

# Random and Deterministic Digit Permutations of the Halton Sequence\*

Giray Ökten, Manan Shah, Yevgeny Goncharov

*Department of Mathematics, Florida State University, Tallahassee FL 32312*

---

## Abstract

The Halton sequence is one of the classical low-discrepancy sequences. It is effectively used in numerical integration when the dimension is small, however, for larger dimensions, the uniformity of the sequence quickly degrades. As a remedy, generalized (scrambled) Halton sequences have been introduced by several researchers since the 1970s. In a generalized Halton sequence, the digits of the original Halton sequence are permuted using a carefully selected permutation. Some of the permutations in the literature are designed to minimize some measure of discrepancy, and some are obtained heuristically.

In this paper, we investigate how these carefully selected permutations differ from a permutation simply generated at random. We use a recent genetic algorithm, test problems from numerical integration and computational finance, and a recent randomized quasi-Monte Carlo method, to compare generalized Halton sequences with randomly chosen permutations, with the traditional generalized Halton sequences. Numerical results suggest that the random permutation approach is as good as, or better than, the "best" deterministic permutations.

*Key words:* Halton sequence, scrambled and generalized Halton sequences, Monte Carlo, quasi-Monte Carlo

---

## Introduction

The Halton sequences are arguably the best known and most studied low-discrepancy sequences. They are obtained from one-dimensional van der Corput sequences which have a simple and easy to implement definition. The  $n$ th term of the van der Corput sequence in base  $b$ , denoted by  $\phi_b(n)$ , is defined

---

\* This material is based upon work supported by the National Science Foundation under Grant No. DMS 0703849.

as follows: First, write  $n$  in its base  $b$  expansion:

$$n = (a_k \cdots a_1 a_0)_b = a_0 + a_1 b + \dots + a_k b^k,$$

then compute

$$\phi_b(n) = (0.a_0 a_1 \cdots a_k)_b = \frac{a_0}{b} + \frac{a_1}{b^2} + \dots + \frac{a_k}{b^{k+1}}. \quad (1)$$

The Halton sequence in the bases  $b_1, \dots, b_s$  is  $(\phi_{b_1}(n), \dots, \phi_{b_s}(n))_{n=1}^\infty$ . This is a uniformly distributed mod 1 (u.d. mod 1) sequence (see Niederreiter [11] for its definition) if the bases are relatively prime. In practice,  $b_i$  is usually chosen as the  $i$ th prime number.

One useful application of the Halton sequences (in general, low-discrepancy sequences) is to numerical integration. The celebrated Koksma-Hlawka inequality states,

**Theorem 1** *If  $f$  has bounded variation  $V(f)$  in the sense of Hardy and Krause over  $[0, 1]^s$ , then, for any  $x_1, \dots, x_N \in [0, 1]^s$ , we have*

$$\left| \frac{1}{N} \sum_{n=1}^N f(x_n) - \int_{[0,1]^s} f(x) dx \right| \leq V(f) D_N^*(x_i). \quad (2)$$

For the definition of bounded variation in the sense of Hardy and Krause, see Niederreiter [11]. The term  $D_N^*(x_i)$ , called the star-discrepancy of vectors  $x_1, \dots, x_N$  in  $[0, 1]^s$ , is defined as follows: For a subset  $S$  of  $[0, 1]^s$ , let  $A_N(S)$  be the number of vectors  $x_i$  that belong to  $S$ , and let  $\lambda(S)$  be the  $s$ -dimensional Lebesgue measure of  $S$ .

**Definition 2** *The star-discrepancy of vectors  $x_1, \dots, x_N \in [0, 1]^s$  is*

$$D_N^*(x_i) = \sup_S \left| \frac{A_N(S)}{N} - \lambda(S) \right|$$

where  $S$  is an  $s$ -dimensional interval of the form  $\prod_{i=1}^s [0, \alpha_i)$ , and the supremum is taken over the family of all such intervals. If the supremum is taken over intervals of the form  $\prod_{i=1}^s [\alpha_i, \beta_i)$ , then we obtain the so-called (extreme) discrepancy.

The star-discrepancy of the Halton sequence, or any low-discrepancy sequence, is  $O(N^{-1}(\log^s N))$ . This fact, together with the Koksma-Hlawka inequality, lay the foundation of the quasi-Monte Carlo integration.

There is a well-known defect of the Halton sequence: in higher dimensions when the base is larger, certain components of the sequence exhibit very poor

uniformity. This phenomenon is sometimes described as *high correlation between higher bases*. Figure 1, which plots 500 Halton vectors in bases 227 and 229 (corresponding to 49th and 50th prime numbers) illustrate this high correlation. Similar plots have been reported by several authors in the past.

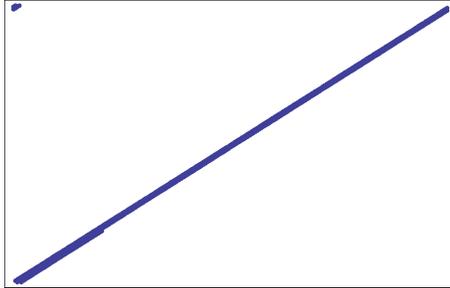


Fig. 1. 500 Halton vectors in bases 227 and 229

Observing this deficiency of the Halton sequence, Braaten & Weller [2] offered a remedy by generalizing the Halton sequence by using appropriately chosen permutations to scramble the digits in equation (1). Let  $\sigma_{b_i}$  be a permutation on the digit set  $\{0, \dots, b_i - 1\}$ , and generalize equation (1) as

$$\phi_{b_i}(n) = \frac{\sigma_{b_i}(a_0)}{b_i} + \frac{\sigma_{b_i}(a_1)}{b_i^2} + \dots + \frac{\sigma_{b_i}(a_k)}{b_i^{k+1}} \quad (3)$$

and define the Halton sequence in bases  $b_1, \dots, b_s$  as  $(\phi_{b_1}(n), \dots, \phi_{b_s}(n))_{n=1}^{\infty}$ . Halton sequences generalized in this way are called generalized Halton, or scrambled Halton sequences. Here we will use the term *digit permuted Halton sequences*.

