Improper Integrals and QMC Integration in a Grid Environment

Diplomarbeit

zur Erlangung des akademischen Grades
Diplom-Ingenieur
an der Naturwissenschaftlichen Fakultät
der Universität Salzburg

eingereicht von
Heinz Hofbauer

Betreuer
Ao.Univ.-Prof. Dr. Andreas Uhl

Salzburg, March 9, 2007

Contents

Chapter 1

Introduction

High dimensional numerical integration problems may require a significant amount of computation power. Therefore, substantial effort has been invested in finding techniques for performing these computations on all kinds of parallel architectures (see [9, 26, 27, 46] for an exhaustive overview). In order to minimize the communication amount within a parallel system, each processing element (PE) requires its own source of integration nodes. Therefore, the aim is to investigate techniques for using separately initialized and disjoint sets of integration nodes on a single PE.

The work to date has only been applied to the field of high performance computing, which is usually restricted to supercomputers and clusters. A trademark of those is that the PEs run often at the same speed (i.e. homogenous clusters) and are connected with high speed links, high speed network architecture or shared memory. In recent years a new method of supplying computational power has become available and was termed 'GRID'. The GRID however does not enforce equal speed PEs nor does it grant a high speed link to those PEs, which makes it necessary to reevaluate the methods proposed for and researched on HPC architectures.

Currently, the most efficient numerical techniques for evaluating high- dimensional integrals are based on Monte Carlo and quasi-Monte Carlo techniques [13]. Whereas in the Monte Carlo (MC) case the integration nodes are produced by a random number generator (RNG), low-discrepancy point sets and sequences (e.g. (t,m,s)-nets or (t,s)-sequences [35]) are employed in quasi-Monte Carlo (QMC) algorithms. QMC techniques improve the probabilistic error bounds of MC techniques especially in higher dimensions. Nevertheless, these techniques are related [11] since a full period random number sequence may be seen as a low-discrepancy point set (e.g. a rank-1 lattice rule in the case of a linear congruential generator) as well.

In this section an overview over numerical integration as well as on the (Austrian) GRID is given, in subsequent chapters a closer look on the various topics is given as well as a more detailed view on the various subjects which influence the performance or applicability of QMC Integration.

1.1 QMC Computation of Integrals

The basic concept of any method for numerical integration is to approximate the integral by a finite sum, such that

$$I(f) := \int_{I^s} f(x)dx \approx \frac{1}{N}\sum_{n=1}^{N} f(x_n) =: I'_N(f), \tag{1.1}$$

where x_n are suitably chosen and I^s is the unit interval. To identify suitable, i.e. uniformly distributed, point sets x_n the star discrepancy is defined as

$$D_N^* := D_N^*(x_1, \ldots, x_n) = \sup_{J\in\mathcal{F}} \left\| \frac{\#\{x|x \in J\}}{N} - m(J) \right\| \tag{1.2}$$

where $\mathcal{F}$ is the family of all subintervals of the form $J = \prod_{i=1}^{s}[0, t_i) \in I^s$ with volume $m(J)$. There are a number of other ways to measure the "well distributedness" of point sets, such as the diaphony, but the most prominent and widely used is the star discrepancy so we stick with it as much as possible.

To gain a proper error estimation we also need to assess the function, i.e. a continuous function is easy to integrate numerically and any amount of points will result in an exact integral, a function which oscillates on the other hand is harder to approximate. The variation is a way to measure a function in this regard and it is defined as

$$V_{[a,b]}(f) = \sup_{P\in\mathcal{P}} (\sum_i f(x_{i+1}) - f(x_i)),$$

where $\mathcal{P}$ is a family of partitions $P := \{a = x_0, x_1, \ldots, x_{n-1}, x_n = b\}$ of the interval $[a, b]$. For dimension one this is rather intuitive and simple, for higher dimensions this is not sufficient. The extension of this variation to higher dimensions, called variation in the sense of Vitali, can't be directly used for the error estimate, rather the variation in the sense of Hardy and Krause, which is incidentally based on the variation in the sense of Vitali, is used. For a more detailed look on the variation (in the sense of Hardy and Krause) see [39].

The approximation error

$$E_N(f) := |I'_N(f) - I(f)| \tag{1.3}$$

depends on D_N^* and the variation $V(f)$ of the function f in the following way (Koksma-Hlawka inequality [35]):

$$E_N(f) \leq V(f)D_N^*. \tag{1.4}$$

Consequently, point sets exhibiting low discrepancy values are attractive candidates to be used in numerical integration. Using low discrepancy sequences as point set x_n is denoted QMC approach.

Ultimately we do not only want to integrate bounded but also unbounded functions. This leads to the problem of an unbound variation, and consequently to an unbounded error estimate. There are a number of methods which try to prevent the variation from becoming unbounded, usually at the cost of a reduced class of functions to which it can be applied.

1.2 Parallel Execution of Numerical Integration

As stated initially high dimension numerical integration requires a high amount of computational power. This computational power can be gained through parallelization and a, relatively, new method of doing this is called the GRID, in allusion to the power grid. The GRID basically is a network middleware which allows the linking of any number of computers into a heterogenous network. While the single machines the GRID utilizes might not be the fastest computers the overall setup gives access to all the machines, such that, if the task at hand can be properly distributed, the whole can compete with even the fastest clusters.

The problem is that the overall network has no restrictions on network links or machines. Thus we can neither assume fast connections nor any homogeneity, which leads directly to a problem with numerical integration. For an error estimate we need the discrepancy of the point set we utilize. However, the discrepancy is usually only known if all points in a given point set are used. A first approach which would grant that is a client server setup where one server generates the points and sends it out to all clients which need them. This setup however turned out to be less than ideal since even for a low number of clients the server soon turns into a bottleneck which slows down the overall process.

Fortunately there are a number of methods which can alleviate this problem. Originally these methods have been used on HPC clusters and some require homogeneity which is not given in the GRID. Thus we have to reevaluate this distribution methods and see how well they are fit to be used in the GRID.

As a result of these methods we also have to take a closer look at the point sets we use, since some of these methods allow for points to be "lost". Lost points are such which are assigned to a client but never used, leaving gaps in the point sequence which in turn leads to problems concerning the discrepancy. While it is often not possible to gain proper discrepancy bounds for these point sequences if they have gaps we will have to experimentally estimate their fitness for such an approach.

1.3 Austrian Grid

"The AUSTRIAN GRID consortium combines Austria's leading researchers in advanced computing technologies with well-recognized partners in grid-dependant application areas. The goal of the AUSTRIAN GRID is to start and support grid computing in Austria in general, and to provide coordination and collaboration between research areas interested in grid computing."[1]

The grid in general tries to subsume computational resources to create a network which can as a whole provide a high amount of computational power or storage space (in which case it is often called data grid). To achieve this an abstraction layer is used, as such the grid can be seen as an middleware abstraction layer for heterogenous networks. In the Austrian Grid the Globus Toolkit, a open source toolkit for building grids, is used.

In the following chapters the stated problems, integration of unbounded functions, quality of the point sequences and quality of the distribution methods,

[1] Taken from the Austrian Grid homepage `http://www.austriangrid.at`.

will be looked at in more detail.

Notice on Prior Publication

Parts of this thesis have already been published or submitted for publication. The work on leaping has been published in Parallel Processing Letters [20], parts of the work on blocking have been published in Proceedings of the 1st Austrian Grid Symposium [21]. The work on parameterization is to appear in the Proceedings of the 2nd Austrian Grid Symposium [22].The work on the theoretical aspects of the Zinterhof sequence as well as the work on Zinterhof's method for QMC integration of unbounded integrals have been submitted to the Proceedings of MCQMC 2006.

Acknowledgment

The work described in this thesis is partially supported by the Austrian Grid Project, funded by the Austrian BMBWK (Federal Ministry for Education, Science and Culture) under contract GZ 4003/2-VI/4c/2004.

