

Weighted compound integration rules with higher order convergence for all N

Fred J. Hickernell, Peter Kritzer, Frances Y. Kuo & Dirk Nuyens

Numerical Algorithms

ISSN 1017-1398

Volume 59

Number 2

Numer Algor (2012) 59:161–183
DOI 10.1007/s11075-011-9482-5

VOLUME 38 (2005)
No. 4

This issue completes Vol. 38

ISSN 1017-1398

CODEN NUALEG

April 2005

NUMERICAL ALGORITHMS

Editor-in-Chief:
Claude Brezinski

 Springer

 Springer

Your article is protected by copyright and all rights are held exclusively by Springer Science+Business Media, LLC.. This e-offprint is for personal use only and shall not be self-archived in electronic repositories. If you wish to self-archive your work, please use the accepted author's version for posting to your own website or your institution's repository. You may further deposit the accepted author's version on a funder's repository at a funder's request, provided it is not made publicly available until 12 months after publication.

Weighted compound integration rules with higher order convergence for all N

Fred J. Hickernell · Peter Kritzer ·
Frances Y. Kuo · Dirk Nuyens

Received: 16 February 2011 / Accepted: 9 June 2011 /

Published online: 6 July 2011

© Springer Science+Business Media, LLC 2011

Abstract Quasi-Monte Carlo integration rules, which are equal-weight sample averages of function values, have been popular for approximating multivariate integrals due to their superior convergence rate of order close to $1/N$ or better, compared to the order $1/\sqrt{N}$ of simple Monte Carlo algorithms. For practical applications, it is desirable to be able to increase the total number of sampling points N one or several at a time until a desired accuracy is met, while keeping all existing evaluations. We show that although a convergence rate of order close to $1/N$ can be achieved for all values of N (e.g., by using a good lattice sequence), it is impossible to get better than order $1/N$ convergence for all values of N by adding equally-weighted sampling points in this manner. We then prove that a convergence of order $N^{-\alpha}$ for $\alpha > 1$ can be achieved

F. J. Hickernell

Department of Applied Mathematics, Illinois Institute of Technology,
Room E1-208, 10 West 32nd Street, Chicago, IL 60616, USA
e-mail: hickernell@iit.edu

P. Kritzer

Department of Financial Mathematics, University of Linz,
Altenbergerstr. 69, 4040 Linz, Austria
e-mail: peter.kritzer@jku.at

F. Y. Kuo

School of Mathematics and Statistics, University of New South Wales,
Sydney NSW 2052, Australia
e-mail: f.kuo@unsw.edu.au

D. Nuyens (✉)

Department of Computer Science, K.U.Leuven,
Celestijnenlaan 200A, 3001 Heverlee, Belgium
e-mail: dirk.nuyens@cs.kuleuven.be

by weighting the sampling points, that is, by using a weighted compound integration rule. We apply our theory to lattice sequences and present some numerical results. The same theory also applies to digital sequences.

Keywords Digital nets and sequences · Lattice rules and sequences · Higher order convergence · Numerical integration · Quasi-Monte Carlo

1 Introduction

Quasi-Monte Carlo (QMC) integration rules,

$$Q_{\text{qmc}}(f) = \frac{1}{N} \sum_{k=0}^{N-1} f(\mathbf{x}_k), \quad (1)$$

which are equally-weighted sample averages of function values, have been popular for approximating multivariate integrals $I(f) = \int_{[0,1]^s} f(\mathbf{x}) d\mathbf{x}$ over the s -dimensional unit cube $[0, 1]^s$. Their popularity is due to their simple form and their convergence rates that are superior to simple Monte Carlo algorithms, i.e., close to order $1/N$ versus $1/\sqrt{N}$. The key to superior convergence rates is the clever choice of the sampling points $\mathbf{x}_0, \dots, \mathbf{x}_{N-1}$. For integrands with higher smoothness, it is possible to obtain higher orders of convergence, however, there is a drawback that the choice of the points depends on knowing N in advance. As we shall explain in Section 2, it is impossible to have higher convergence than order $1/N$ for all values of N with a QMC rule that draws the sampling points from a single infinite sequence $\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \dots$.

For high orders of convergence one must either: (i) change sampling points when incrementing N , (ii) restrict oneself to a geometric progression of N , or (iii) weight the function values unequally in the integration rule. This article explores the third option. More precisely, we partition the N sampling points into M sets, with $N = \sum_{i=1}^M N_i$, to obtain M integration rules Q_1, \dots, Q_M , where each Q_i uses N_i points and has $O(N_i^{-\alpha})$ convergence rate with $\alpha > 1$. We consider a weighted compound rule of the form

$$\overline{Q}(f) = \sum_{i=1}^M w_i Q_i(f),$$

with weights w_i satisfying $\sum_{i=1}^M w_i = 1$. In Section 3 we prove that the special choice of weights

$$w_i = \frac{N_i^a}{N_1^a + \dots + N_M^a}, \quad a > 0, \quad (2)$$

can yield $O(N^{-\alpha})$ convergence rate for *all* values of N , provided that the convergence rates for the Q_i are $O(N_i^{-\alpha})$, M is small, and $a \geq \alpha$. We also derive the optimal choice of weights w_i^* for a given set of rules Q_i .

In Section 4 we apply the result of Section 3 to *lattice sequences*. More precisely, we introduce an algorithm which partitions the first N points of a lattice sequence into blocks based on the base b representation of N ; these blocks then form the rules Q_i . This strategy allows us to use a lattice sequence with a higher order of convergence by adding one point at a time in practice, updating the weights as we go, until we achieve a desired numerical accuracy. The same strategy also applies to digital sequences.

The effectiveness of our approach is demonstrated by numerical experiments in Section 5 where we take a simple test function as integrand as well as a function arising from option pricing. One focus of the experiments is on the choice of a in (2) with respect to the smoothness of the integrands. Another focus is on comparing the weights (2) with the optimal weights w_i^* . Our numerical results are encouraging and support our theory.

2 Worst case error for simple QMC sequences

In the following, we always assume that $\mathcal{H} = \mathcal{H}(K)$ is a *reproducing kernel Hilbert space* of s -variate functions over the unit cube $[0, 1]^s$, where $K : [0, 1]^s \times [0, 1]^s \rightarrow \mathbb{R}$ denotes the reproducing kernel. This means that we have $K(\cdot, \mathbf{y}) \in \mathcal{H}(K)$ for all $\mathbf{y} \in [0, 1]^s$ and $\langle f, K(\cdot, \mathbf{y}) \rangle_{\mathcal{H}} = f(\mathbf{y})$ for all $f \in \mathcal{H}$ and $\mathbf{y} \in [0, 1]^s$, where $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ denotes the inner product in \mathcal{H} . Later we shall make some additional assumptions about the function space to exclude trivial cases.

The *worst case error* of any (equal- or unequal-weight) integration rule Q in \mathcal{H} is defined by

$$e(Q, K) := \sup_{\|f\|_{\mathcal{H}} \leq 1} |I(f) - Q(f)|,$$

where $\|\cdot\|_{\mathcal{H}}$ denotes the norm in \mathcal{H} . In this article, we are mainly concerned with the question of how the worst case error behaves when we combine QMC rules based on different point sets. When a QMC rule (1) is based on a point set P , we sometimes write

$$Q_{\text{qmc}}(f) = \frac{1}{N} \sum_{\mathbf{x}_k \in P} f(\mathbf{x}_k),$$

and we also write $e(Q_{\text{qmc}}, K) = e(P, K)$ for the worst case error. Note that by “point set” we mean a multi-set, i.e., points may occur multiple times.

It is well known in the QMC literature (see, e.g., [20]) that there exist uniformly distributed sequences for which certain subsets with N points have a worst case error close to $O(N^{-\alpha})$, where $\alpha > 1$ is a smoothness parameter of the function space under consideration. However, such a result generally does *not* hold for arbitrary values of N , but is usually limited to N of a very specific form, for example, for N being powers of a fixed prime. The next proposition formalizes this assertion, and is based on an argument similar to the one in [24]. First we make an assumption about the function space \mathcal{H} .

Assumption 1 For the Hilbert space $\mathcal{H}(K)$ and any fixed number A , an A -point QMC rule using any point set P has a positive worst case error, and this error is bounded below away from zero no matter how well one chooses the set P , i.e.,

$$\inf_{\substack{P \subseteq [0,1]^s \\ |P|=A}} e(P, K) = \inf_{\substack{P \subseteq [0,1]^s \\ |P|=A}} \sup_{\|f\|_{\mathcal{H}} \leq 1} \left| I(f) - \frac{1}{A} \sum_{x_k \in P} f(x_k) \right| \geq c_{K,A} > 0,$$

where $c_{K,A}$ may naturally depend on K and A .