Since the publication of Braaten and Weller [2], several authors introduced different permutations to scramble the digits of the Halton sequence; see, for example, [1], [3], [4], [5], [6], [9], [17], [18], [19]. Some of these permutations were obtained using heuristics, such as [9] & [18], and some others were obtained by searching for the optimal permutations that minimize the discrepancy of the one-dimensional or two-dimensional projections of the Halton sequence, such as [2], [3], [4], [5], and [6].

As we will elaborate further in Section 1, most authors cited above use a numerical approach to compare various digit permuted Halton sequences and we will follow the same methodology. Before we get into more details, let us entertain a simple question: Do these digit permuted Halton sequences avoid the phenomenon of *high correlation between higher bases* (see Figure 1), which was a defect of the Halton sequence? To answer this, we pick two permuted sequences, one by Faure [5], which was obtained by minimizing the discrepancy of one-dimensional projections, and the other by Kocis & Whiten [9], which was obtained heuristically. Both sequences were reported to be among the best

in recent numerical work (see [6]). Figure 2 plots 500 vectors in bases 1031 and 1033, for the Faure permutation, and, 191 and 193, for the Kocis & Whiten permutation. Note that 1033 is the 174th prime number and a dimension as large as 174 is not uncommon, for example, in financial applications.

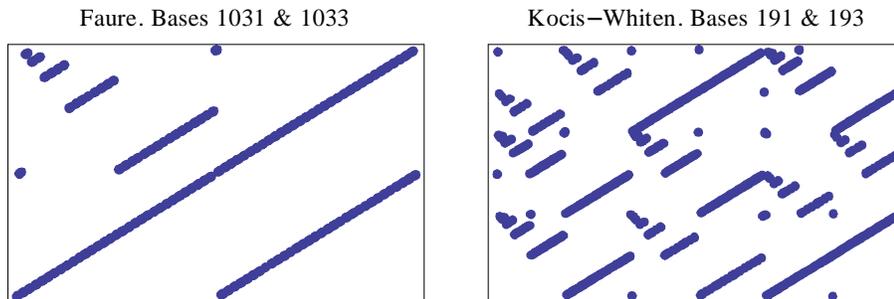


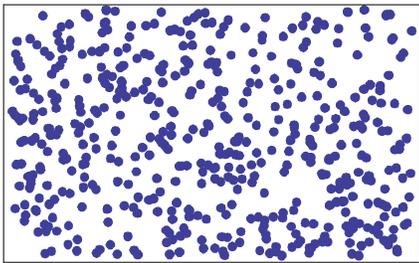
Fig. 2. 500 vectors from digit permuted Halton sequences

Figure 2 suggests that the digit permuted Halton sequences are also prone to the same deficiency of the Halton sequence. The bases used in the above plots were obtained by a computer search, and there are several other projections that have a similar behavior. In Section 3, we will go further than a visual inspection, and show how such large bases in digit permuted Halton sequences can result in high error in numerical integration. We will also compute approximations for the star-discrepancy of 10-dimensional permuted Halton sequences, and show how large bases result in high star-discrepancy. The algorithm we use to compute approximations for the star-discrepancy is a novel approach that uses genetic algorithms (see Shah [15]). The complexity of this algorithm grows much slower with the dimension of the sequence, compared to the existing algorithms.

In this paper we want to investigate the following question: What if we pick the permutation  $\sigma_{b_i}$  in equation (3), simply at random, from the space of all permutations? How would this approach, which we call *random digit permuted Halton sequence*, compare with the existing deterministic digit permuted Halton sequences? Perhaps a quick test for this idea would be to plot its vectors that correspond to the same bases we considered in Figure 2.

Inspecting Figure 3, we do not see a visual correlation we can speak of. Moreover, the same computer search program that we used to detect correlations in digit permuted Halton sequences did not detect similar correlations for any bases for the random digit permuted Halton sequence. On the other hand, one might wonder if these plots are too "pseudorandom like". The rest of the paper is devoted to comparing random digit permuted sequences with their deterministic counterparts, as well as pseudorandom numbers. We will compare these sequences by the exact error they produce in some test problems. We will also apply a randomized quasi-Monte Carlo method, called random start-

Random permutation. Bases 1031 &amp; 1033



Random permutation. Bases 191 &amp; 193

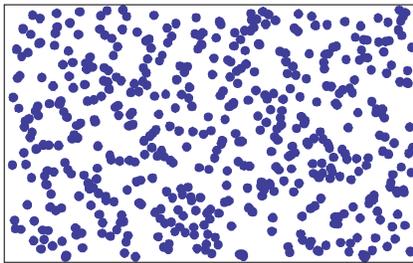


Fig. 3. 500 vectors from randomly permuted Halton sequences

ing, to these sequences. This randomization method enables us to compute unbiased estimates using independently randomized, digit permuted Halton sequences. We will discuss the details in Section 3.

Based on the numerical results we discuss in the paper, our final conclusion will be perhaps quite surprising: the random digit permuted Halton sequence is as good as the best deterministic digit permuted sequences, or better, in the test problems. In addition, we will consider a problem artificially designed to magnify the impact of the correlation between higher bases, and we will see that the random digit permutation approach is significantly better than the deterministic permutations for this problem.

## 1 Methodology

There are two main approaches to decide whether a given low-discrepancy sequence is better than another: theoretical, and empirical. The conventional theoretical approach computes upper bounds for the star-discrepancy of the sequences, and chooses the one with the smaller upper bound. The star discrepancy of  $N$  vectors of an  $s$ -dimensional low-discrepancy sequence is bounded by  $c_s(\log N)^s + O((\log N)^{s-1})$  where  $c_s$  is a constant that depends on the dimension  $s$ . The theoretical approach compares different sequences by their corresponding  $c_s$  values. There are two disadvantages of this approach. First, the upper bound for the star-discrepancy becomes very large as  $s$  and  $N$  get larger. Comparing the star-discrepancy of different sequences by comparing the upper bounds they satisfy becomes meaningless when these upper bounds are several orders of magnitude larger than the actual star-discrepancy.

