Chapter 1

Quasi-Monte Carlo Sampling *by* *Art B. Owen*

In Monte Carlo (MC) sampling the sample averages of random quantities are used to estimate the corresponding expectations. The justification is through the law of large numbers. In quasi-Monte Carlo (QMC) sampling we are able to get a law of large numbers with deterministic inputs instead of random ones. Naturally we seek deterministic inputs that make the answer converge as quickly as possible. In particular it is common for QMC to produce much more accurate answers than MC does. Keller [19] was an early proponent of QMC methods for computer graphics.

We begin by reviewing Monte Carlo sampling and showing how many problems can be reduced to integrals over the unit cube $[0,1)^d$. Next we consider how stratification methods, such as jittered sampling, can improve the accuracy of Monte Carlo for favorable functions while doing no harm for unfavorable ones. Method of multiple-stratification such as Latin hypercube sampling (n-rooks) represent a significant improvement on stratified sampling. These stratification methods balance the sampling points with respect to a large number of hyperrectangular boxes. QMC may be thought of as an attempt to take this to the logical limit: how close can we get to balancing the sample points with respect to every box in $[0,1)^d$ at once? The answer, provided by the theory of discrepancy is surprisingly far, and that the result produce a significant improvement compared to MC. This chapter concludes with a presentation of digital nets, integration lattices and randomized QMC.

1.1 Crude Monte Carlo

As a frame of reference for QMC, we recap the basics of MC. Suppose that the average we want to compute is written as an integral

$$I = \int_{\mathcal{D}} f(x)q(x)dx. \tag{1.1}$$

The set $\mathcal{D} \subseteq \mathbb{R}^d$ is the domain of interest, perhaps a region on the unit sphere or in the unit cube. The function q is a probability density function on $\mathcal{D}$. That is $q(x) \geq 0$ and $\int_{\mathcal{D}} q(x)dx = 1$. The function f gives the quantity whose expectation we seek: I is the expected value of $f(x)$ for random x with density q on $\mathcal{D}$.

In crude Monte Carlo sampling we generate n independent samples $x_1, \ldots, x_n$ from the density q and estimate I by

$$\hat{I} = \hat{I}_n = \frac{1}{n}\sum_{i=1}^{n} f(x_i). \tag{1.2}$$

The strong law of large numbers tells us that

$$\Pr\big(\lim_{n\to\infty} \hat{I}_n = I\big) = 1. \tag{1.3}$$

That is, crude Monte Carlo always converges to the right answer as n increases without bound.

Now suppose that f has finite variance $\sigma^2 = \mathrm{Var}(f(x)) \equiv \int_{\mathcal{D}}(f(x)-I)^2 q(x)dx$. Then $E((\hat{I}_n - I)^2) = \sigma^2/n$ so the root mean square error (RMSE) of MC sampling is $O(1/\sqrt{n})$. This rate is slow compared to that of classical quadrature rules (Davis and Rabinowitz [7]) for smooth functions in low dimensions. Monte Carlo methods can improve on classical ones for problems in high dimensions or on discontinuous functions.

A given integration problem can be written in the form (1.1) in many different ways. First, let p be a probability density on $\mathcal{D}$ such that $p(x) > 0$ whenever $q(x)|f(x)| > 0$. Then

$$I = \int_{\mathcal{D}} f(x)q(x)dx = \int_{\mathcal{D}} \frac{f(x)q(x)}{p(x)} p(x)dx$$

and we could as well sample $x_i \sim p(x)$ and estimate I by

$$\hat{I}_p = \hat{I}_{n,p} = \frac{1}{n}\sum_{i=1}^{n} \frac{f(x_i)q(x_i)}{p(x_i)}. \tag{1.4}$$

The RMSE can be strongly affected, for better or worse, by this re-expression, known as importance sampling. If we are able to find a good p that is nearly proportional to fq then we can get much better estimates.

Making a good choice of density p is problem specific. Suppose for instance, that one of the components of x describes the angle $\theta = \theta(x)$ between a ray and a surface normal. The original version of f may include a factor of $\cos(\theta)^\eta$ for some $\eta > 0$. Using a density $p(x) \propto q(x)\cos(\theta)^\eta$ corresponds to moving the cosine power out of the integrand and into the sampling density.

We will suppose that a choice of p has already been made. There is also the possibility of using a mixture of sampling densities p_j as with the balance heuristic of Veach and Guibas [42, 43]. This case can be incorporated by increasing the dimension of x by one, and using that variable to select j from a discrete distribution.

Monte Carlo sampling of $x \sim p$ over $\mathcal{D}$ almost always uses points from a pseudo-random number generator simulating the uniform distribution on the interval from 0 to 1. We will take this to mean the uniform distribution on the half-open interval $[0, 1)$. Suppose that it takes d^* uniform random variables to simulate a point in the dimensional domain $\mathcal{D}$. Often $d^* = d$ but sometimes $d^* = 2$ variables from $[0, 1)$ can be used to generate a point within a surface element in $d = 3$ dimensional space. In other problems we might use $d^* > d$ random variables to generate a p distributed point in $\mathcal{D} \subseteq \mathbb{R}^d$. Chapter **??** describes general techniques for transforming $[0, 1)^d$ into $\mathcal{D}$ and provides some specific examples of use in ray tracing. Devroye [8] is a comprehensive reference on techniques for transforming uniform random variables into one's desired random objects.

Suppose that a point having the $U[0, 1)^{d^*}$ distribution is transformed into a point $\tau(x)$ having the density p on $\mathcal{D}$. Then

$$I = \int_{\mathcal{D}} \frac{f(x)q(x)}{p(x)} p(x)dx = \int_{[0,1)^{d^*}} \frac{f(\tau(x))q(\tau(x))}{p(\tau(x))} dx \equiv \int_{[0,1)^{d^*}} f^*(x)dx \tag{1.5}$$

where f^* incorporates the transformation τ and the density q. Then I is estimated by

$$\hat{I} = \frac{1}{n}\sum_{i=1}^{n} \frac{f(\tau(x_i))q(\tau(x_i))}{p(\tau(x_i))} = \frac{1}{n}\sum_{i=1}^{n} f^*(x_i) \tag{1.6}$$

where x_i are independent $U[0, 1)^{d^*}$ random variables.

Equation (1.5) expresses the original MC problem (1.1) as one of integrating a function f^* over the unit cube in d^* dimensions. We may therefore reformulate the problem as finding $I = \int_{[0,1)^d} f(x)dx$. The new d is the old d^* and the new f is the old f^*.

1.2 Stratification

Stratified sampling is a technique for reducing the variance of a Monte Carlo integral. It was originally applied in survey sampling (see Cochran [4]) and has been adapted in Monte Carlo methods, Fishman [12]. In stratified sampling, the domain of x is written as a union of strata $\mathcal{D} = \bigcup_{h=1}^{H} \mathcal{D}_h$ where $\mathcal{D}_j \bigcap \mathcal{D}_k = \emptyset$ if $j \neq k$. An integral is estimated from within each stratum and then combined. Following the presentation in chapter 1.1, we suppose here that $\mathcal{D} = [0,1)^d$.

Figure 1.1 shows a random sample from the unit square along with 3 alternative stratified samplings. The unit cube $[0,1)^d$ is very easily partitioned into box shaped strata like those shown. It is also easy to sample uniformly in such strata. Suppose that $a, c \in [0,1)^d$ with $a < c$ componentwise. Let $U \sim U[0,1)^d$. Then $a + (c - a)U$ interpreted componentwise is uniformly distributed on the box with lower left corner a and upper right corner c.