This assumption is not very restrictive. If it is violated then one can get an arbitrarily accurate approximation of the integral by evaluating the integrand at a finite number of points.

Proposition 1 Let $S = (x_k)_{k \geq 0}$ be any sequence of points in $[0, 1]^s$, and let S_N denote the first N points of S .

- If Assumption 1 holds for $A = 1$, then the sequence $(e(S_N, K))_{N \geq 1}$ cannot converge to zero faster than $O(N^{-1})$.
- If Assumption 1 holds for some $A \geq 1$, then the subsequence $(e(S_{N_0+nA}, K))_{n \geq 0}$ cannot converge to zero faster than $O(N^{-1})$, with N of the form $N = N_0 + nA$, for any fixed $N_0 \geq 1$.

Proof The proof for $A = 1$ and arbitrary A are the same. For any f in the unit ball of \mathcal{H} and $N \geq 1$, the signed error of the QMC rule using the point set S_N is

$$E_N(f) = I(f) - \frac{1}{N} \sum_{k=0}^{N-1} f(x_k).$$

Thus, for any positive integers N_0 , n and A , and $N = N_0 + nA$, we have

$$\begin{aligned} A \left| I(f) - \frac{1}{A} \sum_{k=N}^{N+A-1} f(x_k) \right| &= |(N+A)E_{N+A}(f) - NE_N(f)| \\ &\leq (N+A)|E_{N+A}(f)| + N|E_N(f)|, \end{aligned}$$

which leads to

$$A \sup_{\|f\|_{\mathcal{H}} \leq 1} \left| I(f) - \frac{1}{A} \sum_{k=N}^{N+A-1} f(x_k) \right| \leq (N+A)e(S_{N+A}, K) + Ne(S_N, K).$$

Suppose there exist constants $C > 0$ and $\beta > 0$ such that

$$e(S_N, K) \leq CN^{-\beta} \quad \text{for all } N = N_0 + nA \quad \text{with } n = 0, 1, \dots.$$

Then Assumption 1 implies that

$$A c_{K,A} \leq A \sup_{\|f\|_{\mathcal{H}} \leq 1} \left| I(f) - \frac{1}{A} \sum_{k=N}^{N+A-1} f(\mathbf{x}_k) \right| \leq \frac{C}{(N+A)^{\beta-1}} + \frac{C}{N^{\beta-1}}.$$

Since the left-hand side of the above inequality cannot converge to 0 as N increases, it is necessary that $\beta \leq 1$. \square

Proposition 1 shows that, for any sequence of points $S = (\mathbf{x}_k)_{k \geq 0}$, the sequence of worst case errors $(e(S_N, K))_{N \geq 1}$ cannot have a better convergence rate than $O(N^{-1})$. Moreover, subsequences indexed by arithmetic progressions also have no better convergence rate than $O(N^{-1})$. To obtain better convergence rates without indexing by geometric progression, it is necessary to relax the condition that every function value be weighted equally. This leads us to consider weighted compound rules.

3 Worst case error for weighted compound rules

Suppose we are given M integration rules Q_i , $1 \leq i \leq M$, where each Q_i uses N_i function evaluations. We define the weighted compound rule \bar{Q} by

$$\bar{Q}(f) := \sum_{i=1}^M w_i Q_i(f), \quad \text{with} \quad \sum_{i=1}^M w_i = 1. \quad (3)$$

The total number of function evaluations in \bar{Q} is $N := \sum_{i=1}^M N_i$. In particular, if each rule Q_i is a QMC rule and $w_i = N_i/N$, then \bar{Q} is a QMC rule with N points. Given the result from the previous section, we are especially interested in choosing w_i such that \bar{Q} is *not* a QMC rule, although the Q_i may well be.

It is well known (see, e.g., [12]) that the worst case error of an integration rule Q in a reproducing kernel Hilbert space $\mathcal{H}(K)$ equals the \mathcal{H} -norm of the error representer, which is the difference between the integral of the kernel and the approximation to this integral using Q . That is, $e(Q, K) = \|\xi_{Q,K}\|_{\mathcal{H}}$ with $\xi_{Q,K}(\mathbf{y}) := I(K(\cdot, \mathbf{y})) - Q(K(\cdot, \mathbf{y}))$. Thus for the compound rule (3) we have

$$\xi_{\bar{Q},K}(\mathbf{y}) = I(K(\cdot, \mathbf{y})) - \sum_{i=1}^M w_i Q_i(K(\cdot, \mathbf{y})) = \sum_{i=1}^M w_i \xi_{Q_i,K}(\mathbf{y}),$$

where $\xi_{Q_i,K}$ is the representer for the error of rule Q_i , and hence

$$e(\bar{Q}, K) = \left\| \sum_{i=1}^M w_i \xi_{Q_i,K} \right\|_{\mathcal{H}} = \left(\sum_{i,j=1}^M w_i w_j \langle \xi_{Q_i,K}, \xi_{Q_j,K} \rangle_{\mathcal{H}} \right)^{1/2}. \quad (4)$$

3.1 The optimal choice of weights

We can rewrite (4) as $e(\bar{Q}, K) = (\mathbf{w}^\top \mathbf{A} \mathbf{w})^{1/2}$, where \mathbf{w} is the column vector containing the weights w_1, \dots, w_M , and \mathbf{A} is the positive definite matrix

$$\mathbf{A} = (\langle \xi_{Q_i, K}, \xi_{Q_j, K} \rangle_{\mathcal{H}})_{i,j=1}^M. \quad (5)$$

The minimum of $\mathbf{w}^\top \mathbf{A} \mathbf{w}$ subject to the constraint on the weights that $\mathbf{1}^\top \mathbf{w} = 1$ is attained by

$$\mathbf{w}^* := \frac{\mathbf{A}^{-1} \mathbf{1}}{\mathbf{1}^\top \mathbf{A}^{-1} \mathbf{1}}, \quad (6)$$

that is, w_i^* is the sum of the entries in the i th row of \mathbf{A}^{-1} divided by the sum of all entries of \mathbf{A}^{-1} . The corresponding optimal compound rule \bar{Q}^* using these weights w_i^* has the worst case error

$$e(\bar{Q}^*, K) = \left(\frac{1}{\mathbf{1}^\top \mathbf{A}^{-1} \mathbf{1}} \right)^{1/2}. \quad (7)$$

This serves as a lower bound on the worst case error for a compound rule (3) with any choice of weights w_i . We note that (6) and (7) are the deterministic counterparts of the *unbiased minimum-variance linear estimator* employed in statistical analysis, see, e.g., [10].

The optimal choice of weights and the sharp lower bound on the worst case error require us to solve an $M \times M$ system whenever there is a change in the rules Q_i : we need to compute the matrix \mathbf{A} , solve $\mathbf{A}\mathbf{u} = \mathbf{1}$ for \mathbf{u} , and then set $\mathbf{w}^* = \mathbf{u}/\sum_i u_i$. Solving $\mathbf{A}\mathbf{u} = \mathbf{1}$ may be feasible if $M = O(\log N)$. However, evaluating each element $\langle \xi_{Q_i, K}, \xi_{Q_j, K} \rangle_{\mathcal{H}}$ of the matrix \mathbf{A} may require as much as $O(N_i N_j)$ operations (except for some special circumstances which we shall discuss later). Thus using the optimal choice of weights w_i^* might not be a practical approach to obtain a compound rule.

3.2 A simple upper bound for arbitrary weights

Using the triangle inequality, we can bound (4) from above by

$$e(\bar{Q}, K) \leq \sum_{i=1}^M |w_i| \|\xi_{Q_i, K}\|_{\mathcal{H}} = \sum_{i=1}^M |w_i| e(Q_i, K). \quad (8)$$

This provides a simple, albeit loose, error bound for the compound rule \bar{Q} when we already have error estimates for the rules Q_i . It follows immediately from (8) that if $w_i = N_i/N$ and each rule Q_i achieves close to $O(N_i^{-1})$ convergence, then the compound rule \bar{Q} achieves close to $O(N^{-1})$ convergence, provided that M is small compared to N , e.g., $M = O(\log N)$. In particular, this applies to the special case where the compound rule \bar{Q} is a QMC sequence and the rules Q_i are obtained by partitioning \bar{Q} in a certain way (we will discuss

this in Section 4). This is similar to the classical result [16, Theorem 2.6, p. 115] which is expressed in terms of “discrepancy”.