Chapter 2

Numerical Integration of Unbounded Functions

A big problem with error estimation in numerical integration is the fact that the variation is rather restrictive. Even simple functions like $m(x) = max(x_1 + x_2 + x_3 - 1, 0)$ are of unbounded variation $V(m) = \infty$, see [39]. Also if a function is unbounded the variation is unbounded resulting in an error estimate $E_N(f) \leq \infty$.

There have already been a number of methods proposed in literature which aim at tackling the problem of numerically integrating unbounded functions, which will be discussed in Section 2.1. All of these methods severely restrict the class of functions to which they can be applied. Some methods however also are problematic in application.

In this chapter we will give an overview over the state of the art methods. Afterwards, Zinterhof's method will be described in detail since this is the method used in all experiments. The closer look at Zinterhof's method will especially concern the restrictiveness of the method as well as experiments to get a better understanding how it works.

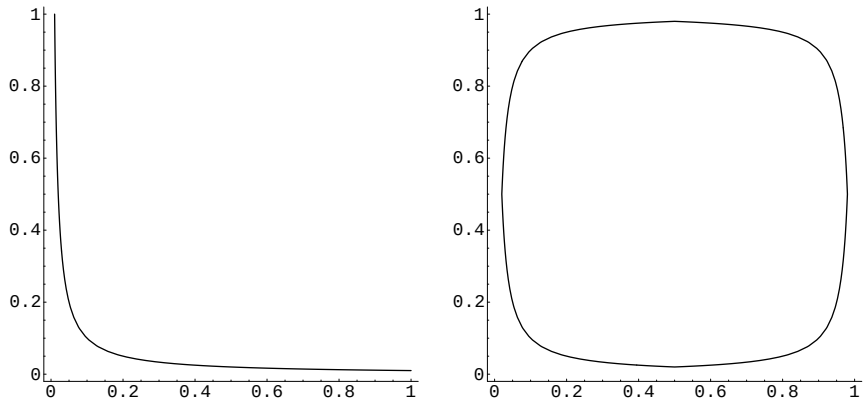
2.1 Discussion of Previous Work

In the case of singularities the Koksma-Hlawka inequality becomes meaningless since functions containing singularities are unbound and thus of infinite variance. When examining methods of numerical integration for integrands with singularities usually the distinction is made whether the singularities are in the interior of the unit cube or on the boundary.

2.1.1 Singularities on the Boundary

Sobol' [48] investigated a number of functions which have singularities in the origin. By restricting the growth of the integral near the singularity to $o(N)$ he shows that

$$\lim_{N\to\infty} \frac{1}{N} \sum_{\mu=1}^{N} f(P_\mu) = \int_{I^s} f(P) dP$$



$xy = 0.01$ $\min(x, 1-x)\min(y, 1-y) = 0.01$

Figure 2.1: Hyperbolic origin and all corner avoidance.

holds, but unfortunately fails to give an error bound. He allows one dimensional functions $f(x)$ to have a rational singularity $0 < \xi < 1$ but reduces them to functions with singularities in the origin by linearly transforming $[0, \xi)$ into $(0, 1)$ and $(\xi, 1]$ into $(0, 1]$. Thus two functions are obtained $f^1_\xi = (1-\xi)f(\xi + x + x\xi)$, $f^2_\xi = \xi f(\xi + x\xi)$ where $f(x) = f^1_\xi + f^2_\xi$. In the multi dimensional case Sobol's test function is $f(x) = x_1^{-\beta_1} \cdots x_s^{-\beta_s}$ where the growth condition holds if $\forall i\, \beta_i < 1$, which is the general form of a test function we use (i.e. $\beta_1 = \cdots = \beta_s = \alpha$ and thus $f(x) = x_1^{-\alpha} \cdots x_s^{-\alpha}$). Sobol' supplied no experimental evaluation of his theory in the paper.

The basic idea in [38] is to replace the part of the function which is not touched by the numerical integration, i.e. the part lying in the hyperbolic or L-shaped region avoided by the Halton sequence, by a bounded extension of the function. This way the he attains a finite variance for the function and can proof error bounds for the numerical integration.

Owen shows that the Halton sequences avoid all corners in a hyperbolical sense. He also shows that for a finite $C > 0$ the Halton points $x_1, \ldots, x_n$ avoid the hyperbolic region $\{x | \prod_j x^j \le Cn^{-1}\}$, illustrated in fig. 2.1, while independent uniform points x_n enter that region infinitely often, with probability one. It is also shown that while points from the Halton sequence avoid the 0 and 1 corner stronger than independent uniform points this doesn't hold for all other corners. To show an error bound for numerical integration with point sequences which avoid the origin in a hyperbolic $\{x \in [0,1]^d | \prod_{1 \le i \le d} x^i \ge \epsilon\}$ or L-shaped $\{x \in [0,1]^d | \min_{1 \le i \le d} x^i \ge \epsilon\}$ way Owen also imposes growth conditions on the functions. The grow condition for functions on $(0,1]^d$ that are singular as x approaches the origin is as follows where $\chi_u(j)$ is the characteristic function of j in u. The function f fulfills the growth condition if, where for a set $u \subset \{1, \ldots, d\}$ of indices the symbol $\partial^u f(x)$ represents $(\prod_{i \in u} \partial / \partial x^i) f(x)$ (with the convention

that $\partial^{\emptyset} f(x) = f(x)$),

$$|\partial^u f(x)| \leq B \prod_{j=1}^{d} (x^j)^{-A_j - \chi_u(j)}$$

holds for some $A_j > 0$, and $B < \infty$ and all $u \subset \{1, \ldots, d\}$.

Putting the growth condition and the corner avoidance properties (for hyperbolic regions) together he states the following theorem:

Let the function f satisfy the condition $|\partial^u f(x)| \leq B \prod_{i=1}^{d} (x^i)^{-A_i - \xi_u(i)}$ for some $A_i > 0$, $B < \infty$ and all $u \subset \{1, \ldots, s\}$. Also suppose that for $N \geq 1$ and all $1 \leq n \leq N$ the sequence $(\mathbf{x}_n)_{1 \leq n}$ satisfies $\prod_{i=1}^{s} x_n^{(i)} \geq cN^{-r}$. Then for any $\eta > 0$ we have

$$|I' - I| \leq C_1 D_N^*(x_1, \ldots, x_N) N^{\eta + r \max_i A_i} + C_2 N^{r(\max_u A_i - 1)}$$

with finite constants C_1 and C_2, that may depend on η. The estimate holds with $\eta = 0$, if there exists a unique maximum among $A_1, \ldots, A_s$.

Finally it is shown that for L-shaped regions Monte Carlo methods attain a root mean square error of $O(n^{-1/2})$ when $max_j A_j < 1/2$. While low discrepancy sequences which are confined to an L-shaped region with $\epsilon = C/n$ are asymptotically superior to Monte Carlo when $\sum_{j=1}^{d} A_j < 1/2$, they can be inferior if $\sum_{j=1}^{d} A_j > 1/2$, especially when $\max_j A_j < 1/2$ and $\sum_{j=1}^{d} A_j > 1$ where Monte Carlo sampling has a root mean square of $O(n^{-1/2})$ and for low discrepancy sequences we only have the estimate $|I' - I| \leq \infty$.

For hyperbolic corner avoidance the error of Monte Carlo sampling is the same as for the L-shaped case. The error bound of the Halton sequence is asymptotically smaller than $O(n^{-1/2})$ when $\max_j A_j < 1/(2r)$. For origin avoidance or for $d = 1$, r becomes 1 and then the Halton sequence is superior to Monte Carlo. When not in the origin there is always a bound, dependent on r, where the Halton sequence is provably superior to Monte Carlo but not necessarily if $1/2r < max_j A_j < 1/2$.

The only experimental results in the paper are plots illustrating the corner avoidance properties of the Halton sequence regarding a measure used in the proofs and in relation to uniformly distributed points.