In the simplest form of stratified sampling, a Monte Carlo sample $x_{h1}, \ldots x_{hn_h}$ is taken from within stratum $\mathcal{D}_h$. Each stratum is sampled independently, and the results are combined as

$$\hat{I}_{\text{STRAT}} = \hat{I}_{\text{STRAT}}(f) = \sum_{h=1}^{H} \frac{|\mathcal{D}_h|}{n_h} \sum_{i=1}^{n_h} f(x_{hi}), \tag{1.7}$$

where $|\mathcal{D}_h|$ is the volume of stratum $\mathcal{D}$.

For any $x \in [0,1)^d$ let $h(x)$ denote the stratum containing x. That is $x \in \mathcal{D}_{h(x)}$. The mean and variance of f within stratum h are

$$\mu_h = |\mathcal{D}_h|^{-1} \int_{\mathcal{D}_h} f(x)dx, \quad \text{and,} \tag{1.8}$$

$$\sigma_h^2 = |\mathcal{D}_h|^{-1} \int_{\mathcal{D}_h} (f(x) - \mu_h)^2 dx \tag{1.9}$$

respectively. We can write $E(\hat{I}_{\text{STRAT}})$ as:

$$\sum_{h=1}^{H} \frac{|\mathcal{D}_h|}{n_h} \sum_{i=1}^{n_h} E(f(x_{hi})) = \sum_{h=1}^{H} |\mathcal{D}_h| \mu_h = \sum_{h=1}^{H} \int_{\mathcal{D}_h} f(x)dx = I,$$

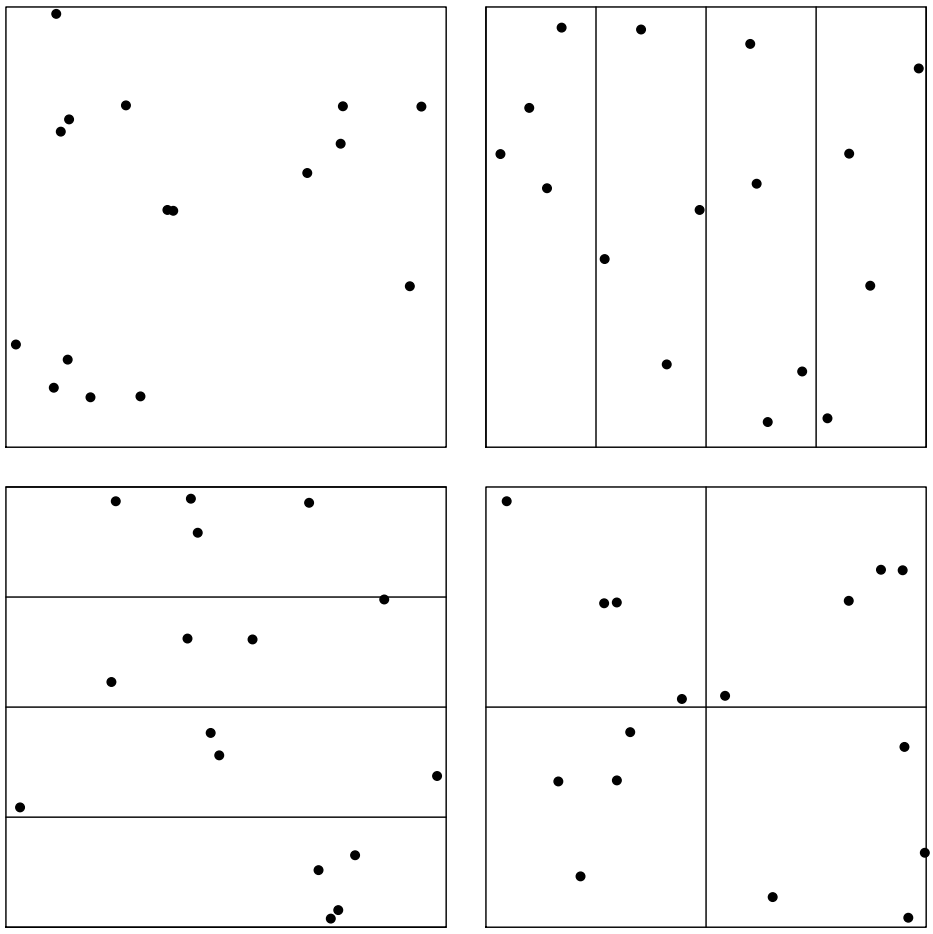

Figure 1.1: The upper left figure is a simple random sample of 16 points in $[0,1)^2$. The other figures show stratified samples with 4 points from each of 4 strata.

so that stratified sampling is unbiased.

The variance of stratified sampling depends on the allocation of sample size n_h to strata. We will suppose that n_h is allocated proportionally, so that $n_h = n|\mathcal{D}_h|$ for the total sample size n. First we note that when $x \sim U[0,1)^d$, then $h(x)$ is a random variable taking the value ℓ with probability $|\mathcal{D}_\ell|$. Then from a standard

variance formula

$$\sigma^2 = \mathrm{Var}(f(x)) = E(\mathrm{Var}(f(x) \mid h(x))) + \mathrm{Var}(E(f(x) \mid h(x))) \tag{1.10}$$

$$= \sum_{h=1}^{H} |\mathcal{D}_h|\sigma_h^2 + \sum_{h=1}^{H} |\mathcal{D}_h|(\mu_h - I)^2, \tag{1.11}$$

so that σ^2 is a sum of contributions from within and between strata. Now

$$\mathrm{Var}(\hat{I}_{\text{STRAT}}) = \sum_{h=1}^{H} \frac{|\mathcal{D}_h|^2}{n_h}\sigma_h^2 = \frac{1}{n}\sum_{h=1}^{H} |\mathcal{D}_h|\sigma_h^2 \le \frac{\sigma^2}{n}, \tag{1.12}$$

from (1.10).

Equation (1.12) shows that stratified sampling with proportional allocation does not increase the variance. Proportional allocation is not usually optimal. Optimal allocations take $n_h \propto |\mathcal{D}_h|\sigma_h$. If estimates of σ_h are available they can be used to set n_h, but poor estimates of σ_h could result in stratified sampling with larger variance than crude MC. We will assume proportional allocation.

A particular form of stratified sampling is well suited to the unit cube. Haber [13] proposes to partition the unit cube $[0,1)^d$ into $H = m^d$ congruent cubical regions and to take $n_h = 1$ point from each of them. This stratification is known as jittered sampling in graphics, following Cook, Porter and Carpenter [5].

Any function that is constant within strata is integrated without error by $\hat{I}_{\text{STRAT}}$. If f is close to such a function, then f is integrated with a small error. Let $\bar{f}$ be the function defined by $\bar{f}(x) = \mu_{h(x)}$, and define the residual $f_{\text{RES}}(x) = f(x) - \bar{f}(x)$. This decomposition is illustrated in Figure 1.2 for a function on $[0,1)$. Stratified sampling reduces the Monte Carlo variance from $\sigma^2(f)/n$ to $\sigma^2(f_{\text{RES}})/n$.

1.3 Multiple Stratification

Suppose we can afford to sample 16 points in $[0,1)^2$. Sampling one point from each of 16 vertical strata would be a good strategy if the function f depended primarily on the horizontal coordinate. Conversely if the vertical coordinate is the more important one, then it would be better to take one point from each of 16 horizontal strata.