Minimizing the right-hand side of (8) corresponds to choosing $w_k = 1$ for the rule Q_k with the smallest worst case error and choosing $w_i = 0$ for all $i \neq k$. This would seem to suggest that a compound rule consisting of just one (best) rule Q_k is better than a compound rule which combines all of Q_i . This is *not* the right conclusion. One needs to bear in mind that (8) is merely an upper bound, a crude one indeed, to the true worst case error for the compound rule \bar{Q} . There could be many cancelations in the worst case errors when the rules Q_i are combined, but such cancelation effect is lost in the estimate (8).

3.3 A special choice of weights

Here we propose a special choice of weights and derive the corresponding worst case error bound by making use of the simple error estimate (8).

Theorem 1 Consider the weighted compound rule (3) with weights

$$w_i = \frac{N_i^a}{N_1^a + \dots + N_M^a} \quad \text{for some } a > 0.$$

Suppose there exist constants $C > 0$ and $\alpha > 0$ such that

$$e(Q_i, K) \leq C N_i^{-\alpha} \quad \text{for all } i = 1, 2, \dots, M.$$

Then we have

$$e(\bar{Q}, K) \leq C \frac{M^{\max(\min(\alpha, a), 1)}}{N^{\min(\alpha, a)}}.$$

Proof It follows from (8) that

$$e(\bar{Q}, K) \leq \sum_{i=1}^M w_i e(Q_i, K) \leq C \frac{\sum_{i=1}^M N_i^{a-\alpha}}{\sum_{i=1}^M N_i^a}. \quad (9)$$

To obtain further estimates on (9) we shall make use of two inequalities. The first one is often (incorrectly) referred to as Jensen's inequality (see [11, 15]):

$$\sum_{i=1}^M y_i \leq \left(\sum_{i=1}^M y_i^p \right)^{1/p} \quad \text{for } 0 \leq p \leq 1.$$

The second one is a generalized mean inequality (see [1]):

$$\frac{1}{M} \sum_{i=1}^M y_i \leq \left(\frac{1}{M} \sum_{i=1}^M y_i^p \right)^{1/p} \quad \text{for } 1 \leq p < \infty.$$

In both inequalities we assume that all y_i are non-negative. Note that Jensen's inequality is tight when all but one of the y_i are zero, and the generalized mean inequality is tight when all the y_i are equal. Therefore our estimate will not be sharp, unless $M = 1$.

Consider first the denominator in (9). Choose $y_i = N_i$ and $p = a$. If $0 < a \leq 1$ then Jensen's inequality implies that $\sum_{i=1}^M N_i^a \geq (\sum_{i=1}^M N_i)^a = N^a$. On the other hand, if $a \geq 1$ then the generalized mean inequality yields $(\sum_{i=1}^M N_i^a / M)^{1/a} \geq \sum_{i=1}^M N_i / M = N / M$, implying $\sum_{i=1}^M N_i^a \geq N^a / M^{a-1}$. In summary, we have

$$\sum_{i=1}^M N_i^a \geq N^a M^{-\max(a-1,0)}. \quad (10)$$

Next, consider the numerator in (9). For $a \leq \alpha$ it follows that $N_i^{a-\alpha} \leq 1$ for all N_i , and so this numerator is bounded by M . Substituting this and (10) into (9) yields

$$e(\bar{Q}, K) \leq C \frac{M}{N^a M^{-\max(a-1,0)}} = C \frac{M^{\max(a,1)}}{N^a} \quad \text{for } a \leq \alpha. \quad (11)$$

For the case of $a > \alpha$ the generalized mean inequality is applied with $y_i = N_i^{a-\alpha}$ and $p = a/(a-\alpha)$ to yield $\sum_{i=1}^M N_i^{a-\alpha} / M \leq (\sum_{i=1}^M N_i^a / M)^{(a-\alpha)/a}$, implying $\sum_{i=1}^M N_i^{a-\alpha} \leq M^{\alpha/a} (\sum_{i=1}^M N_i^a)^{(a-\alpha)/a}$. Substituting this into (9) and then applying the estimate (10) yields

$$\begin{aligned} e(\bar{Q}, K) &\leq C \frac{M^{\alpha/a} \left(\sum_{i=1}^M N_i^a \right)^{(a-\alpha)/a}}{\sum_{i=1}^M N_i^a} = C \frac{M^{\alpha/a}}{\left(\sum_{i=1}^M N_i^a \right)^{\alpha/a}} \\ &\leq C \frac{M^{\alpha/a}}{\left(N^a M^{-\max(a-1,0)} \right)^{\alpha/a}} = C \frac{M^{\max(\alpha, \alpha/a)}}{N^\alpha} \leq C \frac{M^{\max(\alpha, 1)}}{N^\alpha} \quad \text{for } \alpha \leq a. \end{aligned}$$

Combining this result with (11) completes the proof of the theorem. \square

Note that as $a \rightarrow \infty$ the special choice of weights in Theorem 1 approaches the weights that minimize the right hand side of (8). For the application of Theorem 1 we have in mind the situation where $M = O(N^\epsilon)$ for ϵ positive and arbitrarily close to 0. This implies that $e(\bar{Q}, K) \leq CN^{-\min(\alpha, a)+\epsilon}$. In the next section we partition the first N points of a lattice sequence to form the rules Q_i based on the base b representation of N ; in this case $M = O(\log N)$.

4 Weighted compound rules built from lattice sequences

4.1 Background on lattice rules and lattice sequences

An s -dimensional N -point lattice rule (of rank 1) is uniquely specified by its generating vector $\mathbf{z} \in \mathbb{Z}^s$: the lattice points are given by

$$\mathbf{x}_k = \left\{ \frac{k \mathbf{z}}{N} \right\}, \quad \text{for } k = 0, 1, \dots, N-1, \quad (12)$$

where the braces around a vector indicate that we take the fractional part of each component in the vector. Traditionally lattice rules are defined with a fixed number of points N and a fixed dimension s , see [20, 26]. By now it is well known that good generating vectors can be constructed *component-by-component* (CBC), that is, the components of $\mathbf{z} = (z_1, z_2, \dots)$ are chosen one at a time, while holding previously chosen components fixed, see [27]. The criterion used in the CBC construction is the worst case error in some reproducing kernel Hilbert space. In particular, it is shown that the resulting lattice rules can achieve the optimal rate of convergence $O(N^{-\alpha+\delta})$, $\delta > 0$, in the periodic *weighted Korobov spaces* with smoothness parameter $\alpha > 1/2$, where α is roughly the number of derivatives available, see [5, 17]. (Note that the smoothness parameter α for the Korobov space is often also taken as $\alpha' = 2\alpha$, to denote the decay of the Fourier coefficients.) The fast CBC algorithm introduced in [21, 22] has a cost of only $O(s N \log N)$, which means that in practice we may obtain generating vectors with a huge number of points in very high dimensions. Moreover, the nature of the CBC construction means that the dimension s can be “extended” at a later time as we wish. We remark that to be able to use lattice rules in a non-periodic setting, e.g., in *weighted Sobolev spaces*, we need to use *shifted* lattice points, which means that we add the same vector $\Delta \in [0, 1]^s$ to every lattice point, and if any point falls outside of the unit cube then it is “wrapped” in from the other side, i.e., the shifted lattice points are given by

$$\{\mathbf{x}_k + \Delta\}, \quad k = 0, 1, \dots \quad (13)$$

We shall not go into further details here. See [18] for a story of recent developments on the construction of lattice rules.

For practical reasons it would be nice to have a lattice point set that is “extensible” in N as well as in s (the latter is already achieved by the CBC construction). In [14] the existence of infinite lattice sequences is shown, however no constructive approach is presented there. To obtain a lattice sequence we must change the way the points are indexed in (12) so that N is no longer part of the definition. Following [14], we define the points of a lattice sequence in base b by

$$\mathbf{x}_k = \{\varphi_b(k)\mathbf{z}\}, \quad \text{for } k = 0, 1, \dots, \quad (14)$$

where the index k of the point is first mapped into $[0, 1)$ by the function φ_b , which, for any non-negative integers m and q , satisfies the condition

$$\left\{ \varphi_b(k) : qb^m \leq k < (q+1)b^m \right\} = \left\{ \left\{ \frac{k}{b^m} + \Delta \right\} : 0 \leq k < b^m \right\} \quad (15)$$

for some $\Delta \in [0, 1)$ dependent on b , m and q . This condition enforces that each block of the point set of size b^m , which is aligned at a multiple of b^m , is a shifted version of the equally spaced points $\{kb^{-m} : 0 \leq k < b^m\}$. Typically, we have $\Delta = 0$ for $q = 0$, and this means that $\mathbf{x}_0 = \mathbf{0}$ and that the definition (14) yields the full lattice rule as defined by (12) at every power of the base b .