The second disadvantage is that we do not know how tight the known bounds are for the constant  $c_s$ . For example, the Halton sequence used to be considered as the worst sequence among Faure, Sobol', Niederreiter, and Niederreiter-Xing sequences, based on the behavior of its  $c_s$  value. However, recent error bounds of Atanassov [1] imply significantly lower  $c_s$  values for the Halton

sequence. In fact, a special case of these upper bounds, which apply to a digit-permuted Halton sequence introduced by Atanassov [1], has lower  $c_s$  values than the Faure, Sobol', Niederreiter, and Niederreiter-Xing sequences. For details see Faure & Lemieux [6].

There are two empirical approaches used in the literature to compare low-discrepancy sequences. The first one is to apply the sequences to test problems with known solutions, and compare the sequences by the exact error they produce. The test problems are usually chosen from numerical integration, as well as various applications such as particle transport theory and computational finance. Numerical results are sometimes surprising. For example, even though the digit-permuted Halton sequence by Atanassov [1] has the best known bounds for its star-discrepancy, after extensive numerical results, Faure & Lemieux [6] conclude that several other digit-permuted sequences (Chi, Mascagni & Warnock [4], Faure & Lemieux [6], Kocis & Whiten [9]) are as good as the one by Atanassov [1].

The second empirical approach is to compute the discrepancy of the sequence numerically. The star-discrepancy is difficult to compute, but a variant of it, the  $L_2$ -discrepancy, is somewhat easier. In some papers, the  $L_2$ -discrepancy is used to compare different sequences. We will discuss a major drawback of this approach in the next section.

In this paper, we will use the empirical approach to compare various digit permuted Halton sequences including the random digit permutation approach. Since it is not very practical to compare *all* digit permuted sequences, we will proceed as follows: Following the conclusion of Faure & Lemieux [6], we will pick the Kocis-Whiten permutation as a representative of the permutations that performed equally well in their numerical results. We will also consider the permutation by Faure [5], which was used successfully in previous numerical studies of the authors (Goncharov, Ökten, Shah [8]), and the permutation by Braaten-Weller [2]. The standard Halton sequence, and the permutation by Vandewoestyne & Cools [18], will be included in the numerical results as benchmarks.

Our empirical approach has two components. We will compare the selected digit permuted sequences by computing an approximation to their star discrepancy, for some relatively small choices for sample size  $N$ , using a recent genetic algorithm developed by Shah [15]. For larger sample sizes, computing the discrepancy becomes intractable, and thus we will compare the sequences by the exact and statistical error they produce when applied to some test problems from numerical integration and computational finance with known solutions.

The test problem we will consider from numerical integration is estimating

the integral of

$$f(x_1, \dots, x_s) = \prod_{i=1}^s \frac{|4x_i - 2| + a_i}{1 + a_i} \tag{4}$$

in  $[0, 1]^s$ . This function was first considered by Radović, Sobol', Tichy [14], and used subsequently by several authors. The exact value of the integral is one, and the sensitivity of the function to  $x_i$  quickly decreases as  $a_i$  increases. The other two test problems we will consider are from computational finance: pricing European and ratchet options.

## 2 Computing the discrepancy

A modified version of the star-discrepancy, which is easier to compute, is the  $L_2$ -star discrepancy:

**Definition 3** *The  $L_2$ -star discrepancy of vectors  $x_1, \dots, x_N \in [0, 1]^s$  is*

$$T_N^*(x_i) = \left[ \int_{[0,1]^s} \left( \frac{A_N(S)}{N} - \lambda(S) \right)^2 d\alpha_1 \dots d\alpha_s \right]^{1/2}$$

where  $S = \prod_{i=1}^s [0, \alpha_i]$ .

Similarly, we can define the  $L_2$ -extreme discrepancy,  $T_N(x_i)$ , by replacing the sup norm in the definition of extreme discrepancy (Definition 2) by the  $L_2$ -norm. There are explicit formulas to compute  $T_N^*$  and  $T_N$  of a finite set of vectors. However, the formulas are ill-conditioned and they require high precision; see Vandewoestyne & Cools [18] for a discussion.

A more serious drawback of the  $L_2$ -discrepancies  $T_N^*$  and  $T_N$  is that they tend to give small values when the vectors are close to the vertex  $(1, 1, \dots, 1)$  of the  $s$ -dimensional cube. Moreover, as was observed by Matoušek [10], if  $N$  is small and  $s$  is large, the  $L_2$ -discrepancy of any point set is close to the best possible  $L_2$ -discrepancy.

We now discuss a recent example where the  $L_2$ -discrepancies give misleading results. In Vandewoestyne & Cools [18], a new permutation for the Halton sequence, called the reverse permutation, was introduced. The authors compared several digit permuted Halton sequences by their  $T_N^*$  and  $T_N$ , in dimensions that varied between 8 to 32. They considered at most  $N = 1000$  vectors in their computations. They concluded that the reverse permutation performed as good, or better, than the other permutations, in terms of the  $L_2$ -discrepancies. For example, Figure 9 on page 355 of [18] shows that  $T_N^*$  of the reverse permutation is much lower than the Braaten-Weller permutation,

as  $N$  varies between 1 and 1000. Table 1 displays  $T_N^*$  and  $D_N^*$  for these permutations, for  $N = 10, 100$ , and 200. The  $D_N^*$  values are computed using a genetic algorithm, which we will discuss in more detail later.

$N$	$T_N^*$		$D_N^*$	
	BW	Reverse	BW	Reverse
50	$13.5 \times 10^{-4}$	$2.00 \times 10^{-4}$	0.295	0.404
100	$7.01 \times 10^{-4}$	$1.77 \times 10^{-4}$	0.261	0.356
200	$3.64 \times 10^{-4}$	$1.53 \times 10^{-4}$	0.152	0.268

TABLE 1:  $T_N^*$  and  $D_N^*$

Observe that although  $T_N^*$  values for the reverse permutation are lower than the Braaten-Weller permutation for each  $N$ , exactly the opposite is true for  $D_N^*$ ! Which one of these results indicate a better sequence in terms of numerical integration? Next, we compare these sequences by comparing the exact error they produce when used to integrate the function  $f$  (see (4)). Table 2 displays the absolute error against the sample size  $N$ . The choices we make for  $N$  match the values used in Figure 9 of [18].