It is possible to stratify both ways with the same sample, in what is known as Latin hypercube sampling (McKay, Beckman and W. J. Conover [24]) or n-rooks

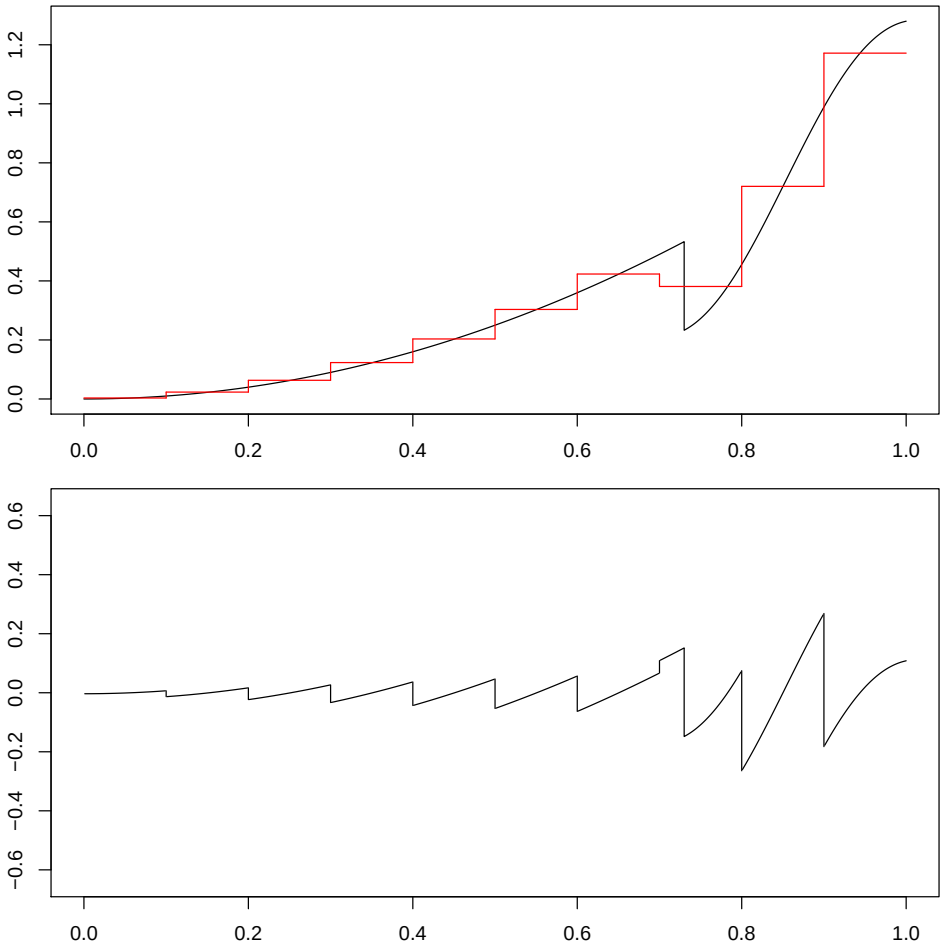

Figure 1.2: The upper plot shows a piece-wise smooth function f on $[0,1)$. The step function is the best approximation $\bar{f}$ to f, in mean square error, among functions constant over intervals $[j/10,(j+1)/10)$. The lower plot shows the difference $f-\bar{f}$ using a vertical scale similar to the upper plot.

sampling (Shirley [33]). Figure 1.3 shows a set of 16 points in the square, that are simultaneously stratified in each of 16 horizontal and vertical strata.

If the function f on $[0,1)^2$ is dominated by either the horizontal coordinate or the vertical one, then we'll get an accurate answer, and we don't even need to know which is the dominant variable. Better yet, suppose that neither variable is

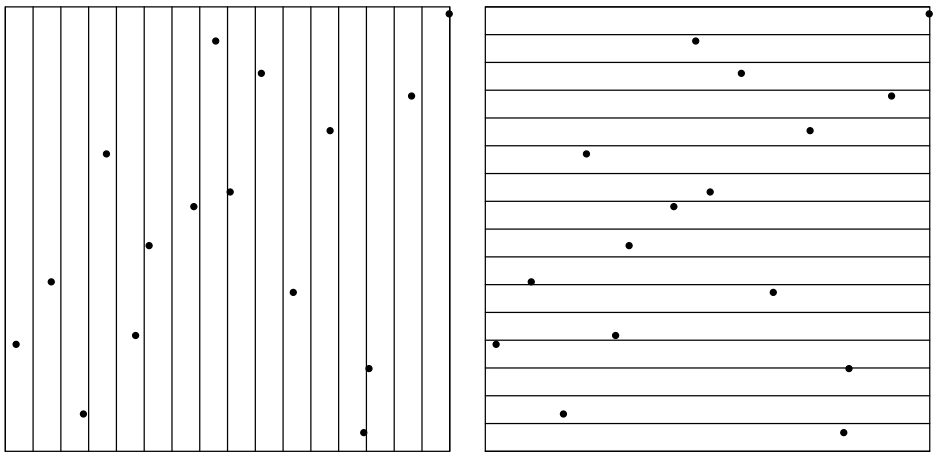

Figure 1.3: The left plot shows 16 points, one in each of 16 vertical strata. The right plot shows the same 16 points. There is one in each of 16 horizontal strata. These points form what is called a Latin hypercube sample, or an n-rooks pattern.

dominant but that

$$f(x) = f_H(x) + f_V(x) + f_{\text{RES}}(x) \tag{1.13}$$

where f_H depends only on the horizontal variable, f_V depends only on the vertical one, and the residual f_{RES} is defined by subtraction. Latin hypercube sampling will give an error that is largely unaffected by the additive part $f_H + f_V$. Stein [37] showed that the variance in Latin hypercube sampling is approximately σ^2_{RES}/n where σ^2_{RES} is the smallest variance of f_{RES} for any decomposition of the form (1.13). His result is for general d, not just $d = 2$.

Stratification with proportional allocation is never worse than crude MC. The same is almost true for Latin hypercube sampling. Owen [28] shows that for all $n \geq 2$, $d \geq 1$ and square integrable f, that

$$\text{Var}(\hat{I}_{\text{LHS}}) \leq \frac{\sigma^2}{n-1}.$$

For the worst f, Latin hypercube sampling is like using crude MC with one observation less.

The construction of a Latin hypercube sample requires uniform random permutations. A uniform random permutation of 0 through $n-1$ is one for which all $n!$

possible orderings have the same probability. Devroye [8] gives algorithms for such random permutations. One choice is to have an array $A_i = i$ for $i = 0, \dots, n-1$ and then for $j = n-1$ down to 1 swap A_j with A_k where k is uniformly and randomly chosen from 0 through j.

For $j = 1, \dots, d$, let π_j be independent uniform random permutations of $0, \dots, n-1$. Let $U_{ij} \sim U[0,1)^d$ independently for $i = 1, \dots, n$ and $j = 1, \dots, d$ and let X be a matrix with

$$X_{ij} = \frac{\pi_j(i-1) + U_{ij}}{n}.$$

Then the n rows of X form a Latin hypercube sample. That is we may take $x_i = (X_{i1}, \dots, X_{id})$. An integral estimate $\hat{I}$ is the same whatever order the $f(x_i)$ are summed. As a consequence we only need to permute $d-1$ of the d input variables. We can take $\pi_1(i-1) = i-1$ to save the cost of one random permutation.