Kainhofer et al. show in [16] that generalized Niederreiter sequences possess corner avoidance properties similar to Halton sequences around the origin. They also show the corner avoidance rates for Halton and Faure sequences for corners different than the origin. This is essentially an extension of the results found by Owen ([38]) to Niederreiter sequences. To get efficient QMC rules for the integrands one has to find point sets satisfying the condition

$$\prod_{i=1}^{s} x_n^{(i)} \geq cN^{-r}$$

with small r as stated in [38]. The theorem also holds for an all corner case

when the avoidance condition is written as

$$\min_{1\le n\le N}\prod_{i=1}^{s}\min(1-x_n^{(i)},x_n^{(i)})\ge cN^{-r}.$$

They show that for each point $\mathbf{x}_n \le n < b^l$ of a generalized Niederreiter (t,s)-sequence in base b the bound $\prod_{i=1}^{s} x_n^{(i)} \ge b^{-l-t-s}$ holds. Consequently, for a generalized Niederreiter $(0,s)$-sequence in base b (sometimes called generalized Faure sequence) $(\mathbf{x}_n)_{1\le n}$ and $0 \le n \le b^l$ the bound $\prod_{i=1}^{s} x_n^{(i)} \ge b^{-l-s}$ holds.

Now for $b^{l-1} \le n < b^l$ with arbitrary $l > 1$ we have $\prod_{i=1}^{s} x_n^{(i)} \ge b^{-l-t-s} \ge b^{-t-s+1}n^{-1}$, and thus (t,s) sequences avoid the origin hyperbolically with order $r = 1$.

For the numerical evaluation of functions with singularities on all boundaries of the unit cube Kainhofer uses the minimal volume intervals defined by a given corner $\mathbf{h} = (h_1,\ldots,h_s) \in \{0,1\}^s$ of the unit cube and the points $\mathbf{x}_n$, called the minimal hyperbolic distance of the points $\mathbf{x}_n$ to the corner $\mathbf{h}$,

$$M_N(\mathbf{h}) = \min_{1\le n\le N}\prod_{i=1}^{s}|h_i - x_n^{(i)}|.$$

The left part can be written $M_N(\min)$ meaning $\min_{\mathbf{h}} M_N(\mathbf{h})$. In the following $J := \{i \in \{1,\ldots,s\} | h_i = 0\}$ and $K := \{i \in \{1,\ldots,s\} | h_i = 1\}$, i.e for a corner $\mathbf{h}$ J is the index set where the h_i are 0, and K is the index set where the h_i are 1.

For Halton sequences Kainhofer et al. show that for $n > 1$ and $\mathbf{x}_n$ the n-th point of the Halton sequence in distinct prime base $p_1,\ldots,p_s$ there exist subsequences $\mathbf{y}_n = \mathbf{x}_{N(n)}$ for which the minimum distance $M_{N(n)}(\mathbf{h})$ to any mixed corner $(\mathbf{h})$ is bounded by

$$M_{N(n)}(\mathbf{h}) = O(\frac{1}{N(n)\log N(n)}).$$

Owen [38] showed that $M_N(\mathbf{0}) \ge cN^{-1}$ and thus Halton sequences tend faster to mixed corners than towards the origin.

Additionally he shows that for Halton sequences in relative prime bases p_i the corner avoidance bound becomes $M_N(\min) = O(N^{-1-\epsilon})$ for every $\epsilon > 0$ if and only if there are $C, \tilde{C}, \alpha_j$ and β_k which fulfill

$$1 = \tilde{C}\prod_{k\in K} p_k^{\beta_k} - C\prod_{j\in J} p_j^{\alpha_j}.$$

For Faure sequences we know that $M_N(\mathbf{0}) > cN^{-1}$ since they are generalized Niederreiter $(0,s)$-sequences. For the mixed corner case, let s be prime and p be the least prime larger or equal to s, than there exists a subsequence $\mathbf{y}_n = \mathbf{x}_{N(n)}$ such that $\prod_{i=1}^{s}|h_i - y_n^{(i)}| \le \frac{p^3}{N(n)^2}$.

For the corner $\mathbf{1} = (1,1,\ldots,1,1)$ Kainhofer et al. show that for $s = 2$ there exists a subsequence $\mathbf{y}_n = \mathbf{x}_{N(n)}$ such that

$$\prod_{i=1}^{s}|h_i - y_n^{(i)}| \le \frac{p^2}{N(n)^2},$$

and for $s > 2$ there exists a subsequence $\mathbf{y}_n = \mathbf{x}_{N(n)}$ such that

$$\prod_{i=1}^{s} |h_i - y_n^{(i)}| \leq \frac{p^{5/2}}{(N(n)+1)^{3/2}},$$

where for both cases p is the least prime larger or equal to s.

No experimental results are given.

In relation to Zinterhof's work the requirements are rather stiff, hyperbolic corner avoidance, singularities on the boundaries, rather specific growth conditions, but if these requirements are fulfilled the method can be applied without further ado and is under such conditions rather elegant.

Kainhofer, Hartinger, and Tichy [18] also dealt with QMC methods for multidimensional integrals with respect to a measure other than the uniform distribution. They allow the integrand to be unbound on the lower boundary of the interval and justify the "strategy of ignoring the singularity" by using weighted integration with an non uniform distribution. This means integration problems of the form

$$I_{[\mathbf{a},\mathbf{b}]} := \int_{[\mathbf{a},\mathbf{b}]} f(\mathbf{x}) dH(\mathbf{x})$$

where H denotes a s-dimensional distribution with support $K = [\mathbf{a}, \mathbf{b}] \subset \mathbb{R}^s$ and f is a function with singularities on the left boundary of K. To use a generalized version of the Koksma-Hlawka inequality they have to define a H-discrepancy of $\omega = (y_1, \ldots)$ which measures the distribution properties of the sequence. It is defined as

$$D_{N,H}(\omega) = \sup_{J \subset K} |N^{-1} A_N(J, \omega) - H(J)|$$

where A_N counts the number of elements in $(y_1, \ldots, y_N)$ falling into the interval J, e.g. $A_N(J, \omega) = \sum_{n=1}^{s} \chi_J(y_n)$, and $H(J)$ denotes the probability of $J \subset K$ under H. With this $D_{N,H}$ they can define the define the Koksma-Hlawka inequality for this case as

$$|I_K - N^{-1} \sum_{n=1}^{N} f(y_n)| \leq V(f) D_{N,H}(\omega).$$

Now for the one-dimensional case let $a \leq c \leq c_N$. If a sequence $(y_i)_i \in \mathbb{N}$ and a differentiable function $f(x)$ on $[a, b]$ with a singularity only at the left boundary satisfy the condition

$$D_{N,H}(\omega) \int_c^b |f'(x)| dx = o(1)$$

as well as $c_N \to a$ for $N \to \infty$, then the QMC estimator converges to the value of the improper integral of $f(x)$ on $[a, b]$:

$$\lim_{N \to \infty} N^{-1} \sum_{n=1}^{N} f(y_n) = \int_a^b f(x) dH(x)$$

where $c_N = min_{1 \leq n \leq N} y_n$.

While the authors state that there is a certain lack of sequences with low H-discrepancy they also propose a fix by using the Hlawka and Mück method [19] for constructing such sequences. However for such a sequence $\tilde{\omega}$ there might be some elements $\tilde{y}_k$ which are 0. Since the singularities of $f(x)$ are on the lower boundary these sequences are not directly fit to be used with the numerical integration, however a simple change is proposed, generating a new sequence $\bar{\omega}$ as follows:

$$\bar{y}_i = \begin{cases} \tilde{y}_k & \text{if } \tilde{y}_k \geq N^{-1}, \\ N^{-1} & \text{if } \tilde{y}_k = 0 \end{cases}$$

for $i = 1, \ldots, N$. The H-discrepancy of $\bar{\omega}$ is then bound by

$$D_{N,H}(\bar{\omega}) \leq (M+1)(D_N + \frac{1}{N})$$

where $M = \sup_{x \in [0,1]} h(x)$. Since the discrepancy of uniform low discrepancy sequences is typically of the order $O(\frac{logN}{N})$ the additional factor N^{-1} inherited through the transformation doesn't change the asymptotic behavior of the integration error.