$N$	Reverse	BW	REV/BW
100	$434 \times 10^{-5}$	$34.1 \times 10^{-5}$	12.7
200	$138 \times 10^{-5}$	$13.7 \times 10^{-5}$	10.0
300	$47.4 \times 10^{-5}$	$44.2 \times 10^{-5}$	1.1
400	$113 \times 10^{-5}$	$7.28 \times 10^{-5}$	15.5
500	$18.2 \times 10^{-5}$	$18.0 \times 10^{-5}$	1.0
600	$17.2 \times 10^{-5}$	$38.8 \times 10^{-5}$	0.4
700	$66.5 \times 10^{-5}$	$9.84 \times 10^{-5}$	6.8
800	$37.2 \times 10^{-5}$	$11.4 \times 10^{-5}$	3.3
900	$8.93 \times 10^{-5}$	$8.89 \times 10^{-5}$	1.0
1000	$25.8 \times 10^{-5}$	$11.8 \times 10^{-5}$	2.2

TABLE 2: Integration error for  $f$

We observe that except for  $N = 600$ , the Braaten-Weller permutation error is less than or equal to the reverse permutation error. In fact, in almost all of the numerical results of this paper, the reverse permutation, together with the standard Halton sequence, gave the largest error among the digit permuted sequences.

## 2.1 Computing star-discrepancy using a genetic algorithm

Here we will discuss a recent genetic algorithm by Shah [15] that computes lower bounds for the star-discrepancy. The parameters of the algorithm were determined so that the algorithm provides good estimates for the star discrepancy when applied to two types of examples. The first type of examples included a small number of low-discrepancy vectors and dimension, so that the exact star-discrepancy could be computed using a brute force search algorithm. For example, the star-discrepancy of the first 50 vectors of the 5-dimensional Halton sequence was computed using a brute force search algorithm. Then the genetic algorithm was run, independently, forty times to obtain forty estimates (lower bounds) for the star-discrepancy. Thirty-eight of these estimates were in fact the exact discrepancy, and the remaining two were within 1.64% of the exact value.

The other type of examples Shah used to determine the algorithm parameters had larger number of vectors or dimension, and a brute force search was not practical. However, lower and upper bounds for the star-discrepancy could be computed using an algorithm by Thiémond [16]. Shah used the examples and the bounds given in [16], and was able to show that the genetic algorithm consistently yielded discrepancy estimates within Thiémond's bounds.

In the next two tables, we compute the star-discrepancy of the first 100 digit permuted Halton vectors,  $D_{100}^*$ , using the genetic algorithm. We consider the permutations by Vandewoestyne & Cools [18] (called reverse permutation), Faure [5], Kocis & Whiten [9], and the standard Halton sequence; these sequences are labeled as Reverse, Faure, KW, and Halton, respectively, in the tables. We want to compare these digit permuted sequences with our proposed random digit permuted sequences, with respect to their star-discrepancy. To do this, we generate forty random permutations independently, which gives forty random digit permuted Halton sequences. We then compute the star-discrepancy of the first 100 vectors of these sequences. The sample mean of the star-discrepancies of these forty sequences, together with a 95% bootstrap confidence interval, is reported in the last row of the tables.

In Table 3, there are three cases labeled as A, B, and C. In each case, we compute  $D_{100}^*$  when the dimension of the sequence is five, however, different cases use different bases. In A, the bases of the Halton sequence are the first five prime numbers;  $p_1, p_2, \dots, p_5$  ( $p_i$  is the  $i$ th prime number). In B, the bases are  $p_{14}, p_{20}, p_{27}, p_{33}, p_{39}$ , and in C the bases are  $p_{46}, p_{47}, p_{48}, p_{49}, p_{50}$ . We would

like to see how increasing the prime base affects the discrepancy.

$D_{100}^*$	Case A	Case B	Case C
Halton	0.110	0.601	0.961
Reverse	0.084	0.401	0.563
Faure	0.097	0.143	0.185
KW	0.100	0.149	0.124
Random	0.104	0.146	0.188
	(0.101, 0.106)	(0.142, 0.151)	(0.181, 0.196)

TABLE 3: Star-discrepancy for different bases. Dimension is five.

When the prime bases and the dimension (which is five) are low, as in Case A, we do not expect to see the standard Halton sequence have poor star-discrepancy, and the results support that. The star-discrepancy of Halton, KW, and Random are very close. Reverse and Faure are slightly better. In Case B, we increase the prime bases, in a mixed way, and the results change considerably. Now Halton is the worst, followed by Reverse. Permutations Faure, KW, and Random are in good agreement. Further increasing the bases in Case C spreads out the values; KW gives the lowest star-discrepancy, and Faure & Random come next.

In Table 5 we repeat a similar analysis, but now the problem is slightly more difficult: the dimension of the vectors is 10. In Case A, the bases are the first ten primes, and Halton and Reverse give the highest star-discrepancy. Faure and KW give the lowest values, followed by Random. In Case B, C, and D, the bases are the  $i$ th prime numbers where  $i \in \{11, 17, 21, 22, 24, 29, 31, 35, 37, 40\}$ ,  $i \in \{41, 42, 43, 44, 45, 46, 47, 48, 49, 50\}$ , and  $i \in \{43, 44, 49, 50, 76, 77, 135, 136, 173, 174\}$ . In all of these cases, Halton and Reverse give the highest star-discrepancy values, and the others remain close to each other. Note that the Random permutation approach gives confidence intervals that are lower than the star-discrepancy of the best deterministic digit permuted sequences in Cases B and C. In other words, a randomly permuted sequence has a lower star-discrepancy than the deterministic permutations we consider,

with a 95% probability, in these cases.

$D_{100}^*$	Case A	Case B	Case C	Case D
Halton	0.251	0.769	0.910	0.860
Reverse	0.244	0.429	0.485	0.903
Faure	0.157	0.238	0.209	0.360
KW	0.171	0.285	0.212	0.419
Random	0.182	0.212	0.259	0.294
	(0.177, 0.187)	(0.205, 0.220)	(0.253, 0.267)	(0.288, 0.303)

TABLE 4: Star-discrepancy for different bases. Dimension is ten.