Most commonly, one chooses for φ_b the *radical inverse* function or the radical inverse of the Gray code, both in the base of the sequence. The radical inverse function in base b maps a non-negative integer k with base b representation $\sum_{\ell \geq 0} k_\ell b^\ell$, $k_\ell \in \{0, \dots, b-1\}$ for all $\ell \geq 0$, to the real number $\sum_{\ell \geq 0} k_\ell b^{-\ell-1}$.

We see from (14) that, in theory, the number of points could now be infinite. We denote this infinite lattice sequence by $S = (\mathbf{x}_k)_{k \geq 0}$ and the initial portion of N points by S_N . Moreover, we use the notation $S_N + \Delta$ for the point set obtained by applying a shift Δ to every point in S_N , see (13).

The theoretical guarantee of a good lattice sequence is that the worst case errors converge with a certain speed when N is of the form b^m for any m . For instance, in weighted Korobov spaces with reproducing kernel K and smoothness parameter $\alpha > 1/2$, we have

$$e(S_N, K) \leq C N^{-\alpha+\delta} \quad \text{for all } N = b^m \quad \text{with } m = 0, 1, \dots,$$

for some constants C and $\delta > 0$, see [14]. This is what we referred to earlier as “indexing by geometric progression”. We do not know the rate of convergence for the values of N in between powers of b . In fact, Proposition 1 indicates that it is impossible to have a convergence rate better than order $1/N$ if we weight the points equally. In the next subsection we will make use of Theorem 1 to get a good rate of convergence when N is *not* of the form b^m .

Note that contrary to the existence results in [14], a practical algorithm for obtaining a lattice sequence might not construct an infinite sequence, but an *embedded* sequence having a maximum number of points b^{m_2} , in some base b , where all initial portions of size b^m are optimal for $m_1 \leq m \leq m_2$, or for an even more restricted set. Some practical algorithms can be found in [4, 7, 9, 13]. The paper [7] contains a proof on the optimality of the construction presented there, and in the same paper [7, Theorem 13] this is also shown to hold for the algorithm in [4]. Both papers [4, 7] make use of the fast CBC algorithm [21, 22].

To proceed we need to make one further simplifying assumption that the reproducing kernel K of the Hilbert space $\mathcal{H}(K)$ is *shift-invariant*, i.e.,

$$K(\mathbf{x}, \mathbf{y}) = K(\{\mathbf{x} - \mathbf{y}\}, \mathbf{0}) \quad \text{for all } \mathbf{x}, \mathbf{y} \in [0, 1]^s.$$

It then follows that the worst case error of a shifted point set equals the worst case error of the original point set. In particular, we have

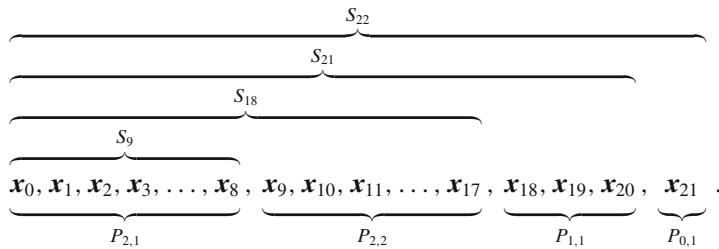
$$e(S_N + \Delta, K) = e(S_N, K) \quad \text{for all } N \geq 1 \quad \text{and } \Delta \in [0, 1]^s.$$

Note that this shift-invariant assumption on the kernel is *not* really a limitation. When a given reproducing kernel is *not* shift-invariant, one can obtain an *associated* shift-invariant kernel by “averaging” over all shifts. For the full details, see, e.g., [12].

4.2 Partitioning a lattice sequence

Before we present the main theorem of this section, we first consider a simple example. Suppose we want to make use of the first $N = 22$ points of a lattice sequence $S = (\mathbf{x}_k)_{k \geq 0}$ in base $b = 3$. We now partition S_{22} into smaller sets each

with cardinality 3^ℓ for some ℓ . Expressing 22 in base 3, we have $22 = (211)_3 = 2 \times 3^2 + 1 \times 3^1 + 1 \times 3^0$, indicating that we get 2 sets with 9 points each in level 2, plus 1 set with 3 points in level 1, plus 1 set with just 1 point in level 0. This can be illustrated as follows:



Thus we have partitioned S_{22} into $P_{2,1}$, $P_{2,2}$, $P_{1,1}$, $P_{0,1}$, and we may form the compound rule as in Theorem 1 with $M = 4$ using these four point sets as four QMC rules.

More generally, if $N = \sum_{\ell=0}^{L-1} n_\ell b^\ell$ in base b representation, then there are L levels in total, indexed from 0, and there are n_ℓ sets in level ℓ , with each set containing b^ℓ points. For each set $P_{\ell,j}$, the first subscript ℓ represents the level it is in, and the second subscript j enumerates the sets within the same level. Two key observations are needed before we can apply Theorem 1:

1. Due to the condition (15) and how the lattice sequence is defined, see (14), the point set $P_{\ell,j}$ is a shifted version of S_{b^ℓ} , i.e.,

$$P_{\ell,j} = S_{b^\ell} + \Delta_{\ell,j} \quad \text{for some } \Delta_{\ell,j} \in [0, 1)^s.$$

2. Since K is shift-invariant, the worst case error of $P_{\ell,j}$ is the same as the worst case error of S_{b^ℓ} , i.e.,

$$e(P_{\ell,j}, K) = e(S_{b^\ell}, K).$$

In our example above, the point set $P_{2,2}$ is a shifted version of S_9 , $P_{1,1}$ is a shifted version of S_3 , and $P_{0,1}$ is a shifted version of S_1 . Moreover, we have $e(P_{2,2}, K) = e(S_9, K)$, $e(P_{1,1}, K) = e(S_3, K)$, and $e(P_{0,1}, K) = e(S_1, K)$.

We are now ready to formalize the main result of this section.

Theorem 2 Let $S = (x_k)_{k \geq 0}$ be a lattice sequence in base b for which the worst case error in a shift-invariant reproducing kernel Hilbert space $\mathcal{H}(K)$ satisfies

$$e(S_{b^\ell}, K) \leq C (b^\ell)^{-\alpha} \quad \text{for all } \ell = 0, 1, \dots,$$

for some constants $C, \alpha > 0$. Given any N with base b representation

$$N = \sum_{\ell=0}^{L-1} n_\ell b^\ell, \quad L := \lfloor \log_b N \rfloor + 1, \quad n_{L-1} > 0, \quad 0 \leq n_\ell < b \quad \text{for all } \ell,$$

we define, for each $\ell = L-1, \dots, 0$ and $j = 1, \dots, n_\ell$, the QMC rule

$$Q_{\ell,j}(f) := \frac{1}{b^\ell} \sum_{k=T_{\ell,j}}^{T_{\ell,j}+b^\ell-1} f(\mathbf{x}_k), \quad \text{with} \quad T_{\ell,j} := \sum_{r=\ell+1}^{L-1} n_r b^r + (j-1)b^\ell. \quad (16)$$

Then for any $a > 0$, the compound rule defined by

$$\overline{Q}(f) := \sum_{\ell=0}^{L-1} \sum_{j=1}^{n_\ell} w_{\ell,j} Q_{\ell,j}(f), \quad (17)$$

with weights (which are the same for rules in the same level)

$$w_{\ell,j} = w_\ell := \frac{(b^\ell)^a}{\sum_{\ell'=0}^{L-1} \sum_{j=1}^{n_{\ell'}} (b^{\ell'})^a} = \frac{(b^\ell)^a}{\sum_{\ell'=0}^{L-1} n_{\ell'} (b^{\ell'})^a},$$

has the worst case error satisfying

$$e(\overline{Q}, K) \leq C \frac{((\lfloor \log_b N \rfloor + 1)(b-1))^{\max(\min(\alpha, a), 1)}}{N^{\min(\alpha, a)}}.$$

Proof We see from (17) that \overline{Q} is made up of M QMC rules $Q_{\ell,j}$, with

$$M = \sum_{\ell=0}^{L-1} n_\ell \leq L(b-1) = (\lfloor \log_b N \rfloor + 1)(b-1).$$

Moreover, the weights of these M rules $Q_{\ell,j}$ are defined as in Theorem 1 with a parameter a . To invoke Theorem 1, it suffices that we determine an error bound for $e(Q_{\ell,j}, K)$ for each rule $Q_{\ell,j}$.