For the multi-dimensional case the idea is basically the same, Kainhofer et al. state a convergence criterium as follows.

Let $f(x)$ be a function in $[a, b]$ with singularities only at the left boundary of the definition interval (i.e. $f(x) \to \infty$ only if $x^{(j)} \to a_j$ for at least on j), and let furthermore $c_{Nj} = \min_{1 \leq n \leq N} y_n^{(j)}$ and $a_j < c_j \leq c_{Nj}$. If the improper integral exists and if

$$D_{N,H}(\omega) V_{[a,b]}(f) = o(1),$$

then the QMC estimator converges to the value of the improper integral:

$$\lim_{N \to \infty} N^{-1} \sum_{n=1}^{N} f(y_n) = \int_{[a,b]} dH(x).$$

Similar to the one-dimensional case the construction of H-distributed sequences leads to problems when using Hlawkas method. However a fix to the generated sequence is given by the authors:

Let H be a s-dimensional distribution with independent marginal distribution $H_1, \ldots, H_s$ and $M = \sup h(x) < \infty$. Let furthermore $\omega = (x_1, \ldots)$ be a sequence with uniform discrepancy $D_N(\omega)$ and define the sequence $\tilde{\omega} = (\tilde{y}_1, \ldots)$ by

$$\tilde{y}_i^{(j)} = \frac{1}{N} \sum_{n=1}^{N} \lfloor 1 + x_i^{(j)} - H_i(x_n^{(j)}) \rfloor,$$

for $j = 1, \ldots, s$ and $n = 1, \ldots, N$. Then the sequence $\bar{\omega}$ with elements

$$\bar{y}_i^{(j)} = \begin{cases} \tilde{y}_k^{(j)} & \text{if } \tilde{y}_k^{(j)} \geq \frac{1}{N}, \\ \frac{1}{N} & \text{if } \tilde{y}_k^{(j)} = 0 \end{cases}$$

has the following two properties:

$$D_{N,H}(\bar{\omega}) \leq (1 + 4M)^s D_N(\omega),$$

$$\min_{1\le j\le s, 1\le i\le N} \tilde{y}_i^{(j)} \ge \frac{1}{N}.$$

No experimental results were given.

In [17] Hartinger and Kainhofer deal with the problem of generating low discrepancy sequences with an arbitrary distribution H. While they did so before ([18]) they identify some disadvantages which carry over to the transformed sequence they proposed. They specifically deal with the property of the Hlawka-Mück method that for some applications the points of the generated sequence of a set with cardinality N lie on a lattice with spacing $1/N$. Their solution is to use a smoothed approximation where the values between the jumps are interpolated in the empirical distribution function. Specifically they state the following theorem.

Let $\omega_N = (x_1, \ldots, x_N)$ be a sequence in U^s with discrepancy $D_N(\omega_N)$, and $H(x)$ a distribution function with bounded, continuous density $h(x) = \prod_{i=1}^{s} h_i(x^{(i)})$ and $h_i(x^{(i)}) \le M < \infty$ for all i. Furthermore, let $H_i(x) = \int_0^x h_i(u)du$ and define for $k = 1, \ldots, N$ and $l = 1, \ldots, s$ the values

$$x_k^{(l)-} = \max_{\mathcal{A}=\{x_i\in\omega_N | H_i(x_i^{(l)})\le x_k^{(l)}\}} x_i^{(l)} \text{ and } x_k^{(l)-} = 0 \text{ for } \mathcal{A} = \emptyset,$$

$$x_k^{(l)+} = \max_{\mathcal{B}=\{x_i\in\omega_N | H_i(x_i^{(l)})\ge x_k^{(l)}\}} x_i^{(l)} \text{ and } x_k^{(l)+} = 1 \text{ for } \mathcal{B} = \emptyset.$$

Then the discrepancy of the set $\bar{\omega}_N = (y_k)_{1\le k\le N}$ generated by

$$y_k^{(l)} = \frac{H_l(x_k^{(l)+}) - x_k^{(l)}}{H_l(x_k^{(l)+}) - H_l(x_k^{(l)-})} x_k^{(l)-} + \frac{x_k^{(l)} - H_l(x_k^{(l)-})}{H_l(x_k^{(l)+}) - H_l(x_k^{(l)-})} x_k^{(l)+}$$

can be bound by

$$D_{N,H}(\bar{\omega}) \le (1+2M)^s D_N(\omega).$$

In order to integrate functions with singularities at the boundary it will be convenient to shift the interpolated sequences in an appropriate way to avoid regions that lie too close to that singularity. The authors define how to generate a new sequence $\hat{\omega}$ from the above constructed sequence $\bar{\omega}$ which has the same distance to the boundaries $\min_{k=1,\ldots,N} \min_{1\le j\le s} \min(\hat{y}_k^{(j)}, 1-\hat{y}_k^{(j)})$ as the original sequence ω. The new sequence has the same bounds concerning the discrepancy as the sequence $\bar{\omega}$

$$D_{N,H}(\hat{\omega}) \le (1+2M)^s D_N(\omega)$$

and is constructed by

$$\hat{y}_k^{(l)} = \begin{cases} \hat{y}_k^{(l)+} & \text{if } (A) = \emptyset, \\ \hat{y}_k^{(l)-} & \text{if } (B) = \emptyset, \\ \hat{y}_k^{(l)} & \text{otherwise} \end{cases}$$

where $\hat{\omega} = (\hat{y}_1, \ldots, \hat{y}_n)$.

They show, by utilizing the same techniques as Owen in [38], that the sequence can be used to integrate improper integrals which have a singularity in the corner. For L-shaped corner avoidance of the sequence they give an error by the following theorem, where $K_{om}(\epsilon) = \{x \in U^s | \min_{1\le j\le s} x^{(j)} > \epsilon\}$.

Let $f : U^s \to \mathbb{R}$, and $\omega_N = \{x_1, \ldots, x_n\}$ be a sequence with $x_j \in K_{om}(\epsilon_N)$ for $1 \leq j \leq N$. Let furthermore $H(x)$ be a distribution on U^s with density $h(x)$ and $M_\epsilon = \sup_{x \in U \ K_{om}(\epsilon)} h(x) \leq \infty$. If f fulfills the growth condition given in [38], and $0 \leq \epsilon_N = CN^- r$, then

$$|I' - I| \leq C_1 D_{N,H} N^{r \sum_{j=1}^{s} A_j} + C_2 M_{\epsilon_N} N^{r(\max A_j - 1)}$$

with some explicitly computable, finite constants C_1 and C_2.

Similarly they can show an error for the hyperbolic corner avoidance case, and finally they give an order of the error estimate for hyperbolic corner avoidance by $O(N^{-1+\epsilon+s \max_{j=1,\ldots,s} A_j})$.

The only experiments conducted were done in order to show the difference between the method proposed by Hlawka and Mück and their interpolation method. No experiments directly connected to numerical integration are given.

In [18] and [17] the singularity is ignored but not by bounding the function, as in [54], but by modifying the integration, i.e. weighted integration. In comparison the generation of weighted integration is computationally more expensive than the simple bounding of the function, additionally due to the modification of the H distributed points to avoid 0 this is only applicable to singularities on the lower bounds of the integration domain.

De Doncker [5] reduces the error rate of the methods of Klinger and Sobol' by constructing extensions which reduce the approximation error. She looks at the leading asymptotic order of the error and generates extrapolations for such functions in such a way that error terms vanish. She has shown that for one dimensional functions with algebraic end point singularities her method works very well, furthermore it gains significant convergence acceleration when applied to some logarithmic and interior algebraic singularities. Additionally an asymptotic error expansion was derived for integrands with algebraic singularities at the boundaries of the d-dimensional unit cube.

The improvements were found to occur in stages, as each error term vanishes. She also states that further research is needed to determine conditions for which an exact order of leading error terms can be established, and thus a proper extrapolation can be made.