### 3 Applications

In this section we compare deterministic and random digit permuted sequences when they are applied to the numerical integration of  $f$  (see (4)), and two problems from computational finance; pricing European call and ratchet options. In our numerical comparisons, we will use the following methods:

- (1) In all the problems we consider, exact solutions are known, and thus we can compare different sequences by the exact error they produce. Since it is not meaningful to compare a single occurrence of the random digit permutation approach with the deterministic sequences, we will follow the approach of the previous section, and compute a 95% bootstrap confidence interval using the estimates obtained from forty random digit permuted sequences. We will then examine how the confidence band compares with the deterministic sequences.
- (2) All digit permuted Halton sequences can be randomized by the random-start approach, which is a randomized quasi-Monte Carlo technique, as recently established by Ökten [12]. This enables us to compute the root mean square error of estimates obtained by independently "random-starting" a given digit permuted Halton sequence. For the random permutation approach, we will apply the random-start randomization to one realization of a random permuted Halton sequence.

#### 3.1 Numerical integration

Here we consider the numerical integration of the test function  $f(x_1, \dots, x_s) = \prod_{i=1}^s (|4x_i - 2| + a_i) / (1 + a_i)$ . The sensitivity of the function to  $x_i$  depends

inversely on the magnitude of the constant  $a_i$ . By appropriately choosing  $a_i$ , we can specify which components are more important, i.e., contribute more to the integral of the function. This enables us to test how well the underlying quasi-Monte Carlo sequence performs in different scenarios. For example, in Table 5, Case D, we observed that the confidence interval for the star-discrepancy for the random permutation approach was smaller than the deterministic digit permuted sequences. Case D corresponded to bases  $p_i$  where  $i \in D = \{43, 44, 49, 50, 76, 77, 135, 136, 173, 174\}$ . This result suggests that we might expect the random permutation approach perform better in a numerical integration problem where the function heavily depended on its variables from the index set  $D$ . The test function  $f$  helps us to verify this hypothesis easily: we set  $s = 174$ , and put  $a_i = 0$  if  $i \in D$ , and  $a_i = 1$  otherwise. Figure 4 plots the root mean square error (RMSE) of forty estimates when the Halton (HAL) sequence and the digit permuted sequences by Faure (FAU), Vandewoestyne & Cools (REV), Kocis & Whiten (KW), and Random (RND) are randomized via the random-start method.

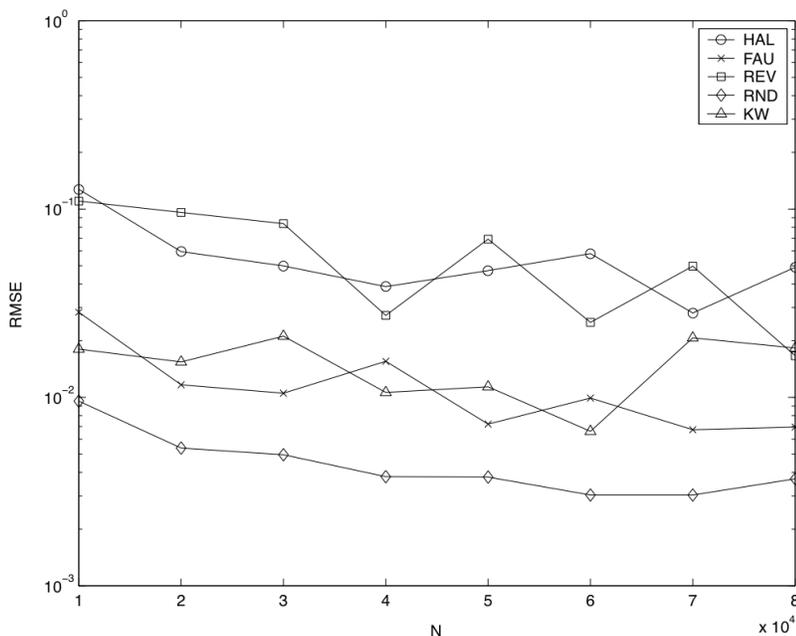


Fig. 4. Random digit permutation versus deterministic permutations

The random permutation approach gives the lowest RMSE consistent with the star-discrepancy estimates of Table 5. Faure & KW permutations and Halton & Reverse permutations give similar estimates, and the former is better, also consistent with the results of Table 5.

We now consider the test integrand when  $s = 10$ , and with three different scenarios for the importance of variables:  $a_i = 1$  (equal importance),  $a_i = i^2$  (decreasing importance as dimension grows), and  $a_i = (10 - i)^2$  (increasing

importance as dimension grows). In addition to the permutations considered in the previous figure, we include the Braaten-Weller (BW) permutation in these results. Figures 5, 6, 7 plot the actual estimate against the sample size  $N$ . The exact value of the integral is plotted as a horizontal line. The random permutation method is compared with the deterministic methods by the 95% confidence intervals, as we discussed earlier. The low and high limits for the confidence interval is denoted as RND Low and RND High, and plotted as dashed curves. The solid curve between the dashed curves represent the sample mean of the estimates.

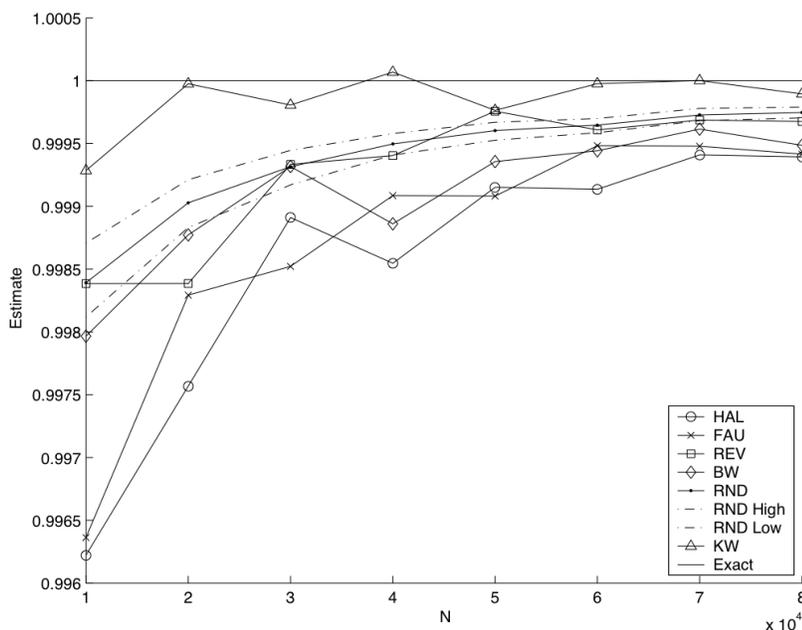


Fig. 5. Estimating the integral of  $f$  for the equal importance case ( $a_i = 1$ )