Jittered sampling uses $n = k^2$ strata arranged in a k by k grid of squares while n-rooks provides simultaneous stratification in both an n by 1 grid and a 1 by n grid. It is natural to wonder which method is better. The answer depends on whether f is better approximated by a step function, constant within squares of size $1/k \times 1/k$ grid, or by an additive function with each term constant within narrower bins of width $1/n$. Amazingly, we don't have to choose. It is possible to arrange $n = k^2$ points in an n-rooks arrangement that simultaneously has one point in each square of a k by k grid. A construction for this was proposed independently by Chiu, Shirley and Wang [2] and by Tang [38]. The former handle more general grids of $n = k_1 \times k_2$ points. The latter reference arranges points in $[0,1)^d$ with $d \geq 2$ in a Latin hypercube such that every two dimensional projection of x_i puts one point into each of a grid of strata.

1.4 Uniformity and Discrepancy

The previous sections look at stratifications in which every cell in a rectangular grid or indeed in multiple rectangular grids gets the proper number of points. It is clear that a finite number of points in $[0,1)^d$ cannot be simultaneously stratified with respect to *every* hyper-rectangular subset of $[0,1)^d$, yet it is interesting to ask how far we might be able to go in that direction. This is a problem that has been studied since Weyl [44] originated his theory of uniform distribution. Kuipers and Niederreiter [21] summarize that theory.

Let a and c be points in $[0,1)^d$ for which $a < c$ holds componentwise, and then let $[a,c)$ denote the box of points x where $a \le x < c$ holds componentwise. We use $|[a,c)|$ to denote the d-dimensional volume of this box.

An infinite sequence of points $x_1, x_2, \cdots \in [0,1)^d$ is uniformly distributed if $\lim_{n\to\infty}(1/n)\sum_{i=1}^n 1_{a\le x_i<c} = |[a,c)|$ holds for all boxes. This means that $\hat{I}_n \to I$ for every function $f(x)$ of the form $1_{a\le x<c}$ and so for any finite linear combination of such indicators of boxes. It is known that $\lim_{n\to\infty}|\hat{I}_n - I| = 0$ for uniformly distributed x_i and any function f that is Riemann integrable. Thus uniformly distributed sequences can be used to provide a deterministic law of large numbers.

To show that a sequence is uniformly distributed it is enough to show that $\hat{I}_n \to I$ when f is the indicator of a suitable subset of boxes. Anchored boxes take the form $[0,a)$ for some point $a \in [0,1)^d$. If $\hat{I}_n \to I$ for all indicators of anchored boxes, then the same holds for all boxes. For integers $b \ge 2$ a b-adic box is a Cartesian product of the form

$$\prod_{j=1}^d \Big[\frac{\ell_j}{b^{k_j}}, \frac{\ell_j+1}{b^{k_j}}\Big). \tag{1.14}$$

for integers $k_j \ge 0$ and $0 \le \ell_j < b^{k_j}$. When $b = 2$ the box is called dyadic. An arbitrary box can be approximated by b-ary boxes. If $\hat{I} \to I$ for all indicators of b-adic boxes then the sequence (x_i) is uniformly distributed. A mathematically more interesting result is the Weyl condition. The sequence (x_i) is uniformly distributed if and only if $\hat{I}_n \to I$ for all trigonometric polynomials $f(x) = e^{2\pi\sqrt{-1}k\cdot x}$ where $k \in \mathbb{Z}^d$.

If x_i are independent $U[0,1)^d$ variables, then (x_i) is uniformly distributed with probability one. Of course we hope to do better than random points. To that end, we need a numerical measure of how uniformly distributed a sequence of points is. These measures are called discrepancies, and there are a great many of them. One of the simplest is the star discrepancy

$$D_n^* = D_n^*(x_1,\dots,x_n) = \sup_{a\in[0,1)^d}\Big|\frac{1}{n}\sum_{i=1}^n 1_{0\le x_i<a} - \big|[0,a)\big|\Big| \tag{1.15}$$

Figure 1.4 illustrates this discrepancy. It shows an anchored box $[0,a) \in [0,1)^2$ and a list of $n = 20$ points. The anchored box has 5 of the 20 points so $(1/n)\sum_{i=1}^n 1_{0\le x_i<a} = 0.20$. The volume of the anchored box is 0.21, so the difference is $|0.2 - 0.21| =$

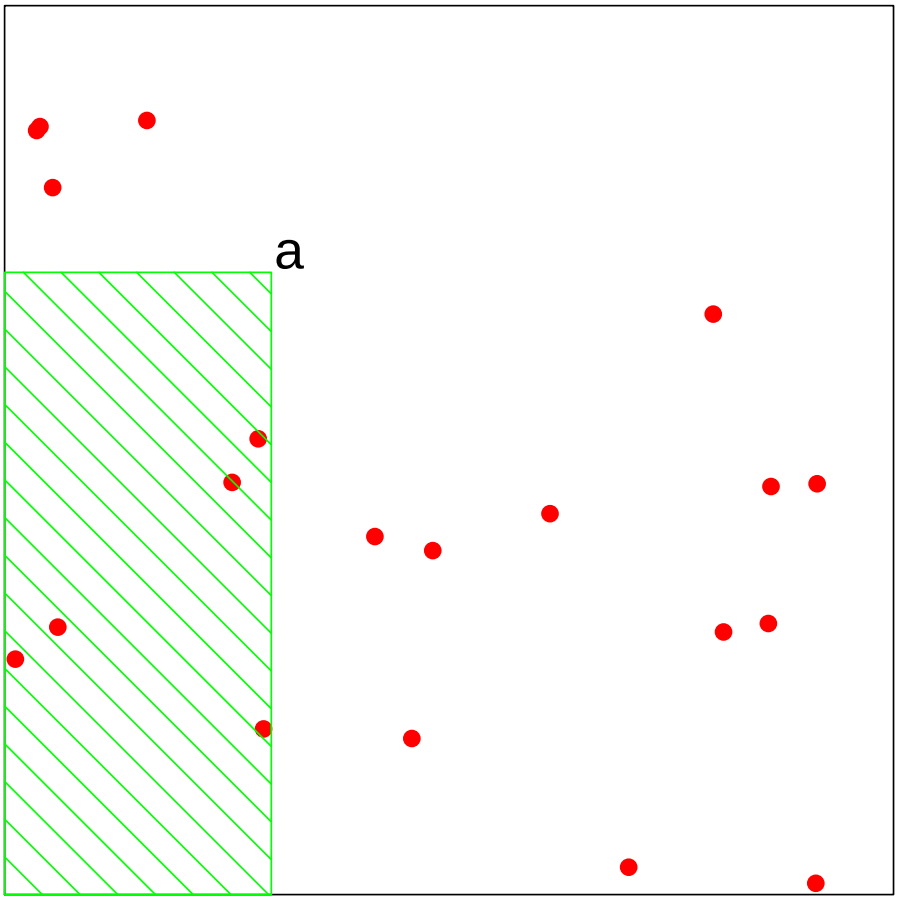

Figure 1.4: Shown are 20 points in the unit square and an anchored box (shaded) from $(0,0)$ to $a = (.3,.7)$. The anchored box $[0,a)$ has volume 0.21 and contains a fraction $5/20 = 0.2$ of the points.

0.01. The star discrepancy D_n^* is found by maximizing this difference over all anchored boxes $[0,a)$.

For $x_i \sim U[0,1)^d$, Chung [3] showed that

$$\limsup_{n\to\infty} \frac{\sqrt{2n}D_n^*}{\sqrt{\log(\log(n))}} = 1 \tag{1.16}$$

so $D_n^* = O((\log\log(n)/n)^{1/2})$ with probability one. An iterated logarithm grows slowly with n, so D_n^* may be only slightly larger than $n^{-1/2}$ for large n.