2.1.2 Singularities in the Interior

In [40] Owen applied and extended his results from [38] to singularities $z \in [1, 0]^d$ inside the unit cube. However since the sequences don't avoid the region of the singularity, which can be in the interior of the unit cube, he proposes using the extended function $\tilde{f}$ instead of the original function f for numerical integration. Like in [38] he requires the function to obey a growth condition

$$|\partial^u f(x)| \leq B||x - z||_p^{-A-|u|}$$

for all $u \subset \{1, \ldots, d\}$, all $x \neq z$, some $0 < A < d$, some $B < \infty$, and some $1 \leq p < \infty$. He then defines an extendible region K around z for which $||x - z||_p \geq \nu$ holds for some $\nu > 0$, additionally he defines an anchor $c \in K$ for which $\text{rect}[c, y] \subset K \ \forall y \in K$ holds, where $\text{rect}[x, y] = \prod_{i=1}^{s} [\min(x_i, y_i), \max(x_i, y_i)]$ is the rectangular hull of x and y. Thus he can use Sobol's extension $\tilde{f}(x)$, see [40] and [38] for details. With the help of the extendible region, $\tilde{f}$ and the growth

condition he shows that for any Lebesgue measurable function f with integral I the equation

$$E(|I' - I|) = O(N^{(-1+\epsilon)(d-A)/d})$$

holds where $I' = \frac{1}{N}\sum_{i=1}^{N} \tilde{f}(x_i)$. In a final corollary he shows that if $A < d/2$ then $E(I' - I) = o(N^{-1/2})$ holds for randomized quasi-Monte Carlo.

Owen has not given any experimental results.

Owen states in the conclusion that it is not clear if such a good extension to f can be found. This and the fact that the generation of the extension requires specific knowledge of the function f makes this method rather unwieldy when compared to the simple transformation used by Zinterhof.

Klinger [24] shows that the numerical integration of a function is still possible when it has a singularity in the origin, or can be transformed such that the singularity is in the origin, by removing the point closest to the origin from the integration. This basically excludes an elemental interval containing the origin from the estimation, for Halton sequences he defines similar intervals. He researches the function $f_{\mathbf{y}}(\mathbf{x}) = \frac{g(\mathbf{x})}{|x_1-y_1|^{\beta_1}+\cdots+|x_s-y_s|^{\beta_s}}$ where g has bounded variation, $g(\mathbf{y}) \neq 0$ and $\beta_k > 0$ for all k. Clearly this function has a singularity in $\mathbf{y}$ and g doesn't influence the asymptotic behavior near $\mathbf{y}$. Klinger only deals with the function $f(\mathbf{x}) := f_{\mathbf{0}}(\mathbf{x})x_1^{-\beta_1}\cdots x_s^{-\beta_s}$, e.g. $\mathbf{y} = \mathbf{0}$ and $g(\mathbf{x}) = 1$. Should for some case $\mathbf{y} \neq \mathbf{0}$ a simple transformation $x^* = \{x - y + 1\}$, where $\{x\}$ is the fractional part of x, will reduce the function $f_{\mathbf{y}}(\mathbf{x})$ to $f(\mathbf{x}^*)$. Only Halton and $(0, s)$-sequences are used and he uses the properties of elemental intervals to find points which are near the singularity and thus not included in the numerical integration. While the $(0, s)$-sequence is a Niederreiter sequence, and thus a notion of elemental intervals already exits, Klinger needs to define a similar notion for Halton sequences:

Let a_k be positive rational numbers which satisfy $\sum_{k=1}^{s} 1/a_k \geq 1$ and define positive numbers, coprime integers p_k, q_k by $p_k/q_k := a_k$. Now let

$$R = \prod_{k=1}^{s} [\delta_k(\delta_k - CN^{-1/a_k}),\ (1-\delta_k)CN^{-1/a_k} + \delta_k)$$

where $\delta_k \in \{0, 1\}$ and $C = \min_{1\leq k\leq s} b_k^{-q_k}$. Then at most one of the first N points $x_0, \ldots, x_N$ of Halton's sequence falls into R.

Consequently he states that since $x_0 = \mathbf{0}$, this arguments also shows that none of the first N points of a Halton sequence x_n fall into the the above interval when $\delta_k = 0$ for all $1 \leq k \leq s$.

He then states the error bounds for Halton and $(0, s)$-sequences as follows.

Let $x_1, \ldots, x_N$ be the s-dimensional Halton sequence and $f(x)$ as above then we have

$$E_N = O(N^{-1+(\sum_{i=1}^{s}\frac{1}{\beta_i})^{-1}}(logN)^s).$$

Let $x_1, x_2, \ldots$ be a $(0, s)$-sequence and $f(x)$ as above then we have

$$|I(f) - S_N^*(f)| = O(N^{-1+(\sum_{i=1}^{s}\frac{1}{\beta_i})^{-1}}(logN)^s),$$

where

$$S_N^* = S_N - \frac{1}{N}\max_{1\leq n\leq N} f(x_n).$$

In the error bound for the $(0, s)$-sequence the S_N is the usual QMC estimator for the integral and $\max_{1\leq n\leq N} f(x_n)$ is the point closest to the singularity, which is in $\mathbf{y} \neq \mathbf{0}$ and thus the point is the only point in the elementary interval bordering the origin. This is essentially the same as removing x_0 from the Halton sequence, except that the n for the point is dependent on N and thus can't be specified in advance as with the Halton sequence.

The computational experiments included in the paper compare the error bounds of the Halton, Sobol' and Niederreiter sequences. The error bounds are shown to be reasonable and also that Halton sequences have, for low dimension, basically the same characteristics as Sobol' or Niederreiter sequences but are less computationally expensive. For high dimension, shown for dimension 10 in the experiments, Halton sequences are worse for at least moderate N.

Clearly these methods, especially for singularities in the interior, are rather hard to implement for non specialists. Owen even notes that for his method it might not be possible to find good extensions around the singularity. The method proposed in the subsequent section is in this respect extraordinarily simple, however this simplicity in application entails a rather problematic (at least theoretically) restriction to the class of functions to which it can be applied.

2.2 A Pragmatic Approach - Zinterhof's Method

A number of people, starting with Sobol' in [48], have conducted research for error bounds for improper integrals. One of the more recent results is by Zinterhof [54].

Theorem 2.2.1. For a function $f(x)$ and a $B > 0$ the functions $f_B(x)$ and $\hat{f}_B(x)$ are defined as

$$f_B(x) = \begin{cases} f(x) & |f(x)| \leq B \\ 0 & |f(x)| > B \end{cases}$$

$$\hat{f}_B(x) = \begin{cases} 0 & |f(x)| \leq B \\ f(x) & |f(x)| > B. \end{cases}$$

The Class $C(\beta, \gamma)$ of s-variate functions $f(x_1, \ldots, x_s)$, $0 \leq x_i \leq 1$, $i = 1, \ldots, s$, consists of all functions which fulfill

$$\begin{array}{lll} \text{(a)} & I(|\hat{f}_B|) & = O(B^{-\beta}) \text{ for some } \beta > 0 \\ \text{(b)} & V(f_B) & = O(B^{\gamma}) \text{ for some } \gamma \geq 1. \end{array}$$

Let $f \in C(\beta, \gamma)$, D_N^* be the discrepancy of the set of nodes $\vec{x}_1, \ldots, \vec{x}_n$ and $B = {D_N^*}^{-1/(\beta+\gamma)}$, then the estimate

$$I(f) = \frac{1}{N} \sum_{n=1}^{N} f_B(\vec{x}_n) + \mathrm{O}({D_N^*}^{\beta/(\beta+\gamma)})$$

holds, where $I(f) = I(f_B) + I(\hat{f}_B)$.