We make the following conclusions:

- (1) When all components have equal importance, the only deterministic method that gives better estimates than the random permutation confidence band consistently is the KW permutation. The overall behavior of the other deterministic methods is worse than the confidence band.
- (2) When the importance of the components quickly diminish ( $a_i = i^2$ ), making it essentially a low-dimensional problem, the overall behavior of estimates obtained from reverse, BW, and KW permutations, is better than the random permutation confidence band. Halton sequence, and Faure permutation, yields worse results, overall, than the confidence band.
- (3) When the importance of the components quickly increase ( $a_i = (10 - i)^2$ ), the confidence band provides a relatively monotonic convergence to the exact value. The BW and KW permutations stay very close to the confidence band for all samples. The Faure permutation, after 40,000

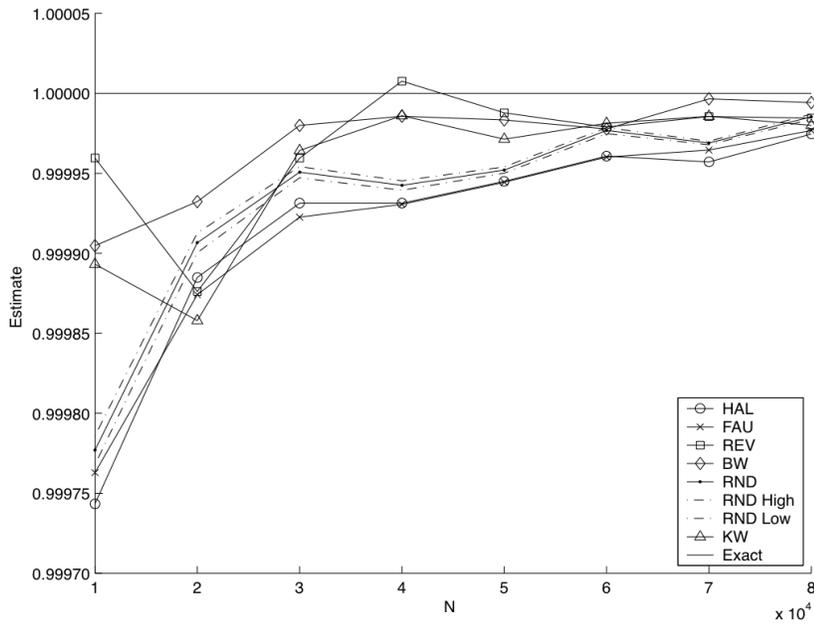


Fig. 6. Estimating the integral of  $f$  for the diminishing importance case ( $a_i = i^2$ )

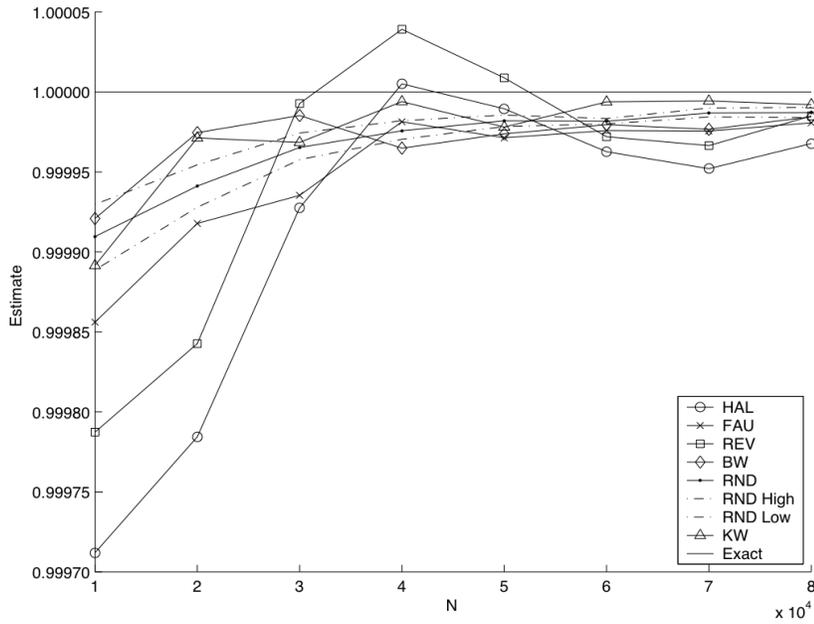


Fig. 7. Estimating the integral of  $f$  for the increasing importance case ( $a_i = (10-i)^2$ )

samples, stays close to the band as well. The estimates of the other two methods, Halton and reverse, show an erratic behavior, and except for a few samples, have larger error.

- (4) We did not include the Monte Carlo results in these figures since the Monte Carlo error was much worse than all the other sequences: the

RMSE of Monte Carlo is worse by factors between approximately 10 and 300 in the three cases we considered. We do not include Monte Carlo results in the rest of the section.

### 3.2 Pricing European options

Details of European options and the risk-free pricing by simulation can be found in Glasserman [7]. In the following, we consider a European call option under the lognormal model with the following parameters: exercise price is 40, volatility is 0.2, risk-free interest rate is 6%, and the expiry is 2 years. The European option can be solved as a one-dimensional problem, however, since we want to see the impact of high dimensions, we treat this as a path dependent option and simulate complete price paths with a uniform discretization of 100 steps. In other words, the dimension of the problem, and the underlying sequence in simulation, is 100. We consider three values for the initial stock price; 36, 40, and 44. For these choices, the option is called out-of-the money, at-the-money, and in-the-money. As the initial stock price increases, more of the price paths will contribute a non-zero value to the price of the option.

