlo techniques, including acceptance-rejection, were used with quasi-random sequences to do this sampling. It was found, however, that the initial sampling error was so small relative to the subsequent error in the simulation, that the sampling method did not influence the results. The particles then move to their new position at time Δt by moving from their current position a distance sampled from a Gaussian distribution with variance Δt . This sampling is done by mapping a random (or quasi-random) number uniform on $[0, 1]$ to the interval $(-\infty, \infty)$ using the inverse error function. Boundary conditions are simulated by reflection, so that a particle initially at 0.95 and moving a distance 0.15 ends up at 0.9.

In order for the quasi-random sequence to work, it is again necessary that the particles be reordered at each time step. The ordering was done by simply sorting the particles according to position, so that particle 1 was always the furthest left and particle N was always the furthest on the right.

Since the state space is continuous in this problem, there is no straightforward way to measure the total error of the simulation, as there was for the discrete problem. Indeed the convergence is only weak, and so the accuracy of the particle solution was measured by computing the integral of the solution $u(x, t)$ over the interval $[\cdot75, 1]$. Because the initial data were normalized to have total mass 1, the particle approximation of this integral is just the percentage of the particles that are in the interval $[\cdot75, 1]$. A time step of $\Delta t = 0.05$ was chosen, and the particles were moved 10 steps out to time $t = 0.5$ before the error was calculated. The fact that the sum of n Gaussian random variables with variance Δt is a Gaussian random variable with variance $n\Delta t$ means that the number of steps taken should not affect the accuracy of the solution.

Figure 3 shows the results of the simulation using the first dimension of the Halton sequence. The relative error is plotted here, obtained by scaling the true value of the integral being evaluated to one. This gives a variance of around 5.23. Since the measure of convergence is only one integral of the distribution, there is considerable scatter in the data; so the results were averaged over 50 runs. The more complicated nature of this problem may have also contributed to this scatter.