Proof. From

$$I(f) = I(f_B) + I(\hat{f}_B)$$

using the Hlawka-Koksma inequality we get

$$\left| I(f_B) - \frac{1}{N}\sum_{n=1}^{N} f_B(x_n) \right| \leq V(f_B) D_N^* \leq C_1(f) B^{\gamma} D_N^*$$

and from condition (a) we get

$$|I(\hat{f}_B)| \leq C_2(f) B^{-\beta}.$$

Consequently

$$E_N = \left| I(f) - \frac{1}{N}\sum_{n=1}^{N} f_B(x_n) \right| \leq C_1(f) B^{\gamma} D_N^* + C_2(f) B^{-\beta},$$

which takes it's minimum of order when using

$$B = {D_N^*}^{\frac{-1}{\beta+\gamma}}.$$

Thus for an error estimate we get

$$E_N \leq C(f) {D_N^*}^{\beta/(\beta+\gamma)}$$

where $C(f)$ is a constant depending on f. □

Remark 2.2.1 (Quality of the error bound). Let us look at dimension $s = 1$, $f(x) = x^{-\alpha}$ and the points $\frac{k}{N}$ $k = 1, \ldots, N$. We can easily get the following:

$$V(f_B) = \mathrm{O}(B)$$

$$I(|\hat{f}_B|) = \frac{1}{1-\alpha}(\frac{1}{B})^{\frac{1}{\alpha}-1} = \mathrm{O}(B^{-\beta})\, \beta = \frac{1}{\alpha - 1}$$

$$D_N^*(\frac{1}{N}, \ldots, \frac{N}{N}) = \frac{1}{N}.$$

Consequently we use $B = N^{\alpha}$ and the point where the functions f_B and $\hat{f}_B$ jump is $a = N^{-1}$. To confine the error from above and below we get the upper bound

$$\begin{aligned} R_N &= I(\hat{f}_B) + \left| \sum_{k=1}^{N} \frac{1}{(\frac{k}{N})^{\alpha}} - I(f_B) \right| \leq I(\hat{f}_B) + V_{[1/N,1]}(f_B) D_N^* \leq \\ &\leq \frac{1}{1-\alpha} N^{-\alpha+1} + (N^{\alpha} - 1)\frac{1}{N}. \end{aligned}$$

And likewise the lower bound

$$\begin{aligned} R_N &= \frac{1}{N}\sum_{k=1}^{N} \frac{1}{(\frac{k}{N})^{\alpha}} - I(f) = \frac{1}{1-\alpha} N^{-\alpha+1} + \sum_{k=1}^{N} \left(\frac{N^{\alpha-1}}{k^{\alpha}} - \int_{k/N}^{(k+1)/N} \frac{1}{x^{\alpha}} dx \right) > \\ &> \frac{1}{1-\alpha} N^{-\alpha+1} + \frac{1}{2N}\sum_{k=1}^{N} \left(\frac{1}{(\frac{k}{N})^{\alpha}} - \frac{1}{(\frac{k+1}{N})^{\alpha}} \right) = \frac{1}{1-\alpha} N^{-\alpha+1} + \frac{1}{2N}(N^{\alpha} - 1). \end{aligned}$$

When we put this together we have:

$$\frac{1}{1-\alpha}N^{-\alpha+1} + \frac{1}{2N}(N^{\alpha}-1) < R_N \leq \frac{1}{1-\alpha}N^{-\alpha+1} + \frac{1}{N}(N^{\alpha}-1)$$

and with $\gamma = 1$, $\beta = 1/(\alpha-1)$ and $D_N^* = 1/N$ we can write

$$C_1(\alpha){D_N^*}^{\beta/(\beta+\gamma)} < R_N \leq C_2(\alpha){D_N^*}^{\beta/(\beta+\gamma)}$$

where

$$C_1(\alpha) = \frac{1}{1-\alpha} + 1/2$$

$$C_2(\alpha) = \frac{1}{1-\alpha} + 1.$$

So concerning $\mathrm{O}({D_N^*}^{\beta/(\beta+\gamma)})$ we can do no better.

2.2.1 The Class $C(\beta, \gamma)$

An important issue remains: the richness of the class $C(\beta, \gamma)$. Generally if the jump line, i.e. $f(x) = B$, $x \in U^s$, is not axis parallel the variation is unbounded and consequently $f \notin C(\beta, \gamma)$ since condition (b) (in Theorem 1) is violated [1]. Fig 2.2 illustrates this for a simple case, a function from $\mathbb{R}^2$ to $\mathbb{R}$, the hatched area is a constant value B greater than zero and the rest is zero. On the left side the jump line from B to zero is axis parallel and on the right side is the simplest version where this is not the case, a diagonal. When we look at the area in the center we see that the variation (in the sense of Vitali) is B for this area. When we refine this part of the partition, the areas displayed by dashes, we see that in the left side only the part which covers the corner has variation greater than zero. All other areas are either constant or cross the jump line in such a way that two adjacent corners lie in the $f(x) = B$ region and two adjacent corners lie in the $f(x) = 0$ region. However, adjacent corners are of opposite sign and thus cancel out. The case on the right hand side is different, here we can create a refinement where there is more than one area which attains a value greater than zero. Only one point of the dashed areas is in the $f(x) = B$ region and thus the variation of the area is B. Through this refinement we have tripled the variation of the original area, and we can in turn refine each of this smaller areas further. It is simple to get an infinite variation, we just have to keep refining. When we use the variation in the sense of Hardy and Krause, for the right hand we are still infinite. For the left side it is easily seen that the lower dimensional faces also can only have axis parallel jump lines and thus are similarly fixed in their value. It can clearly be seen that the class $C(\beta, \gamma)$ is very restrictive.

Consider the class T of bounded step functions $t \in T$, $t(x_1, ..., x_s) : U^s \to \mathbb{C}$. These are functions which are piecewise constant on U^s where U^s is partitioned into a finite number of, pairwise disjoint, intervals of the form $\prod_{i=1}^{s}[a_i, b_i)$. All functions $t \in T$ are of bounded variation, furthermore considering the class $D := \{f | f \in C(\beta, \gamma) \text{ and } f_b \in T\}$ we can state $T \subset D \subset C := C(\beta, \gamma)$.

It is well known that if $g \in L_1$, which also means $\int_{U^s} |g(\vec{x})| d\vec{x} < \infty$, then there exists for every $\epsilon > 0$ a $t_\epsilon(x) \in T$ such that

$$\int_{U^s} |g(\vec{x}) - t_\epsilon(\vec{x})| d\vec{x} < \epsilon.$$

[1] Thanks to Reinhold Kainhofer for pointing this out.

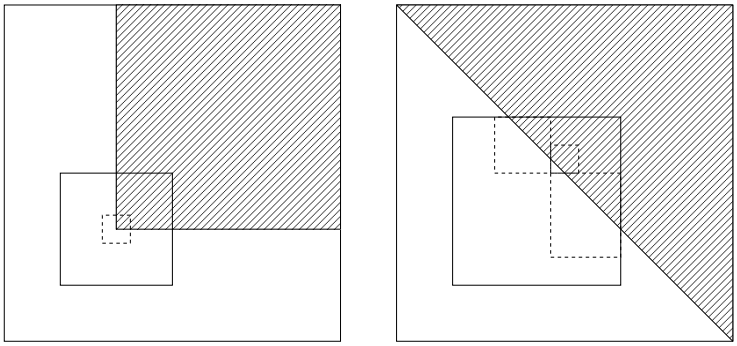


Figure 2.2: Illustration of problems with non axis parallel jump lines.

Also since $T \subset D$ we can easily write

$$\int_{U^s} |g(\vec{x}) - d_\epsilon(\vec{x})| d\vec{x} < \epsilon,$$

with $d_\epsilon(x) \in D$. Generally, the functions $\in D$ will approximate a given function $g \in C \subset L_1$ better than functions $\in T$.

In any case, C is rich since T is rich and $T \subset C$. The restrictiveness of $C(\beta, \gamma)$ is a direct result of the restrictiveness of the variation in the sense of Hardy and Krause.

2.2.2 The function $f(\vec{x}) = \max(x_1, ..., x_s)^{-\beta}$

Now let us consider the function $f = \frac{1}{\max(x_1,...,x_s)^\beta}$ with $0 < \beta < 1$. Certainly $f \notin T$ and $\lim_{\vec{x}\to 0} f(\vec{x}) = \infty$, thus if we can show that $f \in C$ we have $T \subsetneq C$. To do this we need to estimate the integral value and variation of the function f to see if conditions (a) and (b) in Theorem 2.2.1 are met.