It is known that a deterministic choice of (x_i) can yield D_n^* much smaller than (1.16). There are infinite sequences (x_i) in $[0,1)^d$ with $D_n^*(x_1,\dots,x_n) = O(\log(n)^d/n)$. Such sequences are called "low discrepancy" sequences, and some of them are described in chapter 1.5. It is suspected but not proven that infinite sequences cannot be constructed with $D_n^* = o(\log(n)^d/n)$; see Beck and Chen [1].

In an infinite sequence, the first m points of $x_1,\dots,x_n$ are the same for any $n \geq m$. If we knew in advance the value of n that we wanted then we might use a sequence customized for that value of n, such as $x_{n1},\dots,x_{nn} \in [0,1)^d$, without insisting that $x_{ni} = x_{n+1\,i}$. In this setting $D_n^*(x_{n1},\dots,x_{nn}) = O(\log(n)^{d-1}/n)$ is possible. The effect is like reducing d by one, but the practical cost is that such a sequence is not extensible to larger n.

There is a connection between better discrepancy and more accurate integration. Hlawka [16] proved the Koksma-Hlawka inequality

$$|\hat{I} - I| \leq D_n^*(x_1,\dots,x_n) V_{\text{HK}}(f). \tag{1.17}$$

The factor $V_{\text{HK}}(f)$ is the total variation of f in the sense of Hardy and Krause. Niederreiter [26] gives the definition.

Equation (1.17) shows that a deterministic law of large numbers can be much better than the random one, for large enough n and a function f with finite variation $V_{\text{HK}}(f)$. One often does see QMC methods performing much better than MC, but equation (1.17) is not good for predicting when this will happen. The problem is that D_n^* is hard to compute, $V_{\text{HK}}(f)$ is harder still, and that the bound (1.17) can grossly overestimate the error. In some cases V_{HK} is infinite while QMC still beats MC. Schlier [32] reports that even for QMC the variance of f is more strongly related to the error than is the variation.

1.5 Digital Nets and Related Methods

Niedereitter [26] presents a comprehensive account of digital nets and sequences. We will define them below, but first we illustrate a construction for $d = 1$.

The simplest digital nets are the radical inverse sequences initiated by van der Corput [40, 41]. Let $b \geq 2$ be an integer base. The non-negative integer n can be written as $\sum_{k=1}^{\infty} n_k b^{k-1}$ where $n_k \in \{0,1,\dots,b-1\}$ and only finitely many n_k are not zero. The base b radical inverse function is $\phi_b(n) = \sum_{k=1}^{\infty} n_k b^{-k} \in$

ℓ	ℓ base 2	$\phi_2(\ell)$	
0	0.	0.000	0.000
1	1.	0.100	0.500
2	10.	0.010	0.250
3	11.	0.110	0.750
4	100.	0.001	0.125
5	101.	0.101	0.625
6	110.	0.011	0.375
7	111.	0.111	0.875

Table 1.1: The first column shows integers ℓ from 0 to 7. The second column shows ℓ in base 2. The third column reflects the digits of ℓ through the binary point to construct $\phi_2(\ell)$. The final column is the decimal version of $\phi_2(\ell)$.

$[0,1)$. A radical inverse sequence consists of $\phi_b(i)$ for n consecutive values of i, conventionally 0 through $n-1$.

Table 1.1 illustrates a radical inverse sequence, using $b = 2$ as van der Corput did. Because consecutive integers alternate between even and odd, the van der Corput sequence alternates between values in $[0, 1/2)$ and $[1/2, 1)$. Among any 4 consecutive van der Corput points there is exactly one in each interval $[k/4, (k+1)/4)$ for $k = 0, 1, 2, 3$. Similarly any b^m consecutive points from the radical inverse sequence in base b are stratified with respect to b^m congruent intervals of length $1/b^m$.

If $d > 1$ then it would be a serious mistake to simply replace a stream of pseudo-random numbers by the van der Corput sequence. For example with $d = 2$ taking points $x_i = (\phi_2(2i-2), \phi_2(2i-1)) \in [0,1)^2$ we would find that all x_i lie on a diagonal line with slope 1 inside $[0, 1/2) \times [1/2, 1)$.

For $d > 1$ we really need a stream of quasi-random d-vectors. There are several ways to generalize the van der Corput sequence to $d \geq 1$. The Halton [14] sequence in $[0,1)^d$ works with d relatively prime bases $b_1, \ldots, b_d$. Usually these are the first d prime numbers. Then for $i \geq 1$,

$$x_i = (\phi_2(i-1), \phi_3(i-1), \phi_5(i-1), \ldots, \phi_{b_d}(i-1)) \in [0,1)^d.$$

The Halton sequence has low discrepancy: $D_n^* = O((\log n)^d/n)$.

The Halton sequence is extensible in both n and d. For small d the points of the Halton sequence have a nearly uniform distribution. The left panel of Figure 1.5 shows a two dimensional portion of the Halton sequence using prime bases 2 and

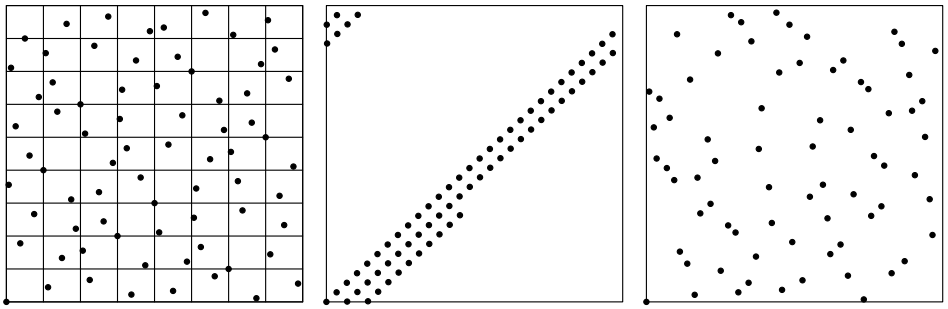

Figure 1.5: The left panel shows the first $2^3 \times 3^2 = 72$ points of the Halton sequence using bases 2 and 3. The middle panel shows the first 72 points for the 10'th and 11'th primes, 29 and 31 respectively. The right panel shows these 72 points after Faure's [11] permutation is applied.

3. The second panel shows the same points for bases 29 and 31 as would be needed with $d = 11$. While the are nearly uniform in one dimensional problems, their two dimensional uniformity is seriously lacking. When it is possible to identify the more important components of x, these should be sampled using the smaller prime bases.

The poorer distribution for larger primes can be mitigated using a permutation of Faure [11]. Let π be a permutation of $\{0, \dots, b-1\}$. Then the radical inverse function can be generalized to $\phi_{b,\pi}(n) = \sum_{k=1}^{\infty} \pi(n_k) b^{-k}$. It still holds that any consecutive b^m values of $\phi_{b,\pi}(i)$ stratify into b^m boxes of length $1/b^m$. Faure's transformation π_b of $0, \dots, b-1$ is particularly simple. Let $\pi_2 = (0, 1)$. For even $b > 2$ take $\pi_b = (2\pi_{b/2}, 2\pi_{b/2} + 1)$, so $\pi_4 = (0, 2, 1, 3)$. For odd $b > 2$ put $k = (b-1)/2$ and $\eta = \phi_{b-1}$. Then add 1 to any member of η greater than or equal to k. Then $\pi_b = (\eta(0), \dots, \eta(k-1), k, \eta(k), \dots, \eta(b-2))$. For example with $b = 5$ we get $k = 2$, and after the larger elements are incremented, $\eta = (0, 3, 1, 4)$. Finally $\pi_5 = (0, 3, 2, 1, 4)$. The third plot in Figure 1.5 shows the effect of Faure's permutations on the Halton sequence.