For each $\ell = L-1, \dots, 0$ and $j = 1, \dots, n_\ell$, let $P_{\ell,j} := \{\mathbf{x}_k \in S : T_{\ell,j} \leq k < T_{\ell,j} + b^\ell\}$ denote the point set of the QMC rule $Q_{\ell,j}$. We observe that $P_{\ell,j} = S_{b^\ell} + \Delta_{\ell,j}$ for some shift $\Delta_{\ell,j} \in [0, 1]^s$. Since the reproducing kernel is shift-invariant, we conclude that

$$e(Q_{\ell,j}, K) = e(P_{\ell,j}, K) = e(S_{b^\ell} + \Delta_{\ell,j}, K) = e(S_{b^\ell}, K) \leq C (b^\ell)^{-\alpha}.$$

We may now invoke Theorem 1 and obtain the desired result. \square

The essence of Theorem 2 is that by using the special choice of weights for the lattice sequence, one can get nearly the $O(N^{-\alpha})$ convergence rate for all values of N , and not just for those values of N being powers of the base b .

The weighted compound rule (17), with subrules (16) built from the lattice sequence, might look rather complicated but can be implemented without too much effort. Indeed, it is sufficient to use a table which contains the weights for each level ℓ in terms of the digit expansion of N , as well as the accumulated sum of function values $Q_{\ell,j}(f)$ using b^ℓ points. When N increases, some accumulated sums might have to be changed to the correct level, but there is no need to redo any calculation. This table only has a memory requirement of $O(\log N)$ and can be fixed beforehand for the maximal value of N . Moreover,

Algorithm 1 Weighted QMC algorithm in base 2

```

1: Input: integrand function  $f$ , QMC sequence  $(\mathbf{x}_k)_{k \geq 0}$  in base 2, parameter
    $a$ , maximum number of levels maxlevels
2: Output: compound rule approximation  $\bar{Q}$ 

3: Initialize  $N := 0$ ,  $Q_\ell := 0$ ,  $w_\ell := (2^\ell)^a$  for  $\ell := 0, \dots, \text{maxlevels} - 1$ 
4: while estimated error still too large do
5:    $Q_0 := Q_0 + f(\mathbf{x}_N)$                                 {draw next point and evaluate}
6:   mask :=  $N$  AND  $(N \text{ XOR } (N + 1))$       {work out which sums to carry}
7:   for  $\ell := 0$  to  $\text{maxlevels} - 2$  do
8:     if bit  $\ell$  set in mask then
9:        $Q_{\ell+1} := Q_{\ell+1} + Q_\ell/2$ , and  $Q_\ell := 0$       {carry the sums through}
10:    end if
11:   end for
12:    $N := N + 1$ 
13:    $\bar{Q} := (\sum_\ell w_\ell Q_\ell)/(\sum_\ell n_\ell w_\ell)$ , where  $n_\ell$  is the  $\ell$ th bit of  $N$       {calculate approximation}
14: end while
15: return  $\bar{Q}$ 

```

we can handle multiple values of the parameter a at the same time with a very small additional cost.

A pseudocode for base $b = 2$ (presumably the most interesting case in practice) is given as Algorithm 1. In base 2, there is at most one point set in level ℓ , and whether or not this point set is present can be trivially determined by testing if the ℓ th bit n_ℓ of $N = \sum_{\ell=0}^{L-1} n_\ell 2^\ell$ is set, see line 13. Furthermore, one can use bit instructions to determine which accumulated sums have to be updated and carried through as is done in line 6. The input parameter `maxlevels` corresponds to the maximal value of $L = \lfloor \log_2 N \rfloor + 1$. This is only included to simplify the presentation of the pseudocode, and is not a constraint on the algorithm. It should be easy to see that `maxlevels` can be dynamically adjusted in the while loop as N increases. The code for base $b \geq 3$ can be derived by extending this pseudocode, however, we then loose the advantage of fast bit operations.

Theorem 2 is formulated based on a lattice sequence for which the worst case error of the first b^m points is optimal for all $m \geq 0$. In a practical construction of a lattice sequence one might choose to only satisfy this criterion for a limited set of values of m . For example, m is restricted to $0 \leq m_1 \leq m \leq m_2$ for the constructions in [4, 7]. When $m_1 \neq 0$, we need to adjust Theorem 2 and Algorithm 1 to consider only N which are multiples of b^{m_1} . (However, since the construction can be done very quickly, there is no reason not to choose $m_1 = 0$.)

As a corollary to Theorem 2, in the special case of $a = 1$ we obtain an error bound which holds for all values of N for the “plain” lattice sequence (i.e., all

the points in a sequence of length N have equal weights $1/N$). The same result can also be obtained directly from the inequality (8).

Corollary 1 Let $S = (\mathbf{x}_k)_{k \geq 0}$ be a lattice sequence in base b for which the worst case error in a shift-invariant reproducing kernel Hilbert space $\mathcal{H}(K)$ satisfies

$$e(S_{b^\ell}, K) \leq C (b^\ell)^{-\alpha} \quad \text{for all } \ell = 0, 1, \dots,$$

for some constants $C, \alpha > 0$. Then for every $N \geq 2$, the quasi-Monte Carlo rule using the first N points of the lattice sequence has the worst case error satisfying

$$e(S_N, K) \leq C \frac{\log_b N}{N^{\min(\alpha, 1)}}.$$

The result indicates that when α is close to 1, we can obtain close to $O(1/N)$ convergence rate for all values of N using just equal weights. We recall from Proposition 1 that a convergence rate of order $1/N$ is the best rate possible using equal weights. If $\alpha > 1$ and one desires to have close to $O(N^{-\alpha})$ convergence then it is necessary to use special weights such as the choice in Theorem 2.

4.3 Compound lattice rules with optimal weights

We can partition a lattice sequence as described in the previous subsection and use the optimal weights (6) to obtain the optimal compound lattice rule \overline{Q}^* . Then we have $e(\overline{Q}^*, K) \leq e(\overline{Q}, K)$, and the error bound in Theorem 2 also applies to \overline{Q}^* . Since \overline{Q}^* has a smaller worst case error but \overline{Q} has much simpler weights, it would be interesting to see how \overline{Q}^* compares to \overline{Q} in practice.

Recall that obtaining the optimal weights (6) requires solving a linear system with the $M \times M$ matrix \mathbf{A} given by (5), but evaluating the elements of \mathbf{A} may require $O(N_i N_j)$ operations. For shift-invariant kernels together with lattice sequences, the latter can be done in $O(\max(N_i, N_j))$ operations, which we explain below.

Proposition 2 Let $S = (\mathbf{x}_k)_{k \geq 0}$ be a lattice sequence in base b and let $\mathcal{H}(K)$ be a shift-invariant reproducing kernel Hilbert space. For any $N \geq 1$ we partition the first N points of S as in Section 4.2, keeping the ordering of the points, and labeling the resulting segments as P_1, P_2, \dots . Then the (i, j) th element $a_{i,j}$ of the matrix \mathbf{A} given by (5) can be expressed as

$$a_{i,j} = \begin{cases} - \int_{[0,1]^s} K(\mathbf{x}, \mathbf{0}) d\mathbf{x} + \frac{1}{N_i} \sum_{k=0}^{N_i-1} K(\{\mathbf{x}_k + \Delta_{i,j}\}, \mathbf{0}) & \text{if } N_i \geq N_j, \\ a_{j,i} & \text{otherwise,} \end{cases}$$

where $\Delta_{i,j}$ is the difference between the first point of P_i and the first point of P_j . Therefore, assuming that the integral over $K(\mathbf{x}, \mathbf{0})$ is known explicitly, the element $a_{i,j}$ can be calculated in $O(\max(N_i, N_j))$ operations.

Proof Using properties of the reproducing kernel, the (i, j) th element of \mathbf{A} reduces to

$$\begin{aligned} \langle \xi_{Q_i, K}, \xi_{Q_j, K} \rangle_{\mathcal{H}} &= \int_{[0,1]^s} K(\mathbf{x}, \mathbf{y}) d\mathbf{x} d\mathbf{y} - \frac{1}{N_i} \sum_{\mathbf{x}_k \in P_i} \int_{[0,1]^s} K(\mathbf{x}_k, \mathbf{x}) d\mathbf{x} \\ &\quad - \frac{1}{N_j} \sum_{\mathbf{x}_k \in P_j} \int_{[0,1]^s} K(\mathbf{x}_k, \mathbf{x}) d\mathbf{x} + \frac{1}{N_i N_j} \sum_{\mathbf{x}_k \in P_i} \sum_{\mathbf{x}_{k'} \in P_j} K(\mathbf{x}_k, \mathbf{x}_{k'}) \\ &= - \int_{[0,1]^s} K(\mathbf{x}, \mathbf{0}) d\mathbf{x} + \frac{1}{N_i N_j} \sum_{\mathbf{x}_k \in P_i} \sum_{\mathbf{x}_{k'} \in P_j} K(\{\mathbf{x}_k - \mathbf{x}_{k'}\}, \mathbf{0}), \end{aligned}$$

where we invoked the shift-invariant property of the kernel. Next we use the fact that the point sets P_i are obtained from a partition of the lattice sequence as described in Section 4.2, with $\mathbf{x}_0 = \mathbf{0}$. (Note that in Section 4.2 we used the notation $P_{\ell, j}$ to describe the j th point set in level ℓ . Here we do not follow that notation.) Due to the structure of the lattice sequence, see (14) and (15), given P_i and P_j we can find q, m, q' , and m' such that $N_i = b^m$, $N_j = b^{m'}$, and

$$P_i = \{\mathbf{x}_k : qb^m \leq k < (q+1)b^m\} = S_{b^m} + \Delta_{q,m},$$

$$P_j = \{\mathbf{x}_{k'} : q'b^{m'} \leq k' < (q'+1)b^{m'}\} = S_{b^{m'}} + \Delta_{q',m'},$$

where the shifts are given explicitly by

$$\Delta_{q,m} = \{\mathbf{x}_{qb^m} - \mathbf{x}_0\} = \mathbf{x}_{qb^m} \quad \text{and} \quad \Delta_{q',m'} = \{\mathbf{x}_{q'b^{m'}} - \mathbf{x}_0\} = \mathbf{x}_{q'b^{m'}}.$$