Integral Value

Theorem 2.2.2. Let $f = \frac{1}{\max(x_1,...,x_s)^\beta}$ with $0 < \beta < 1$, then $\int_{I^s} f(\vec{x}) d\vec{x} = \frac{s}{s-\beta}$ $0 < \beta < 1$.

Proof. With induction. For $s = 1$ the claim holds since $\int_0^1 x_1^{-\beta} dx_1 = \frac{1}{-\beta+1} x_1^{-\beta+1}|_0^1 = \frac{1}{1-\beta}$.

Now

$$\int_0^1 \dots \int_0^1 \max(x_1, \dots, x_s)^{-\beta} dx_1 \dots dx_s =$$
$$= \int_0^1 \left(\int_0^1 \dots \int_0^1 \max(x_1, \dots, x_s)^{-\beta} dx_1 \dots dx_{s-1} \right) dx_s$$
$$= \int_0^1 \dots \int_0^1 dx_1 \dots dx_{s-1} \int_0^1 \max(x_1, \dots, x_s)^{-\beta} dx_s.$$

Let us consider $\int_0^1 \max(x_1,\dots,x_{s-1},x_s)^{-\beta}dx_s$, and let $\hat{x}_s := \max(x_1,\dots,x_{s-1})$, thus $\int_0^1 \max(x_1,\dots,x_{s-1},x_s)^{-\beta}dx_s = \int_0^1 \max(\hat{x}_s,x_s)^{-\beta}dx_s$ where

$$\max(\hat{x}_s,x_s) = \begin{cases} x_s & x_s \geq \hat{x}_s \\ \hat{x}_s & x_s < \hat{x}_s. \end{cases}$$

Thus

$$\int_0^1 \max(\hat{x}_s,x_s)^{-\beta}dx_s = \int_0^{\hat{x}_s} \hat{x}_s^{-\beta}dx_s + \int_{\hat{x}_s}^1 x_s^{-\beta}dx_s = \frac{-\beta\hat{x}_s^{-\beta+1}+1}{1-\beta}$$

Now we have

$$\int_0^1 \dots \int_0^1 dx_1 \dots dx_{s-1} \int_0^1 \max(x_1,\dots,x_s)^{-\beta}dx_s =$$
$$= \int_0^1 \dots \int_0^1 dx_1 \dots dx_{s-1} \frac{1-\beta\hat{x}_s^{-\beta+1}}{1-\beta} =$$
$$= \frac{1}{1-\beta}\left[1-\beta\int_0^1 \dots \int_0^1 \max(x_1,\dots,x_{s-1}^{-\beta+1}dx_1 \dots dx_{s-1}\right].$$

Now using the induction hypothesis we get

$$\int_0^1 \dots \int_0^1 dx_1 \dots dx_{s-1} \int_0^1 \max(x_1,\dots,x_s)^{-\beta}dx_s =$$
$$= \frac{1}{1-\beta}[1-\beta\frac{s-1}{s-1-\beta+1}] = \frac{s-\beta-s\beta+\beta}{(1-\beta)(s-\beta)} = \frac{s}{s-\beta}.$$

□

Remark 2.2.2. In a similar fashion we obtain $\int_{I^s} \hat{f}_B(\vec{x})d\vec{x} = \frac{s}{s-\beta}(\frac{1}{B})^{\frac{s-\beta}{\beta}}$.

Variation

Definition 2.2.1. Let $T = \{0 = t_0 < \dots < t_{n-1} < t_n = 1\}$ and $Z_s(t_0,\dots,t_n) = \{(z_1,\dots,z_s)|z_k \in T,\ 1 \leq k \leq s\}$. The partition of $I^s = [0,1]^s$ belonging to $Z_s(t_0,\dots,t_n)$ is called valid partition if $P(Z_s(t_0,\dots,t_n)) = \bigcup_{n_1,\dots,n_s}^n [t_{n_1-1},t_{n_1}]\times \dots \times [t_{n_s-1},t_{n_s}]$.

Every partition of I^s can be refined to a valid partition. In the following, $V_V(f)$ will denote the variation of f in the sense of Vitali and $V_{HK}(f)$ will denote the variation in the sense of Hardy and Krause.

Lemma 2.2.1. $V_V(\max(x_1,\dots,x_s)) = 1$ for $(x_1,\dots,x_s) \in I^s$.

Proof. The function $f(x_1,\dots,x_s) = \max(x_1,\dots,x_s)$ fulfills $\max(x_1,\dots,x_{k-1}, x_k,x_{k+1},\dots,x_s) = x_k$ for $\max(x_1,\dots,x_{k-1},x_{k+1},\dots,x_s) \leq x_k$. Let $I_{n_1,\dots,n_s} = [t_{n_1-1},t_{n_1}]\times\dots\times[t_{n_s-1},t_{n_s}]$ be an interval of the valid partition $P(Z_s(t_0,\dots,t_n))$, which doesn't cross the principal diagonal $(x_1,\dots,x_s) = t(1,\dots,1)\, 0 \leq t \leq 1$. Let $V_V(f;I_{n_1,\dots,n_s}) = |\sum_{\tau_1=0}^1 \dots \sum_{\tau_s=0}^1 (-1)^{\tau_1+\dots+\tau_s} \max(t_{n_1-1}+\tau_1(t_{n_1}-t_{n_1-1}),\dots,t_{n_s-1}+\tau_s(t_{n_s}-t_{n_s-1})|$. Since $I_{n_1,\dots,n_s}$ is not on the principal diagonal of I^s there is a k_0, $1 \leq k_0 \leq s$ such that

$$\max(t_{n_1-1}+\tau_1(t_{n_1}-t_{n_1-1}),\dots,t_{n_{k_0}-1}+$$
$$\tau_{k_0}(t_{n_{k_0}}-t_{n_{k_0}-1}),t_{n_{k_0}},t_{n_{k_0}+1}+\tau_{k_0+1}(t_{n_{k_0}}-t_{n_{k_0}-1}),\dots) = t_{n_{k_0}}$$

and

$$\max(t_{n_1-1}+\tau_1(t_{n_1}-t_{n_1-1}),\dots,t_{n_s-1}+\tau_s(t_{n_s}-t_{n_s-1}))=t_{n_{k_0}-1}+\tau_{k_0}(t_{n_{k_0}}-t_{n_{k_0}-1})$$

for all $\tau_1,\dots,\tau_{k_0-1},\tau_{k_0+1},\dots,\tau_s$, $\forall i:\tau_i\in\{0,1\}$.

It follows that

$$\begin{aligned}V_V(f;I_{n_1,\dots,n_s}) &= |\sum_{\tau_1=0}^{1}\dots\sum_{\tau_{k_0-1}=0}^{1}\sum_{\tau_{k_0+1}=0}^{1}\dots\sum_{\tau_s=0}^{1}\\ &\quad (-1)^{\tau_1+\dots+\tau_{k_0-1}+\tau_{k_0+1}+\dots+\tau_s}(t_{n_{k_0}}-t_{n_{k_0}-1})|=\\ &= (t_{n_{k_0}}-t_{n_{k_0}-1})|\sum_{\tau_1=0}^{1}\dots\sum_{\tau_{k_0-1}=0}^{1}\sum_{\tau_{k_0+1}=0}^{1}\dots\sum_{\tau_s=0}^{1}\\ &\quad (-1)^{\tau_1+\dots+\tau_{k_0-1}+\tau_{k_0+1}+\dots+\tau_s}|=0.\end{aligned}$$