Figures 8, 9, 10 plot the root mean square error (RMSE) of forty estimates, when the random-start method is applied to the digit permuted sequences HAL, FAU, REV, RND, and KW. In all figures, the RMSE of HAL and REV are higher than the others approximately by factors between 10 and 20. In Figure 8, it is difficult to distinguish between FAU, RND, and KW. In Figures 9 & 10, the RMSE of RND is smaller than both FAU and KW in all except two samples, although the improvements are relatively small.

### 3.3 Pricing ratchet options

Details of ratchet (digital) options can be found in Papageorgiou [13]. Here we consider the following parameters: the initial stock price is 100, risk-free interest rate is 4.5%, the volatility is 0.3, and the expiry is 1. We consider a uniform discretization of 64 steps, i.e., the dimension of simulation is 64. Figure 11 plots the RMSE of the forty estimates obtained via the random-start method. The results are similar to the European option example. Halton (HAL) and Reverse (REV) have larger RMSE, by approximately about factors between 5 and 10. The random permutation (RND) has the lowest RMSE for all sample sizes except one.

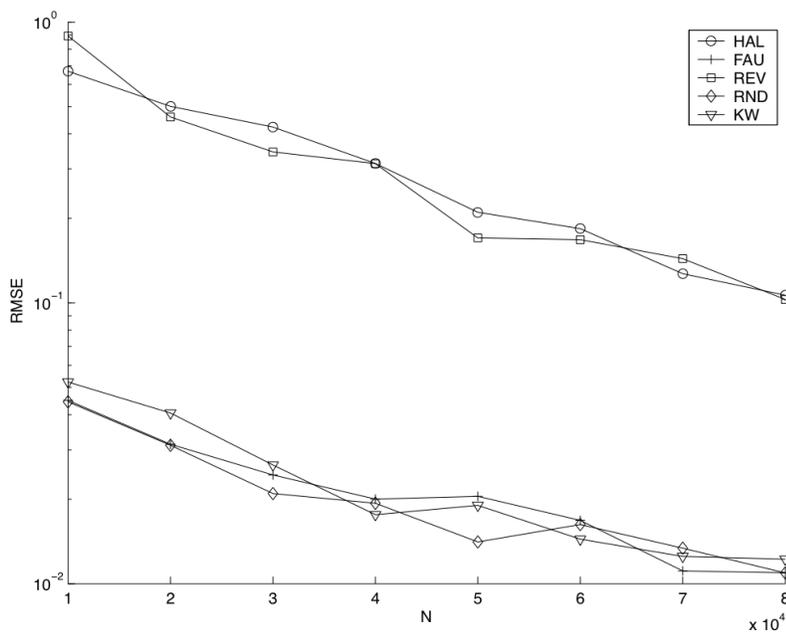


Fig. 8. Pricing an out-of-the-money European call option

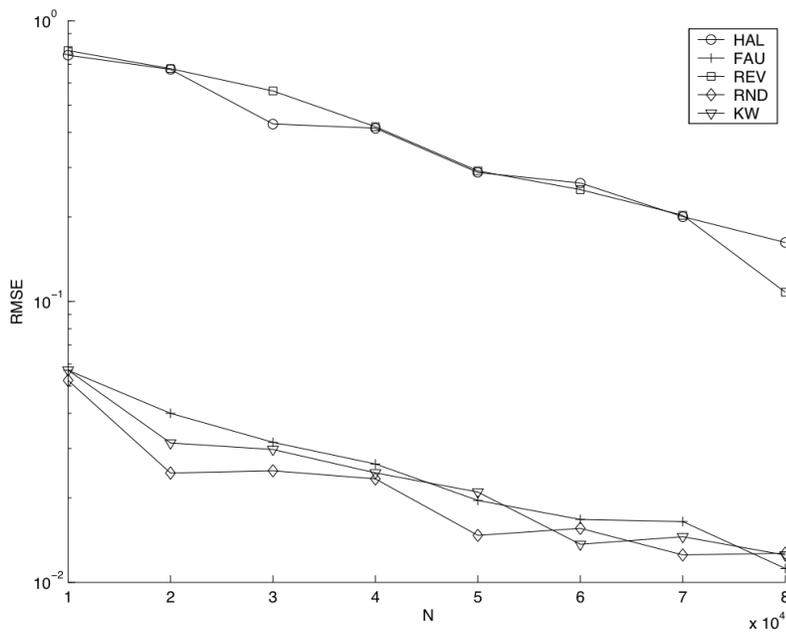


Fig. 9. Pricing an at-the-money European call option

## 4 Conclusions

Deterministic permutations designed to improve the uniformity of the Halton sequence have been around since the 1970s. Although various numerical exper-