Assuming that $b^{m'} \leq b^m$, we have

$$S_{b^m} = \bigcup_{t=0}^{b^{m-m'}-1} (S_{b^{m'}} + \Delta_t), \quad \text{with} \quad \Delta_t = \{\mathbf{x}_{tb^{m'}} - \mathbf{x}_0\} = \mathbf{x}_{tb^{m'}}. \quad (18)$$

Thus the double sum can be written as

$$\begin{aligned} &\frac{1}{N_i N_j} \sum_{\mathbf{x}_k \in P_i} \sum_{\mathbf{x}_{k'} \in P_j} K(\{\mathbf{x}_k - \mathbf{x}_{k'}\}, \mathbf{0}) \\ &= \frac{1}{b^m b^{m'}} \sum_{k=0}^{b^m-1} \sum_{k'=0}^{b^{m'}-1} K(\{\mathbf{x}_k - \mathbf{x}_{k'} + \mathbf{x}_{qb^m} - \mathbf{x}_{q'b^{m'}}\}, \mathbf{0}) \\ &= \frac{1}{b^m b^{m'}} \sum_{t=0}^{b^{m-m'}-1} \sum_{k=0}^{b^{m'-1}-1} \sum_{k'=0}^{b^{m'}-1} K(\{\mathbf{x}_k - \mathbf{x}_{k'} + \mathbf{x}_{qb^m} - \mathbf{x}_{q'b^{m'}} + \mathbf{x}_{tb^{m'}}\}, \mathbf{0}) \\ &= \frac{1}{b^m} \sum_{t=0}^{b^{m-m'}-1} \sum_{k=0}^{b^{m'-1}-1} K(\{\mathbf{x}_k + \mathbf{x}_{qb^m} - \mathbf{x}_{q'b^{m'}} + \mathbf{x}_{tb^{m'}}\}, \mathbf{0}) \\ &= \frac{1}{b^m} \sum_{k=0}^{b^m-1} K(\{\mathbf{x}_k + \mathbf{x}_{qb^m} - \mathbf{x}_{q'b^{m'}}\}, \mathbf{0}), \end{aligned}$$

where we used the fact that $\{x_k - x_{k'}\}$ is a lattice point if both x_k and $x_{k'}$ are, and (18) was used in both directions. This last expression is of order $b^m = \max(N_i, N_j)$ as claimed. \square

Some straightforward modifications in Algorithm 1 allow one to use the optimal weights with respect to the worst case error, but one has to solve the linear system for each new function evaluation (or each new addition of points). The code to calculate these optimal weights can then be inserted just before line 13. The rest of the algorithm can stay the same. A similar modification can be made for any other choice of weights to be used with compound rules based on the partitioning from Section 4.2.

4.4 Remarks

We remark that the same technique as outlined for lattice sequences may also be used to obtain results for higher order digital (t, s) -sequences, see, e.g., [2, 6]. In this case, the assumption that the kernel of the function space be shift-invariant needs to be replaced by the assumption that the kernel be *digitally shift-invariant*. If the space does not have this property then one can obtain an associated digitally shift-invariant kernel, as in the case of lattice sequences. Under these conditions, all the theory presented in this paper is applicable.

Further, there are known algorithms which achieve higher order convergence based on the so-called *Kronecker sequences*. For example, Niederreiter [19] proposed some rather complicated weights in terms of sums over binomial coefficients, and Sugihara and Murota [28] proposed a slightly easier form of weights; see [29, 30] for efficient implementations of these algorithms. The strategy proposed in this article complements these known algorithms for lattice sequences and digital sequences and is relatively easy to implement.

5 Numerical experiments

We construct a lattice sequence using the fast component-by-component algorithm from [4] with the following parameters: a (unweighted) Korobov space with $\alpha = 3$ (which corresponds to $\alpha = 6$ in [4]), base $b = 2$, $m_1 = 0$ and $m_2 = 20$ (i.e., $N = 2^m$ for $0 \leq m \leq 20$), and dimension $s = 10$. This yields the generating vector

$$\mathbf{z} = (1, 364981, 245389, 97823, 488939, 62609, 400749, 385317, 21281, 223487).$$

(Note that this rule was constructed in multiple precision since, e.g., the squared worst case error for the 4-dimensional rule is approximately 5.914×10^{-20} .)

We now carry out three experiments using Algorithm 1, with the weights from Theorem 2, and employing the aforementioned lattice sequence.

5.1 A simple test integrand

We use the weighted compound rule with optimal weights and with parameters $a = 1, 2, \dots, 6$ for the simple test function

$$f_3(\mathbf{x}) := \prod_{j=1}^{10} \left(1 + \left(x_j^3 - \frac{3}{2}x_j^2 + \frac{1}{2}x_j \right) \right),$$

whose integral is $I(f_3) = 1$. Here $x^3 - \frac{3}{2}x^2 + \frac{1}{2}x$ is the Bernoulli polynomial of degree 3. For this test function we expect an order of convergence close to $N^{-\min(a,3)}$.

In Fig. 1 we plot the absolute errors for the six different values of a and for the optimal weights. The top graph shows the errors as N increases from 3 to 10^6 , with all values of a superimposed. (The errors for using 1 and 2 points are zero and thus are omitted.) From Theorem 2 it follows that the errors for all values of a are the same when N is an exact power of the base. Therefore the powers of 2 have been marked on the graph with small discs. The order of convergence of these points is $O(N^{-3})$ as expected. The bottom six graphs in Fig. 1 show the errors for the individual values of a . The dashed lines indicate the orders of convergence, which are $O(N^{-1})$, $O(N^{-2})$, $O(N^{-3})$, $O(N^{-3})$, $O(N^{-3})$ and $O(N^{-3})$, respectively. This is in complete agreement with our theoretical expectation. On the graph for $a = 1$ we observe a wide band between a convergence of $O(N^{-1})$ and $O(N^{-3})$. The effect of increasing a is narrowing the band to $O(N^{-3})$. Furthermore, as a increases it takes longer to enter the asymptotic regime. This confirms the presence of the $(\log N)^{\min(a,3)}$ factor in our error bound.

5.2 An integrand arising from option pricing

Next we consider an option pricing problem where the financial product is a look-back option. This is a product where the payoff is determined by the maximum value of the underlying asset (e.g., a stock) as follows:

$$\max(\max(S_1, S_2, \dots, S_s) - K, 0),$$

where S_i denotes the price of the asset at the discrete monitoring time t_i , and K is the strike price. Without going into the details, we simply say here that we choose the following parameters: initial asset price $S_0 = 100$, strike price $K = 120$, risk-free interest rate $r = 0.1$, volatility $\sigma = 0.4$, time $T = 5$, and $s = 5$ equally-spaced time steps (i.e., yearly monitoring). For this payoff one can obtain an “analytic” expression for the value of the option in terms of a sum over cumulative multivariate normal distributions, see [3]. These multivariate integrals then have to be approximated numerically. One of the integrals appearing in this expression is

$$\int_{-\infty}^{-d_2(t_1)} \int_{-\infty}^{-d_2(t_2)} \int_{-\infty}^{-d_2(t_3)} \int_{-\infty}^{-d_2(t_4)} \int_{-\infty}^{-d_2(t_5)} \frac{\exp\left(-\frac{1}{2}\mathbf{x}^\top \boldsymbol{\Sigma}^{-1} \mathbf{x}\right)}{\sqrt{|\boldsymbol{\Sigma}|(2\pi)^5}} d\mathbf{x},$$