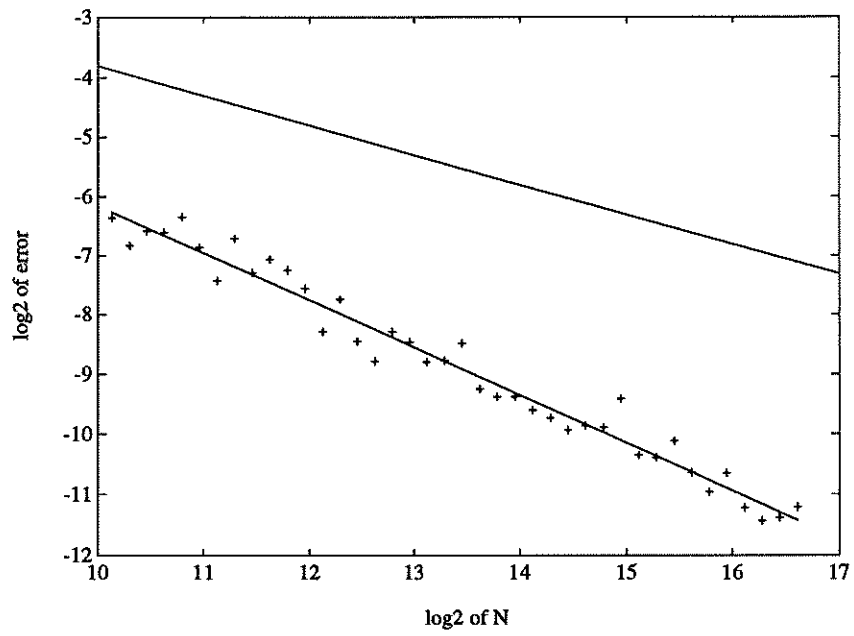


Figure 3: Insulated Heat Equation in One Dimension

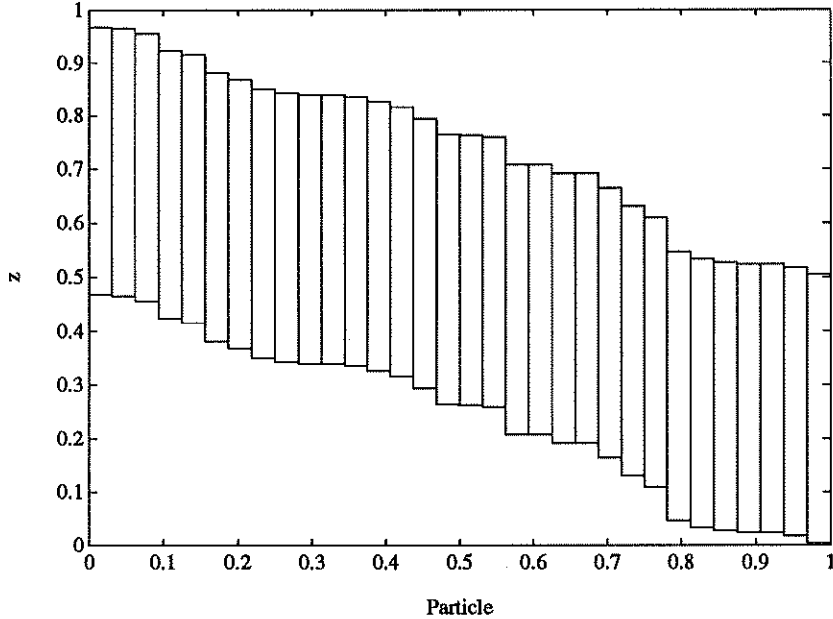


Figure 4: Set Corresponding to Gaussian Step into Interval

The convergence rate is $N^{-.80}$, which is not quite the $1/N$ seen earlier, but still considerably better than the expected random convergence.

The one time step movement of the particles can again be interpreted as the integral of a characteristic function inside the unit square. Again let $\phi_{s,r}$ be the characteristic function of the interval $[s,r]$, where this time $-1 \leq s < r \leq 1$, and for each particle consider the range of the parameter z in $[0,1]$ which under the inverse error function mapping sends the particle into $[s,r]$. Every particle has the potential to end up in $[s,r]$ because in theory, a Gaussian step can be arbitrarily large. Actually, machine precision limits the largest step size to about $5.81\sqrt{2\Delta t}$. If this fact and the effects of reflection at the end points are ignored, then for each particle j there is a subrectangle of the unit square of the form $[j-1/N, j/N) \times [z_j^l, z_j^h]$ which represents movement into $[s,r]$. When the particles are reordered according to position, the union of these rectangles for all N particles looks like the staircase set shown in Figure 4. If the effects of reflection are added, there will be additional, smaller disjoint staircases. The step size limit means there will only be a finite number of the staircases. The accuracy of the simulation is then

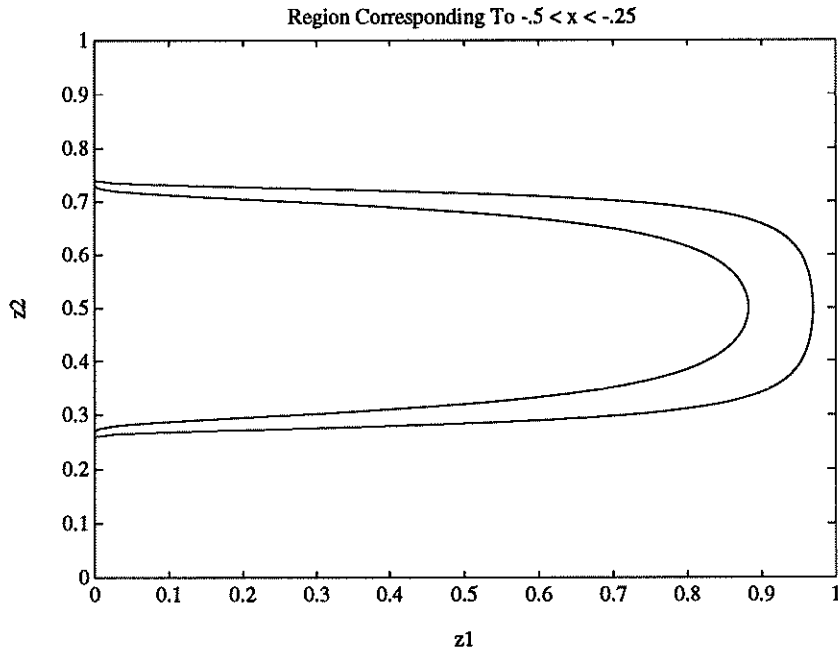


Figure 5: Set Mapped to Interval Using Box-Muller Method

reduced to the question of how accurately quasi-random sequences can integrate this kind of characteristic set. The complicated and non-rectangular shape of this set is believed to be responsible for degradation of the convergence rate from N^{-1} to $N^{-.8}$.