Digital nets provide more satisfactory generalizations of radical inverse sequences to $d \geq 2$. Recall the b-ary boxes in (1.14). The box there has volume b^{-K} where $K = k_1 + \cdots + k_d$. Ideally we would like nb^{-K} points in every such box. Digital nets do this, at least for small enough K.

Let $b \geq 2$ be an integer base and let $m \geq t \geq 0$ be integers. A (t, m, d)–*net in base* b is a finite sequence $x_1, \ldots, x_{b^m}$ for which every b-ary box of volume b^{t-m} contains exactly b^t points of the sequence.

Clearly $t = 0$ corresponds to better stratification. For given values of b, m, and d, particularly for large d, there may not exist a net with $t = 0$, and so nets with $t > 0$ are widely used.

Faure [10] provides a construction of $(0, m, p)$–nets in base p where p is a prime number. The first component of these nets is the radical inverse function in base p applied to 0 through $b^m - 1$. Figure 1.6 shows 81 points of a $(0, 4, 2)$–net in base 3. There are 5 different shapes of 3-ary box with volume $1/81$. The aspect ratios are $1 \times 1/81$, $1/3 \times 1/27$, $1/9 \times 1/9$, $1/17 \times 1/3$, and $1/81 \times 1$. Latin hypercube samples of 81 points balance the first and last of these, jittered sampling balances the third, while multi-jittered sampling balances the first, third, and fifth. A $(0, 4, 2)$–net balances 81 different 3-ary boxes of each of these 5 aspect ratios. If f is well approximated by a sum of the corresponding 405 indicator functions, then $|\hat{I} - I|$ will be small.

The extensible version of a digital net is a digital sequence. A (t, s)–*sequence in base* b is an infinite sequence (x_i) for $i \geq 1$ such that for all integers $r \geq 0$ and $m \geq t$, the points $x_{rb^m+1}, \ldots, x_{(r+1)b^m}$ form a (t, m, d)–net in base b. This sequence can be expressed as an infinite stream of (t, m, d)–nets, simultaneously for all $m \geq t$. Faure [10] provided a construction of $(0, p)$-sequences in base p. Niederreiter [25] showed that construction can be extended to $(0, q)$–sequences in base q where $q = p^r$ is a power of a prime p. The Faure net shown in Figure 1.6 is in fact the first 81 points of the first two variables in a $(0, 3)$-sequence in base 3.

For $m \geq t$ and $1 \leq \lambda < b$, the first λb^m points in a (t, d)–sequence are balanced with respect to all b-ary boxes of volume b^{t-m} or larger. If n is not of the form λb^m, then the points do not necessarily balance any non-trivial b-ary boxes.

The Faure sequence and Niederreiter's generalization of it, require $b \geq d$. When the dimension is large then it becomes necessary to use a large base b, and then either b^m is very large, or m is very small. Then the Sobol' [35] sequences become attractive. They are (t, d)–sequences in base $b = 2$. The quality parameter t depends on d. Niederreiter [25] combined the methods of Sobol' and Faure, generating new sequences. Any (t, s)–sequence is a low discrepancy sequence, as shown in Niederreiter [26].

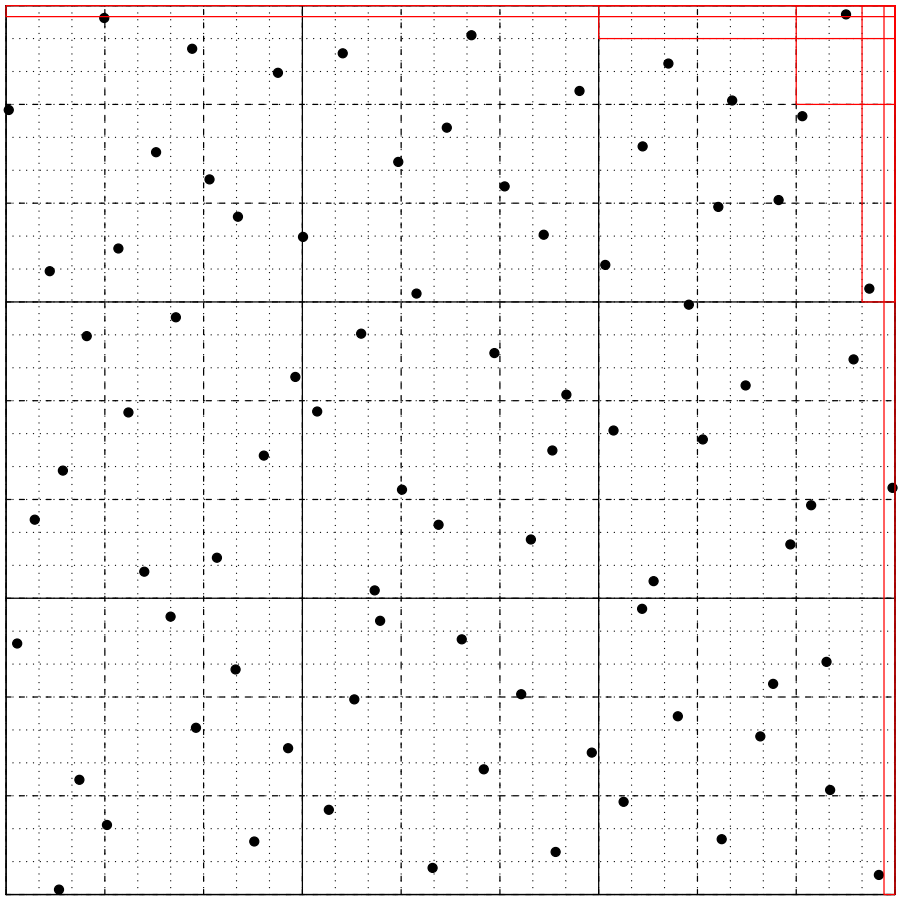

Figure 1.6: Shown are 81 points of a $(0, 4)$–net in base 3. Reference lines are included to make the 3-ary boxes more visible. There 5 different shapes of 3-ary box balanced by these points. One box of each shape is highlighted.

1.6 Integration Lattices

In addition to digital nets and sequences, there is a second major QMC technique, known as integration lattices. The simplest example of an integration lattice is a rank one lattice. These take the form

$$x_i = (i-1)(g_1, \ldots, g_d) \mod n \tag{1.18}$$

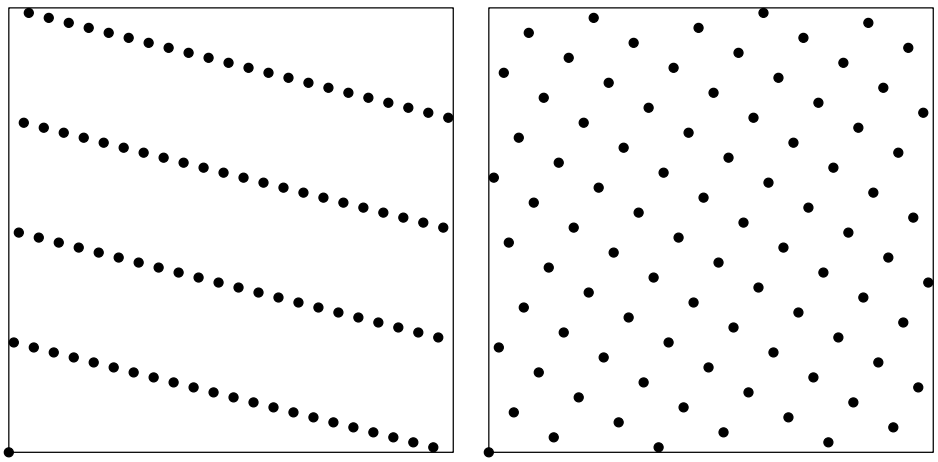

Figure 1.7: Shown are the points of two integration lattices in the unit square. The lattice on the right has much better uniformity, showing the importance of making a good choice of lattice generator.

for $i = 1, \ldots, n$. Usually $g_1 = 1$. Figure 1.7 shows two integration lattices in $[0,1)^2$ with $n = 89$. The first has $g_2 = 22$ and the second one has $g_2 = 55$.