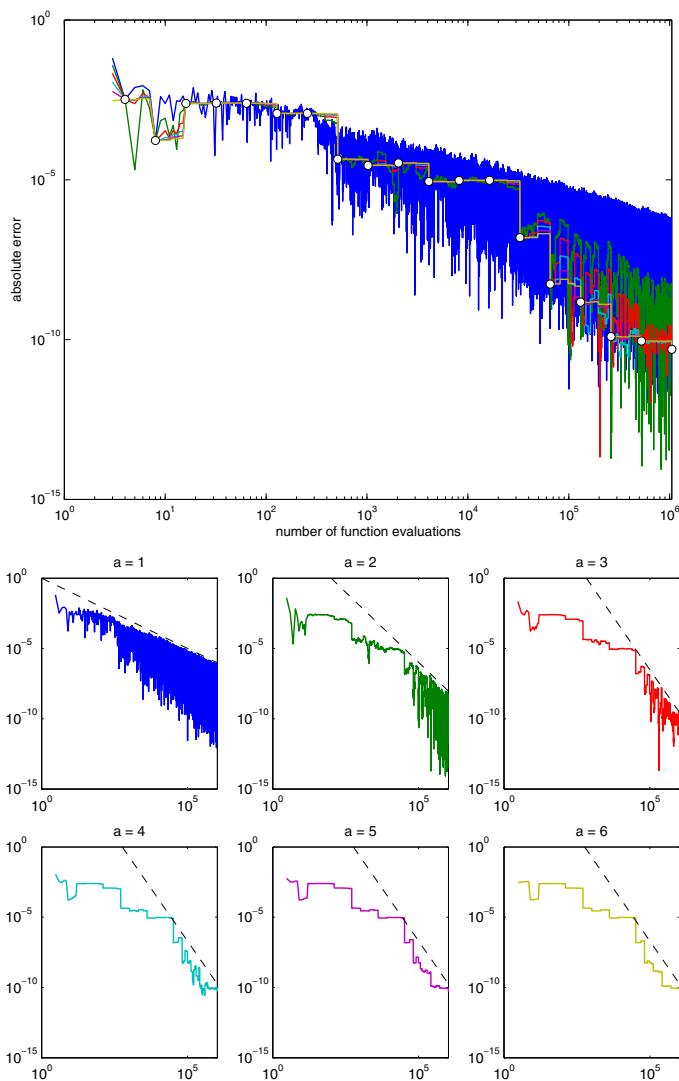


Fig. 1 The error for $I(f_3)$ in 10 dimensions with $a = 1, 2, \dots, 6$. The dashed lines in the bottom six graphs correspond to $O(N^{-1})$, $O(N^{-2})$, $O(N^{-3})$, $O(N^{-3})$, $O(N^{-3})$ and $O(N^{-3})$, respectively

where

$$d_1(t_i) = \frac{\ln(S_0/K) + (r + \sigma^2/2)t_i}{\sigma\sqrt{t_i}}, \quad d_2(t_i) = d_1(t_i) - \sigma\sqrt{t_i}, \quad \Sigma_{jk} = \sqrt{\frac{t_{\min(j,k)}}{t_{\max(j,k)}}}.$$

We use an approach by Genz [8] to calculate this 5-dimensional integral (the dimensionality is reduced to $s = 4$ due to the method in [8]), in combination

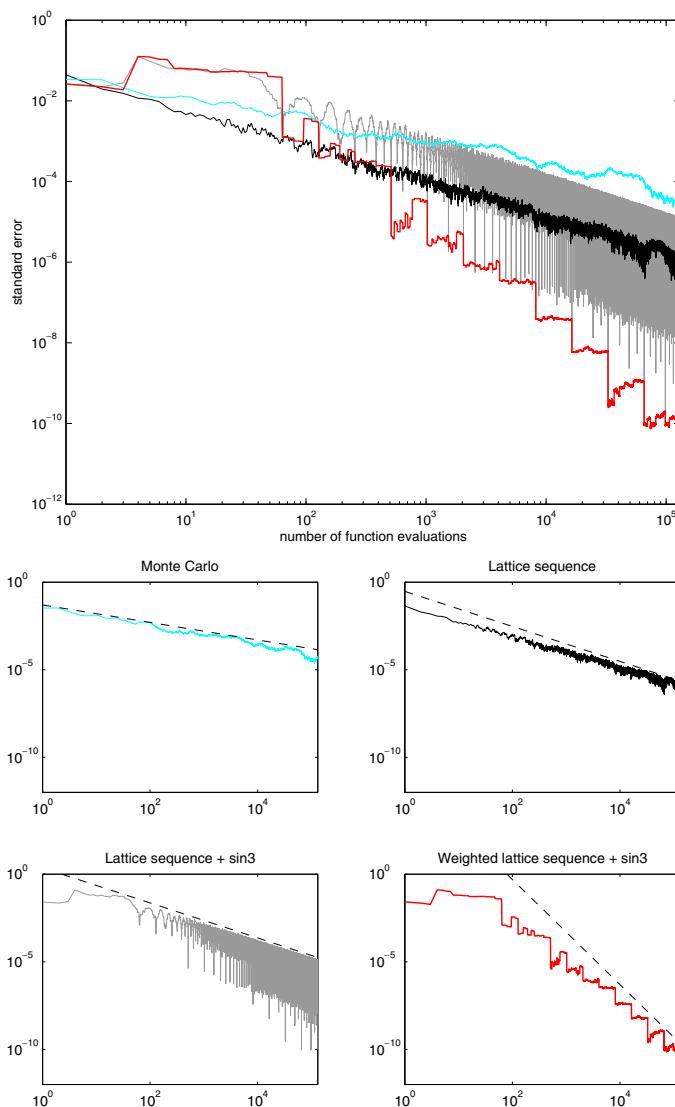


Fig. 2 The error for computing a multivariate normal distribution in 4 dimensions arising from the evaluation of a look-back option with 5 time steps. The *dashed lines* in the bottom four graphs correspond to $O(N^{-1/2})$, $O(N^{-1})$, $O(N^{-1})$, and $O(N^{-3})$, respectively

with the Sidi sin-3 periodizing transform [25]. This integrand is also featured in a comparison of different higher order QMC methods in [23]. The results are presented in Fig. 2. The top graph shows the standard error after 10 random shifts for four different methods: Monte Carlo, the lattice sequence from above

(up to $s = 4$), the same lattice sequence after sin-3 periodization (to achieve order N^{-3} convergence), and finally the new weighted compound rule with $a = 3$ using the same lattice sequence and periodization. The bottom four graphs show the errors for the four methods separately, with the dashed lines indicating $O(N^{-1/2})$, $O(N^{-1})$, $O(N^{-1})$, and $O(N^{-3})$, respectively. These graphs show the advantage of using the new weighted compound rules compared to using plain QMC. For example, comparing the final two graphs (using the lattice sequence with periodization for $a = 1$ and $a = 3$), one can clearly note the wide band of the error between $O(N^{-1})$ and $O(N^{-3})$ when using the unweighted sequence ($a = 1$), while the weighted sequence ($a = 3$) exhibits the convergence rate $O(N^{-3})$ for all N . The improvement can also be seen clearly in the top graph (i.e., $a = 3$ is almost always better than $a = 1$, except at the powers of 2 when they are equal).

5.3 The effect of different choices of weights

The numerical results in the previous two subsections are encouraging in the sense that higher order convergence is observed as our theory predicted. However, one cannot help noticing some “staircases” in the graphs. Natural questions that may come to mind are: *Is the staircase a side effect of our special choice of weights? Would the optimal choice of weights w_i^* give “smoother” convergence?* Unfortunately, it is not easy to make a meaningful comparison since the result depends very much on the criterion. (To compare worst case errors we need to select the function space. To compare actual or estimated errors we need to select the test integrand. There is also the choice of dimension and the QMC sequence.) Our general observation is that even the optimal weights could yield staircases. As an illustration we include in Fig. 3 a comparison of worst case errors for different weighted compound rules built from the same lattice sequence presented at the beginning of this section. (Here we have an unweighted Korobov space with $\alpha = 3$, dimension $s = 4$ in the top graph and $s = 8$ in the bottom two graphs, and base $b = 2$.) The “true” staircase in the graphs correspond to taking just the largest power of 2 no greater than N , i.e., assign the weight 1 for the subrule with the smallest worst case error and the weight 0 for all other subrules; this is the same as taking $a = \infty$. On the other hand, the case of equal weights is the same as taking $a = 1$. (Note that with the exception of the optimal weights being more costly, it is easy to run our algorithm concurrently for a number of choices of the parameter a .) It can be seen from the graphs that our special choice of weights $a = 2$ or 3 generally performs better than the true staircase, but occasionally it can be worse. Since even the optimal choice of weights yield staircases there, our intuition is that the problem lies in the construction of the lattice sequence rather than in the choice of weights. It is plausible that a lattice sequence which is specifically designed to be used with weights may avoid or alleviate the staircase problem.