There are other methods of generating Gaussian numbers from uniform numbers on $[0, 1]$ than inverting the error function. One technique, known as the Box-Muller method, involves using two uniform numbers z_1 and z_2 to produce one Gaussian number x through the formula $x = \sqrt{-2 \log z_1} \cos(2\pi z_2)$. Quasi-random sequences can be used to produce Gaussian numbers in this way as long as z_1 and z_2 come from different dimensions. However, there is an additional disadvantage to using this method in the heat equation simulation, other than the fact that an additional dimension is required. The set in the unit square of points (z_1, z_2) which maps x to an interval $[s, r]$ is no longer a rectangle, but is more like a two pronged fork. Figure 5 shows such a region corresponding to choosing a value of x such that $-.5 \leq x \leq -.25$. Thus the set describing the motion of all particles becomes a three dimensional staircase with pointy steps. This suggests that this method will

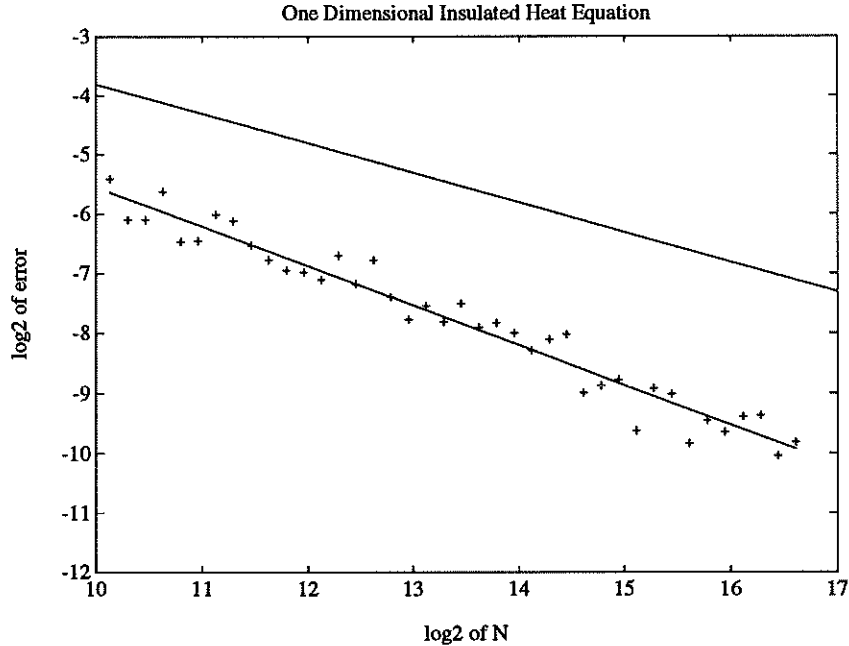


Figure 6: Random Walk Generated Using Box-Muller Method

perform noticeably poorer in the heat equation simulation than the inverse error function approach. Figure 6 confirms this; here the same one dimensional problem was solved as above, but this time using the Box-Muller method to generate the Gaussian steps. The error is larger, and the convergence rate is now only about $N^{-.66}$. Again, it is the relative error that is plotted, and the results are averaged over 22 runs. This shows that extra care must be used when quasi-random sequences are used in place of random numbers. Two methods which may yield identical results in the random case can give quite different results in quasi-random case.

To further investigate the performance of quasi-random numbers for simulations, another experiment was run to solve the two dimensional heat equation. The domain was taken to the unit square, again with insulated boundary conditions. The initial data were taken to be

$$u(x, y, 0) = 1 + \cos(4\pi x) \cos(3\pi y)$$

which satisfies the boundary conditions. This leads to a solution

$$u(x, y, t) = 1 + \cos(4\pi x) \cos(3\pi y) \exp(-25\pi^2 t) .$$

Initially the particle positions were chosen using a quasi-random acceptance-rejection method to sample from $u(x, y, 0)$. The error in this sampling was found to be an order of magnitude smaller than the error from the simulation, so no attempt was made to optimize this step. The particles then moved in steps sampled from a Gaussian distribution with variance Δt . The error was calculated at time $t = 1/1024$ by comparing the percentage of the particles in the rectangle $[1/3, 2/3] \times [0, 1/2]$ with the exact answer

$$\int_0^{1/2} \int_{1/3}^{2/3} u(x, y, t) dx dy = \frac{1}{6} - \frac{\sqrt{3}}{12\pi^2} \exp(-25\pi^2 t) .$$