It is clear that the second lattice in Figure 1.7 is more evenly distributed than the first one. The method of good lattice points is the application of the rule (1.18) with n and g carefully chosen to get good uniformity. Fang and Wang [9] and Hua and Wang [18] describe construction and use of good lattice points, including extensive tables of n and g.

Sloan and Joe [34] describe integration lattices in general, including lattices of rank higher than 1. A lattice of rank r for $1 \le r \le d$ requires r vectors like g to generate it. The origin of lattice methods is in Korobov [20]. Korobov's rules have $g = (1, h, h^2, \ldots, h^{d-1})$ so that the search for a good rule requires only a careful choice of two numbers n and h.

Until recently, integration lattices were not extensible. Extensible integration lattices are a research topic of current interest, following the publication of Hickernell, Hong, L'Ecuyer and Lemieux [15].

Integration lattices are not as widely used in computer graphics as digital nets. Their periodic structure is likely to produce unwanted aliasing artifacts, at least in some applications. Compared to digital nets, integration lattices are very good at integrating smooth functions, especially smooth periodic functions.

1.7 Randomized Quasi-Monte Carlo

QMC methods may be thought of as derandomized MC. Randomized QMC (RQMC) methods re-randomize them. The original motivation is to get sample based error estimates.

In RQMC, one takes a QMC sequence (a_i) and transforms it into random points (x_i) such that x_i retain a QMC property and the expectation of $\hat{I}$ is I. The simplest way to achieve the latter property is to have each $x_i \sim U[0,1)^d$. With RQMC we can repeat a QMC integration R times independently getting $\hat{I}_1, \dots, \hat{I}_R$. The combined estimate $\hat{I} = (1/R)\sum_{r=1}^{R} \hat{I}_r$ has expected value I and an unbiased estimate of the RMSE of $\hat{I}$ is $R^{-1}(R-1)^{-1}\sum_{r=1}^{r}(\hat{I}_r - \hat{I})^2$.

Cranley and Patterson [6] proposed a rotation modulo one

$$x_i = a_i + U \mod 1$$

where $U \sim U[0,1)^d$ and both addition and remainder modulo one are interpreted componentwise. It is easy to see that each $x_i \sim U[0,1)^d$. Cranley and Patterson proposed rotations of integration lattices. Tuffin [39] considered applying such rotations to digital nets. They don't remain nets, but they still look very uniform.

Owen [27] proposes a scrambling of the base b digits of a_i. Suppose that a_i is the i'th row of the matrix A with entries A_{ij} for $j = 1, \dots, d$, and either $i = 1, \dots, n$ for a finite sequence or $i \geq 1$ for an infinite one. Let $A_{ij} = \sum_{k=1}^{\infty} b^{-k} a_{ijk}$ where $a_{ijk} \in \{0, 1, \dots, b-1\}$. Now let $x_{ijk} = \pi_{j \cdot a_{ij1} \dots a_{ij\,k-1}}(a_{ijk})$ where $\pi_{j \cdot a_{ij1} \dots a_{ij\,k-1}}$ is a uniform random permutation of $0, \dots, b-1$. All the permutations required are independent, and the permutation applied to the k'th digits of A_{ij} depends on j and on the preceding $k-1$ digits.

Applying this scrambling to any point $a \in [0,1)^d$ produces a point $x \sim U[0,1)^d$. If (a_i) is a (t, m, d)–net in base b or a (t, d)–sequence in base b, then with probability 1, the same holds for the scrambled version (x_i). The scrambling described above requires a great many permutations. Random linear scrambling is a partial derandomization of scrambled nets, given by Matoušek [23] and also

in Hong and Hickernell [17]. Random linear scrambling significantly reduces the number of permutations required from $O(db^m)$ to $O(dm^2)$.

For integration over a scrambled digital sequence we have $\mathrm{Var}(\hat{I}) = o(1/n)$ for any f with $\sigma^2 < \infty$. Thus for large enough n a better than MC result will be obtained. For integration over a scrambled $(0, m, d)$-net Owen [28] shows that

$$\mathrm{Var}(\hat{I}) \le \Big(\frac{b}{b-1}\Big)^{\min(d-1,m)} \frac{\sigma^2}{n} \le \frac{2.72\,\sigma^2}{n}.$$

That is scrambled $(0, m, d)$–nets cannot have more than $e = \exp(1) \doteq 2.72$ times the Monte Carlo variance for finite n. For nets in base $b = 2$ and $t \ge 0$, Owen [30] shows that

$$\mathrm{Var}(\hat{I}) \le 2^t 3^d \, \frac{\sigma^2}{n}.$$

Compared to QMC, we expect RQMC to do no harm. After all, the resulting x_i still have a QMC structure, and so the RMSE should be $O(n^{-1}(\log n)^d)$. Some forms of RQMC reduce the RMSE to $O(n^{-3/2}(\log n)^{(d-1)/2})$ for smooth enough f. This can be understood as random errors cancelling where deterministic ones do not. Surveys of RQMC appear in Owen [31] and L'Ecuyer and Lemieux [22].

1.8 Padding and Latin Supercube Sampling

In some applications d is so large that it becomes problematic to construct a meaningful QMC sequence. For example the number of random vectors needed to follow a single light path in a scene with many reflective objects can be very large and may not have an a priori bound. As another example, if acceptance-rejection sampling (Devroye [8]) is used to generate a random variable then a large number of random variables may need to be generated in order to produce that variable.

Padding is a simple expedient solution to the problem. One uses a QMC or RQMC sequence in dimension s for what one expects are the s most important input variables. Then one pads out the input with $d - s$ independent $U[0, 1)$ random variables. This technique was used in Spanier [36] for particle transport simulations. It is also possible to pad with a $d - s$ dimensional Latin hypercube sample as described in Owen [29], even when d is conceptually infinite.

In Latin supercube sampling, the d input variables of x_i are partitioned into some number k of groups. The j'th group has dimension $d_j \ge 1$ and of course $\sum_{j=1}^{k} d_j = d$. A QMC or RQMC method is applied in each of the k groups.

Just as the van der Corput sequence cannot simply be substituted for a pseudo-random generator, care has to be taken in using multiple (R)QMC methods within the same problem. It would not work to take k independent randomizations of the same QMC sequence. The fix is to randomize the run order of the k groups relative to each other, just as Latin hypercube sampling randomizes the run order of d stratified samples.

To describe LSS, for $j = 1, \dots, k$ and $i = 1, \dots, n$ let $a_{ji} \in [0,1)^{d_j}$. Suppose that $a_{j1}, \dots, a_{jn}$ are a (R)QMC point set. For $j = 1, \dots, k$, let $\pi_j(i)$ be independent uniform permutations of $1, \dots, n$. Then let $x_{ji} = a_{j\pi_j(i)}$. The LSS has rows x_i comprised of $x_{1i}, \dots, x_{ki}$. Owen [29] shows that in Latin supercube sampling the function f can be written as a sum of two parts. One, from within groups of variables, is integrated with an (R)QMC error rate, while the other part, from between groups of variables, is integrated at the Monte Carlo rate. Thus a good grouping of variables is important as is a good choice of (R)QMC within groups.