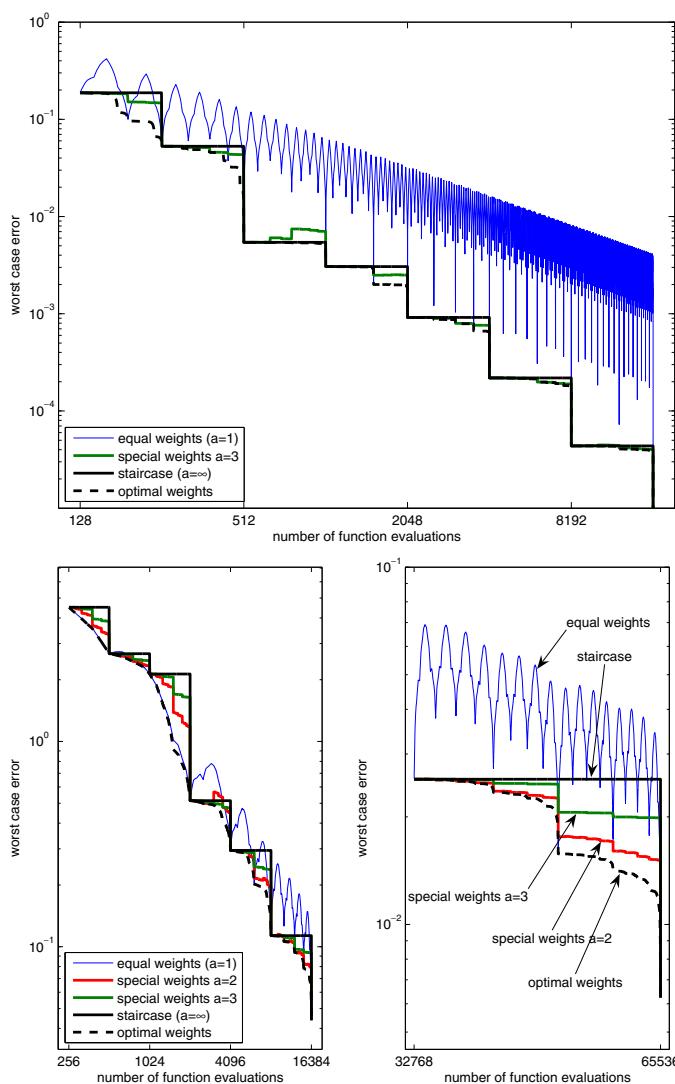


Fig. 3 An illustrative comparison of worst case errors for different choices of weights in 4 dimensions (top graph) and 8 dimensions (bottom two graphs)

Further work is needed in this direction and it falls outside the scope and aim of the present paper.

Acknowledgements The authors thank Josef Dick, Stefan Heinrich and the two anonymous referees for valuable comments and suggestions. The support of the Australian Research Council (ARC), the Austrian Science Fund (FWF-Projects S9609, P21943, and P23389), the Research Foundation Flanders (FWO), and the US National Science Foundation under grant DMS-0713848 are gratefully acknowledged.

References

1. Abramowitz, M., Stegun, I.A. (eds.): *Handbook of Mathematical Functions with Formulas, Graphs and Mathematical Tables*. National Bureau of Standards Applied Mathematics Series, vol. 55. U.S. Government Printing Office, Washington, D.C. (1964)
2. Baldeaux, J., Dick, J., Leobacher, G., Nuyens, D., Pillichshammer, F.: Efficient calculation of the worst-case error and (fast) component-by-component construction of higher order polynomial lattice rules. (2011, submitted)
3. Boyle, P.P., Lai, T., Tan, K.S.: Pricing options using lattice rules. *N. Amer. Actuarial J.* **9**(3), 50–76 (2005)
4. Cools, R., Kuo, F.Y., Nuyens, D.: Constructing embedded lattice rules for multivariate integration. *SIAM J. Sci. Comput.* **28**(6), 2162–2188 (2006)
5. Dick, J.: On the convergence rate of the component-by-component construction of good lattice rules. *J. Complex.* **20**(4), 493–522 (2004)
6. Dick, J., Pillichshammer, F.: *Digital Nets and Sequences: Discrepancy Theory and Quasi-Monte Carlo Integration*. Cambridge University Press, Cambridge (2010)
7. Dick, J., Pillichshammer, F., Waterhouse, B.J.: The construction of good extensible rank-1 lattices. *Math. Comput.* **77**(264), 2345–2374 (2008)
8. Genz, A.: Numerical computation of multivariate normal probabilities. *J. Comput. Graph Stat.* **1**, 141–149 (1992)
9. Gill, S.H., Lemieux, C.: Searching for extensible Korobov rules. *J. Complex.* **23**(4–6), 603–613 (2007)
10. Hammersley, J.M., Handscomb, D.C.: *Monte Carlo Methods*. Methuen & Co. Ltd., London (1964)
11. Hardy, G.H., Littlewood, J.E., Pólya, G.: *Inequalities*. Cambridge University Press, Cambridge (1934)
12. Hickernell, F.J.: Lattice rules: how well do they measure up? In: Hellekalek, P., Larcher, G. (eds.) *Random and Quasi-Random Point Sets*, pp. 109–166. Springer, Berlin (1998)
13. Hickernell, F.J., Hong, H.S., L'Ecuyer, P., Lemieux, C.: Extensible lattice sequences for quasi-Monte Carlo quadrature. *SIAM J. Sci. Comput.* **22**, 1117–1138 (2001)
14. Hickernell, F.J., Niederreiter, H.: The existence of good extensible rank-1 lattices. *J. Complex.* **19**(3), 286–300 (2003)
15. Jensen, J.L.W.V.: Sur les fonctions convexes et les inégalités entre les valeurs moyennes. *Acta Math.* **30**, 175–193 (1906)
16. Kuipers, L., Niederreiter, H.: *Uniform Distribution of Sequences*. Wiley, New York (1974)
17. Kuo, F.Y.: Component-by-component constructions achieve the optimal rate of convergence for multivariate integration in weighted Korobov and Sobolev spaces. *J. Complex.* **19**, 301–320 (2003)
18. Kuo, F.Y., Sloan, I.H.: Lifting the curse of dimensionality. *Not. Am. Math. Soc.* **52**(11), 1320–1328 (2005)
19. Niederreiter, H.: Application of diophantine approximations to numerical integration. In: Osgood, C.F. (ed.) *Diophantine Approximation and its Applications*, pp. 129–199. Academic, New York (1973)
20. Niederreiter, H.: *Random Number Generation and Quasi-Monte Carlo Methods*. Regional Conference Series in Applied Mathematics, Number 63. SIAM, Philadelphia (1992)
21. Nuyens, D., Cools, R.: Fast algorithms for component-by-component construction of rank-1 lattice rules in shift-invariant reproducing kernel Hilbert spaces. *Math. Comput.* **75**(2), 903–920 (2006)
22. Nuyens, D., Cools, R.: Fast component-by-component construction of rank-1 lattice rules with a non-prime number of points. *J. Complex.* **22**(1), 4–28 (2006)
23. Nuyens, D., Cools, R.: Higher order quasi-Monte Carlo methods: a comparison. *AIP Conference Proceedings*, **1281**, 553–557 (2010)
24. Sobol', I.M.: On quasi-Monte Carlo integrations. *Math. Comput. Simul.* **47**(2), 103–112 (1998)
25. Sidi, A.: A new variable transformation for numerical integration. In: Brass, H., Hämerlin, G. (eds.) *Numerical integration. IV. Proceedings of the Fourth Conference held in Oberwolfach, 8–14 Nov 1992*, pp. 359–373. Birkhäuser Verlag, Basel (1993)
26. Sloan, I.H., Joe, S.: *Lattice Methods for Multiple Integration*. Clarendon Press, Oxford (1994)

27. Sloan, I.H., Kuo, F.Y., Joe, S.: Constructing randomly shifted lattice rules in weighted Sobolev spaces. *SIAM J. Numer. Anal.* **40**(5), 1650–1665 (2002)
28. Sugihara, M., Murota, K.: A note on Hasselgrove's method for numerical integration. *Math. Comput.* **39**(160), 549–554 (1982)
29. Vandewoestyn, B., Cools, R.: On obtaining higher order convergence for smooth periodic functions. *J. Complex.* **24**(3), 328–340 (2008)
30. Vandewoestyn, B., Cools, R., Warnock, T.: On obtaining quadratic and cubic error convergence using weighted Kronecker-sequences. *Comput.* **80**(1), 75–94 (2007)