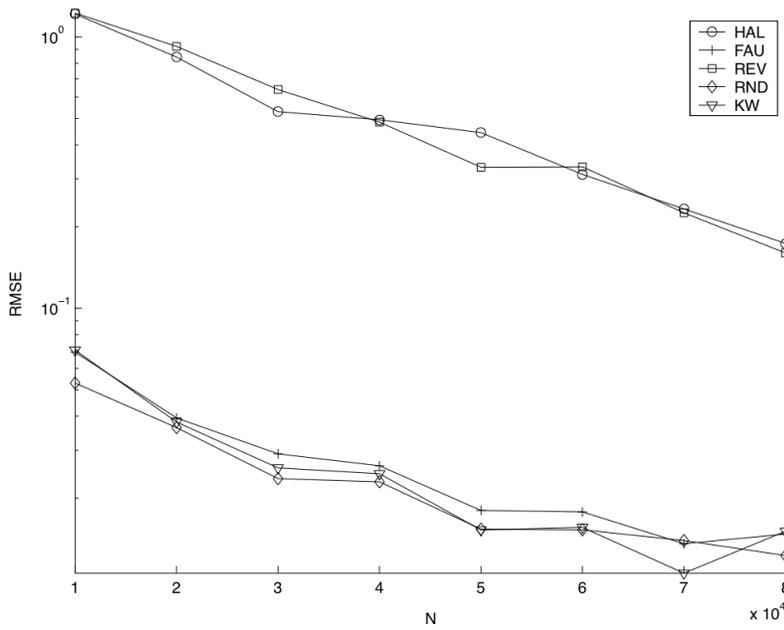


Fig. 10. Pricing an in-the-money European call option

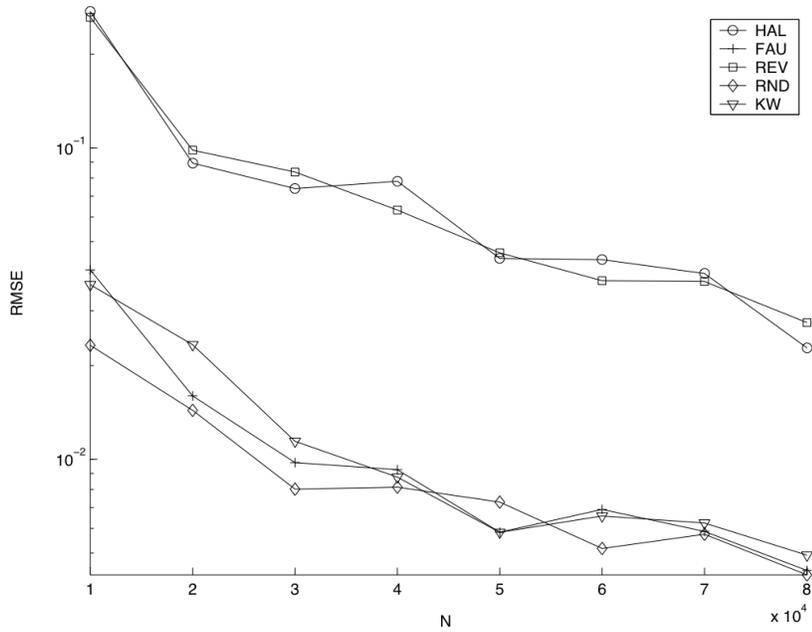


Fig. 11. Pricing a ratchet option

iments have been used to show the benefits of these sequences over the Halton sequence, the simple question of how such a sequence compares with a randomly permuted sequence has not been addressed in the literature. We used genetic algorithms and test problems from numerical integration & computational finance to compare randomly permuted Halton sequences with some

selected deterministic sequences. The comparison was made in two different ways; by applying the random-start method which enables the computation of mean square errors, and by constructing bootstrap confidence intervals. Quite surprisingly, the random permutation approach was as good as, or better, than the "best" deterministic permutations, in the problems we considered. We think further research and numerical studies are needed to understand the surprising success of this approach.

## References

- [1] E. I. Atanassov, On the discrepancy of the Halton sequences, *Math. Balkanica, New Series* **18** (2004) 15-32.
- [2] E. Braaten, G. Weller, An improved low-discrepancy sequence for multidimensional quasi-monte Carlo integration, *Journal of Computational Physics* **33** (1979) 249-258.
- [3] H. Chaix, H. Faure, Discrepance et diaphonie en dimension un, *Acta Arithmetica* LXIII (1993) 103–141.
- [4] H. Chi, M. Mascagni, T. Warnock, On the optimal Halton sequence, *Mathematics and Computers in Simulation* **70** (2005) 9-21.
- [5] H. Faure, Good Permutations for Extreme Discrepancy, *Journal of Number Theory* **42** (1992) 47–56.
- [6] H. Faure, C. Lemieux, Generalized Halton Sequences in 2007: A Comparative Study, Technical report, 2007.
- [7] P. Glasserman, *Monte Carlo Methods in Financial Engineering*, Springer-Verlag, 2004.
- [8] Y. Goncharov, G. Ökten, M. Shah, *Computation of the endogenous mortgage rates with randomized quasi-Monte Carlo simulations*, *Mathematical and Computer Modelling* **46** (2007) 459-481.
- [9] L. Kocis, W. J. Whiten, Computational investigations of low-discrepancy sequences, *ACM Transactions on Mathematical Software* **23** (1997) 266-294.
- [10] J. Matoušek, On the  $L_2$ -Discrepancy for Anchored Boxes, *Journal of Complexity* **14** (1998) 527-556.
- [11] H. Niederreiter. *Random Number Generation and Quasi-Monte Carlo Methods*, SIAM, Philadelphia, 1992.
- [12] G. Ökten, Generalized von Neumann-Kakutani transformation and random-start scrambled Halton sequences, *Journal of Complexity*, forthcoming (available online 11/24/08).

- [13] A. Papageorgiou, The Brownian Bridge Does Not Offer a Consistent Advantage in Quasi-Monte Carlo Integration, *Journal of Complexity* **18** (2002) 171-186.
- [14] I. Radović, I. M. Sobol', R. F. Tichy, Quasi-Monte Carlo Methods for Numerical Integration: Comparison of different low discrepancy sequences, *Monte Carlo Methods and Applications* **2** (1996) 1-14.
- [15] M. Shah, Quasi-Monte Carlo and Genetic Algorithms with Applications to Endogenous Mortgage Rate Computation, Ph.D. Dissertation, Department of Mathematics, Florida State University, 2008.
- [16] E. Thiémond, An Algorithm to Compute Bounds for the Star Discrepancy, *Journal of Complexity* **17** (2001) 850-880.
- [17] B. Tuffin, A new permutation choice in Halton sequences, in: Monte Carlo and Quasi-Monte Carlo 1996, Vol 127, Springer Verlag, New York, 1997, pp 427-435.
- [18] B. Vandewoestyne, R. Cools, Good permutations for deterministic scrambled Halton sequences in terms of  $L_2$ -discrepancy, *Journal of Computational and Applied Mathematics* **189** (2006) 341-361.
- [19] T. T. Warnock, Computational Investigations of low-discrepancy point sets II, in: Harald Niederreiter and Peter J. -S. Shiue, editors, Monte Carlo and quasi-Monte Carlo methods in scientific computing, Springer, New York, 1995, pp. 354-361.