Bibliography

[1] J. Beck and W. W. L. Chen. *Irregularities of Distribution*. Cambridge University Press, New York, 1987.

[2] Kenneth Chiu, Peter Shirley, and Changyaw Wang. Multi-jittered sampling. In Paul Heckbert, editor, *Graphics Gems IV*, pages 370–374. Academic Press, Boston, 1994.

[3] K.-L. Chung. An estimate concerning the Kolmogoroff limit distribution. *Transactions of the American Mathematical Society*, 67:36–50, 1949.

[4] William G. Cochran. *Sampling Techniques (3rd Ed)*. John Wiley & Sons, 1977.

[5] Robert L. Cook, Thomas Porter, and Loren Carpenter. Distributed ray tracing. *Computer Graphics*, 18(4):165–174, July 1984. ACM Siggraph '84 Conference Proceedings.

[6] R. Cranley and T.N.L. Patterson. Randomization of number theoretic methods for multiple integration. *SIAM Journal of Numerical Analysis*, 13:904–914, 1976.

[7] P. J. Davis and P. Rabinowitz. *Methods of Numerical Integration (2nd Ed.)*. Academic Press, San Diego, 1984.

[8] Luc Devroye. *Non-uniform Random Variate Generation*. Springer, 1986.

[9] Kai-Tai Fang and Yuan Wang. *Number Theoretic Methods in Statistics*. Chapman and Hall, London, 1994.

[10] Henri Faure. Discrépance de suites associées à un système de numération (en dimension s). *Acta Arithmetica*, 41:337–351, 1982.

[11] Henri Faure. Good permutations for extreme discrepancy. *Journal of Number Theory*, 42:47–56, 1992.

[12] G. Fishman. *Monte Carlo: Concepts, Algorithms, and Applications*. Springer-Verlag, 1995.

[13] S. Haber. A modified Monte Carlo quadrature. *Mathematics of Computation*, 20:361–368, 1966.

[14] J.H. Halton. On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. *Numerische Mathematik*, 2:84–90, 1960.

[15] F. J. Hickernell, H. S. Hong, P. L'Ecuyer, and C. Lemieux. Extensible lattice sequences for quasi-Monte Carlo quadrature. *SIAM Journal on Scientific Computing*, 22(3):1117–1138, 2000.

[16] E. Hlawka. Funktionen von beschränkter Variation in der Theorie der Gleichverteilung. *Annali di Matematica Pura ed Applicata*, 54:325–333, 1961.

[17] H. S. Hong and F. J. Hickernell. Implementing scrambled digital sequences. *AMS Transactions on Mathematical Software*, 2003. To appear.

[18] L.K. Hua and Y. Wang. *Applications of number theory to numerical analysis*. Springer, Berlin, 1981.

[19] Alexander Keller. A quasi-Monte Carlo algorithm for the global illumination problem in a radiosity setting. In Harald Niederreiter and Peter Jau-Shyong Shiue, editors, *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing*, pages 239–251, New York, 1995. Springer-Verlag.

[20] N. M. Korobov. The approximate computation of multiple integrals. *Dokl. Akad. Nauk SSSR*, 124:1207–1210, 1959.

[21] L. Kuipers and H. Niederreiter. *Uniform Distribution of Sequences*. John Wiley and Son, New York, 1976.

[22] P. L'Ecuyer and C. Lemieux. A survey of randomized quasi-Monte Carlo methods. In M. Dror, P. L'Ecuyer, and F. Szidarovszki, editors, *Modeling Uncertainty: An Examination of Stochastic Theory, Methods, and Applications*, pages 419–474. Kluwer Academic Publishers, 2002.

[23] J. Matoušek. On the L^2–discrepancy for anchored boxes. *Journal of Complexity*, 14:527–556, 1998.

[24] 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):239–45, 1979.

[25] Harald Niederreiter. Point sets and sequences with small discrepancy. *Monatshefte fur mathematik*, 104:273–337, 1987.

[26] Harald Niederreiter. *Random Number Generation and Quasi-Monte Carlo Methods*. S.I.A.M., Philadelphia, PA, 1992.

[27] A. B. Owen. Randomly permuted (t, m, s)-nets and (t, s)-sequences. In Harald Niederreiter and Peter Jau-Shyong Shiue, editors, *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing*, pages 299–317, New York, 1995. Springer-Verlag.

[28] A. B. Owen. Monte Carlo variance of scrambled equidistribution quadrature. *SIAM Journal of Numerical Analysis*, 34(5):1884–1910, 1997.

[29] A. B. Owen. Latin supercube sampling for very high dimensional simulations. *ACM Transactions on Modeling and Computer Simulation*, 8(2):71–102, 1998.

[30] Art B. Owen. Scrambling Sobol' and Niederreiter-Xing points. *Journal of Complexity*, 14(4):466–489, December 1998.

[31] Art B. Owen. Monte Carlo quasi-Monte Carlo and randomized quasi-Monte Carlo. In H. Niederreiter and J. Spanier, editors, *Monte Carlo and quasi-Monte Carlo Methods 1998*, pages 86–97, 1999.

[32] Ch. Schlier. A practitioner's view on qmc integration. *Mathematics and Computers in Simulation*, 2002.

[33] P. Shirley. Discrepancy as a quality measure for sample distributions. In Werner Purgathofer, editor, *Eurographics '91*, pages 183–194. North-Holland, September 1991.

[34] Ian H. Sloan and S. Joe. *Lattice Methods for Multiple Integration*. Oxford Science Publications, Oxford, 1994.

[35] I. M. Sobol'. The distribution of points in a cube and the accurate evaluation of integrals (in Russian). *Zh. Vychisl. Mat. i Mat. Phys.*, 7:784–802, 1967.

[36] J. Spanier. Quasi-Monte Carlo Methods for Particle Transport Problems. In Harald Niederreiter and Peter Jau-Shyong Shiue, editors, *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing*, pages 121–148, New York, 1995. Springer-Verlag.

[37] Michael Stein. Large sample properties of simulations using Latin hypercube sampling. *Technometrics*, 29(2):143–51, 1987.

[38] Boxin Tang. Orthogonal array-based Latin hypercubes. *Journal of the American Statistical Association*, 88:1392–1397, 1993.

[39] Bruno Tuffin. On the use of low discrepancy sequences in Monte Carlo methods. Technical Report 1060, I.R.I.S.A., Rennes, France, 1996.

[40] J. G. van der Corput. Verteilungsfunktionen I. *Nederl. Akad. Wetensch. Proc.*, 38:813–821, 1935.

[41] J. G. van der Corput. Verteilungsfunktionen II. *Nederl. Akad. Wetensch. Proc.*, 38:1058–1066, 1935.

[42] Eric Veach and Leonidas Guibas. Bidirectional estimators for light transport. In *5th Annual Eurographics Workshop on Rendering*, pages 147–162, June 13–15 1994.

[43] Eric Veach and Leonidas Guibas. Optimally combining sampling techniques for Monte Carlo rendering. In *SIGGRAPH '95 Conference Proceedings*, pages 419–428. Addison-Wesley, August 1995.

[44] H. Weyl. Über die gleichverteilung von zahlen mod. eins. *Mathematische Annalen*, 77:313–352, 1916.