The simulation again required that the particles be reordered when a quasi-random sequence was used. In two dimensions or higher, there is no absolute way to do the reordering. The method used was to first sort the particles into 1024 radial bins centered at $(1/2, 1/2)$. The bins were formed from uniform spacing along the radius, so that the ones further out had greater area. The particles were then sorted according to angle within the bins. There are many other possible ways to reorder the particles, and no attempt was made here to determine the optimal choice. What works best for any given simulation will probably be quite problem dependent. Figure 7 shows the results of this simulation using 20 time steps, averaged over 21 runs. The two dimensional Halton sequence was used to determine the particle motion. Once again, the quasi-random sequence produced lower errors than the expected random error and faster convergence, although now the convergence is only about $N^{-.69}$.

4 Conclusions

The analysis and computational experiments described above show that quasi-random sequences can give a valid Monte Carlo simulation of the heat equation. Convergence of the simulation as the numerical parameters are refined has been proved. Moreover, numerical experiments show that the error in the quasi-Monte Carlo simulation is significantly less than the corresponding error for a standard Monte Carlo (i.e. random or pseudo-random) simulation.

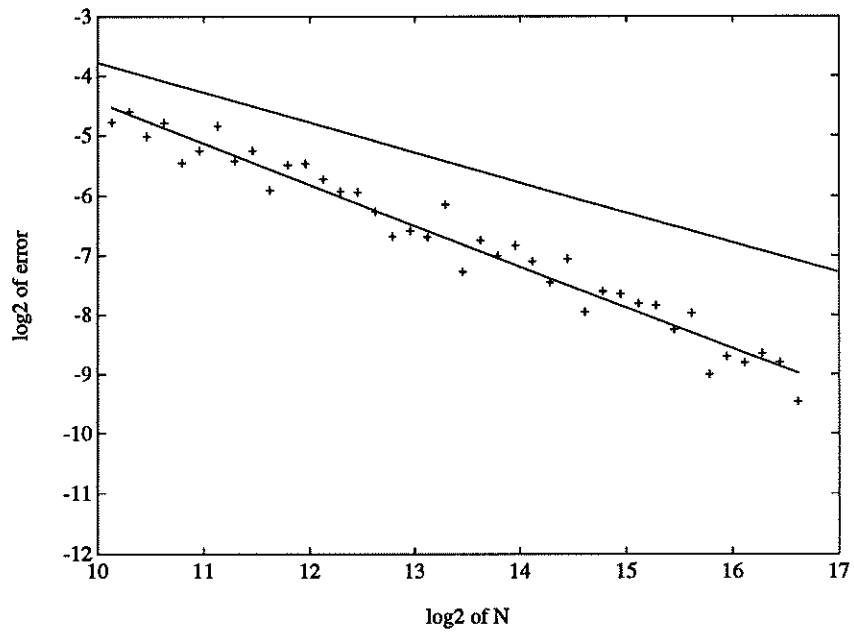


Figure 7: Insulated Heat Equation in Two Dimension

On the other hand the error reduction for quasi-Monte Carlo is limited as the spatial dimension increases, in a way very similar to the behavior of the error for quasi-Monte Carlo evaluation of integrals [10]. For dimension 1 the error for a quasi-Monte Carlo simulation of the heat equation is of size $c_1 N^{-1}$, compared to error size $c_2 N^{-1/2}$ for random simulation (which has this error size for all spatial dimensions). For dimension 2, however, the convergence rate for quasi-Monte Carlo is reduced to $c_3 N^{-.69}$. Still both the exponent and constant are significantly better than those for random simulation. Thus for a fixed error tolerance level, the quasi-random simulation requires significantly smaller number N of simulation points. In many problems, use of the simulation points entails a lot of computation, so that the resulting reduction in computational effort will more than compensate for the increased work required to generate the quasi-random sequence.

Our computations also indicate that use of quasi-random sequences is delicate, since the elements of the sequence are correlated. For example, efficient simulation of the heat equation is only possible if the particle labels are reordered, as described in section 2. While reordering by position is natural in one dimension, there is no reordering scheme that is equally effective in higher dimensions. Moreover, we do not know how to correctly simulate a single realization of a random walk using a quasi-random sequence, as is easily done with random or pseudo-random numbers.

A second example of the difficulty in using quasi-random sequences comes from the Box-Muller method for sampling a Gaussian distribution. This method involves a mapping of the plane into itself that is highly stretched and curved in some regions. Since quasi-random sequences are best at representing the area of rectangular regions, this stretching and curving results in significant loss of efficiency when sampling a quasi-random sequence with the Box-Muller method.

These difficulties suggest that great care must be used in applying variance reduction techniques to quasi-Monte Carlo simulation. We expect this to be even more true for simulation of nonlinear processes, such as a reaction-diffusion equation, the random vortex method for the Navier-Stokes equations, or direct simulation for the nonlinear Boltzmann equation [7].

References

- [1] Bird, G.A.: *Molecular Gas Dynamics*, Clarendon Press, Oxford, 1976.
- [2] Bratley, P. and Fox, B.L.: Implementing Sobol's quasirandom sequence generator, *ACM Trans. Math. Software* 14, 88 - 100, 1988.
- [3] Faure, Henri: Discrépance de suites associées à un système de numération (en dimension s), *Acta Arithmetica* 41, 337-351, 1982.
- [4] Fox, B.L.: Implementation and relative efficiency of quasirandom sequence generators, *ACM Trans. Math. Software* 12, 362 - 376, 1986.
- [5] Halton, J. H.: On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals, *Numer. Math.* 2, 84-90, 1960.
- [6] Kalos, Malvin H. and Whitlock, Paula A.: *Monte Carlo Methods, Volume I*, John Wiley and Son, 1986.
- [7] Lécot, Christian: Low discrepancy sequences for solving the Boltzmann equation, *Journal of Comp. and Applied Math.* 25, 237 - 249, 1989.
- [8] Levitan, Yu. L., et al: Short communications on quasi-random sequences for numerical computations, *U.S.S.R. Computational Math. and Math. Phys.* 28, no. 3, 88-92, 1988.
- [9] Morokoff, William J. and Caffisch, Russel E.: Quasi-random sequences and their discrepancies, to appear.
- [10] Morokoff, William J. and Caffisch, Russel E.: Quasi-Monte Carlo integration, to appear.
- [11] Niederreiter, H.: Quasi-Monte Carlo methods and pseudo-random numbers, *Bull. Amer. Math. Soc.* 84, 957 - 1041, 1978.
- [12] Niederreiter, H.: Quasi-Monte Carlo methods for multidimensional numerical integration, *Numerical Integration III*, International Series of Numerical Math. 85, H. Brass and G. Hämmerlin eds., Birkhäuser Verlag, Basel, 1988.

- [13] Niederreiter, H.: Application of diophantine approximations to numerical integration, *Diophantine Approximation and Its Applications* (C.F. Osgood, ed.) Academic Press, New York, 1973, pp. 129-199.
- [14] Niederreiter, H.: Point sets and sequences with small discrepancy, *Monatsh. Math.* 104, 273-337, 1987.
- [15] Press, William H. and Teukolsky, Saul A.: Quasi- (that is, sub-) random numbers, *Computers in Physics*, Nov/Dec 1989, 76-79.
- [16] Sarkar, P.K. and Prasad, M.A.: A comparative study of pseudo and quasi random sequences for the solution of integral equations, *J. Comput. Physics* 68, 66 - 88, 1987.
- [17] Sobol', I. M.: The distribution of points in a cube and the approximate evaluation of integrals, *U.S.S.R. Computational Math. and Math. Phys.* 7, no. 4, 86-112, 1967.
- [18] Sobol', I. M.: Uniformly distributed sequences with additional uniformity property, *U.S.S.R. Computational Math. and Math. Phys.* 16, no. 5, 1332-1337, 1976.
- [19] Woźniakowski, H.: Average case complexity of multivariate integration, *Bull. Amer. Math. Soc.* 24, 185 - 194, 1991.