If on the other hand $I_{n_1,\dots,n_s}$ lies on the principal diagonal of I^s, then $n_1=\dots=n_s=n_0$ and

$$I_{n_1,\dots,n_s}=I_{n_0,\dots,n_0}=[t_{n_0-1},t_{n_0}]\times[t_{n_0-1},t_{n_0}]\times\dots\times[t_{n_0-1},t_{n_0}],$$

then

$$\begin{aligned}V_V(f;I_{n_0,\dots,n_0}) &= |\sum_{t_1=0}^{1}\dots\sum_{t_s=0}^{1}(-1)^{\tau_1+\dots+\tau_s}\\ &\quad \max(t_{n_0-1}+\tau_1(t_{n_0}-t_{n_0-1}),\dots,t_{n_0-1}+\tau_s(t_{n_0}-t_{n_0-1}))|=\\ &= |\sum_{\tau_1,\dots,\tau_s=0}^{1}(-1)^{\tau_1+\dots+\tau_s}(t_{n_0-1}\max(\tau_1,\dots,\tau_s)(t_{n_0}-t_{n_0-1}))|=\\ &= |t_{n_0-1}\sum_{\tau_1,\dots,\tau_s=0}^{1}(-1)^{\tau_1+\dots+\tau_s}+\\ &\quad (t_{n_0}-t_{n_0-1})(\sum_{\tau_1,\dots,\tau_s=0}^{1}(-1)^{\tau_1+\dots+\tau_s}-\sum_{\tau_1,\dots,\tau_s=0}^{0}(-1)^{\tau_1+\dots+\tau_s})|=\\ &= |t_{n_0-1}0+(t_{n_0}-t_{n_0-1})(0-1)|=t_{n_0}-t_{n_0-1}.\end{aligned}$$

Then holds $V_V(f;I^s)=\sum_{n_0=1}^{n}(t_{n_0}-t_{n_0-1})=1$, where $V_V(f;I^s)$ is attained already at the principal diagonal of I^s.

Remark 2.2.3. If $g(x)$ in $[0,1]$ is monotone or of finite variation $\mathrm{Var}(g)$ then

$$V_V(g(\max(x_1,\dots,x_s);I^s))=|g(1)-g(0)|$$

or

$$V_V(g(\max(x_1,\dots,x_s);I^s))=\mathrm{Var}(g).$$

Remark 2.2.4. Let $0\le a_k<b_k$, $1\le k\le s$ then for $I_{a,b}=\prod_{k=1}^{s}[a_k,b_k]$ we analogously get

$$V_V(g(\max(x_1,\dots,x_s));I_{a,b})=|g(b)-g(a)|$$

for $a_1=\dots=a_s$ and $b_1=\dots=b_s$, otherwise $V_V(g(\max(x_1,\dots,x_s));I_{a,b})=0$. The variation in the sense of Vitali of functions $g(\max(x_1,\dots,x_s))$ is concentrated on the principal diagonal of the unit cube.

Remark 2.2.5. For functions $f = g(\max(x_1, \ldots, x_s))$

$$V_{HK}(f; I^s) = V_V(f; I^s)$$

holds. The variation of Hardy and Krause is defined as

$$V_{HK}(f; I_{a,b}) = \sum_{J \subset I^s} V_V(f; J)$$

where J are all k-dimensional faces $\{(u_1, \ldots, u_s) \in I^s | u_j = 1, j \neq i_1, \ldots, i_k\}$ with $1 \leq k \leq s$ and $1 \leq i_1 < \cdots < i_k \leq s$. Since for $k < s$ there are some $u_j = 1$, the function $f = g(\max(x_1, \ldots, x_{t-1}, 1, x_{t+1}, \ldots, x_s)) = g(1)$, $t \neq i_1, \ldots, i_k$, is constant and consequently $V_V(f; J) = 0$, $\forall J \subsetneq I^s$.

Let now $g(x) = 1/x^\beta$, $0 < x \leq 1$, $0 < \beta < 1$ and $f_\beta(x_1, \ldots, x_s) = 1/\max(x_1, \ldots, x_s)^\beta$, $(x_1, \ldots, x_s) \in I^s$. Now let

$$\hat{f}_{\beta,B} = \begin{cases} 0 & f_\beta(x_1, \ldots, x_s) > B, \max(x_1, \ldots, x_s) < 1/B^\beta = B' \\ f_\beta(x_1, \ldots, x_s) & f_\beta(x_1, \ldots, x_s) \leq B, \max(x_1, \ldots, x_s) \geq 1/B^\beta = B', \end{cases}$$

and

$$\tilde{f}_{\beta,B} = \begin{cases} f_\beta(B', \ldots, B') = B & f_\beta(x_1, \ldots, x_s) > B \\ f_\beta(x_1, \ldots, x_s) & f_\beta(x_1, \ldots, x_s) \leq B, \end{cases}$$

and

$$\chi_{\beta,B} = \begin{cases} B & f_\beta(x_1, \ldots, x_s) > B \\ 0 & f_\beta(x_1, \ldots, x_s) \leq B, \end{cases}$$

and clearly $\tilde{f}_{\beta,B} = \hat{f}_{\beta,B} + \chi_{\beta,B}$. It can be easily seen that $V_V(\chi_{\beta,B}; I^s) = B$ and from the remarks before we know that $V_V(\tilde{f}_{\beta,B}) = |g(0) - g(1)| = B - 1$. Consequently, $V_V(\hat{f}_{\beta,B}; I^s) = V_V(\tilde{f}_{\beta,B} - \chi_{\beta,B}; I^s) \leq V_V(\tilde{f}_{\beta,B}; I^s) + V_V(\chi_{\beta,B}; I^s) = 2B - 1$. □

Thus we have finally shown that $f(\vec{x}) = \max(\vec{x})^{-\beta}$, $0 < \beta < 1$ is in C.

2.3 Experimental Results

First, we want to investigate the behavior of function $f(\vec{x}) = \max(x_1, ..., x_s)^{-\beta}$ as discussed in the last section in numerical experiments. As point sequence we used the Zinterhof sequence [52] (see Chapter 4 for more theoretical results), which is a special case of Weyl sequences defined as follows

$$x_n = (\{ne^{1/1}\}, \ldots, \{ne^{1/s}\}), \;\; n = 1, 2, 3, \ldots,$$

for points $n = 1, 2, ...$ and dimension s. Note that the Zinterhof sequence has certain corner avoidance properties as well, which is due to the high degree of irrationality of the generated points. Caused by corresponding diophantine properties this is true not only for the origin but for all rational points as well.

Figure 2.4 displays the results for the original and integral preserving transformed function (transformed in such a way as to get singularities in the interior of the unit interval as well as on the border, see Fig. 2.3 for an illustration of the transformation)

$$\max = \max(x_1, ..., x_s)^{-0.5}$$

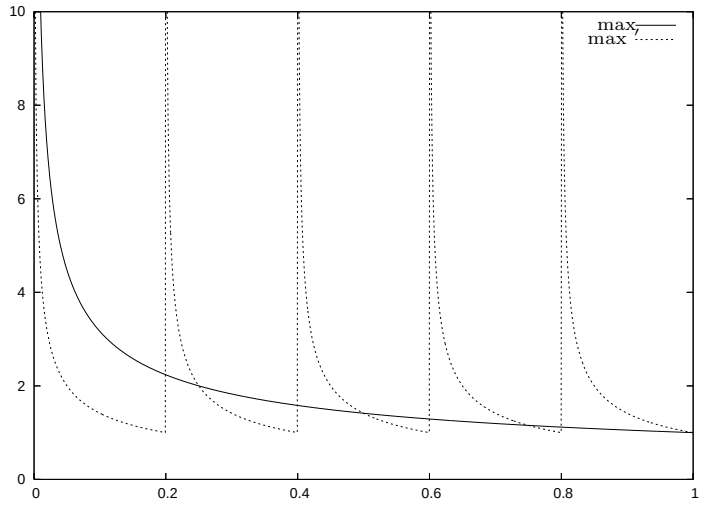


Figure 2.3: Illustration of the transformation of function max to max $'$.

$$\max{}' = \max(\{5x_1\}, ..., \{5x_s\})^{-0.5}$$

respectively, where $\{x\}$ is the remainder of x. We let N run and hold $B = 1000000000$ fixed for dimensions 10 and 15 (labeled `d10` and `d15` respectively). As expected, the the error rates are very good.

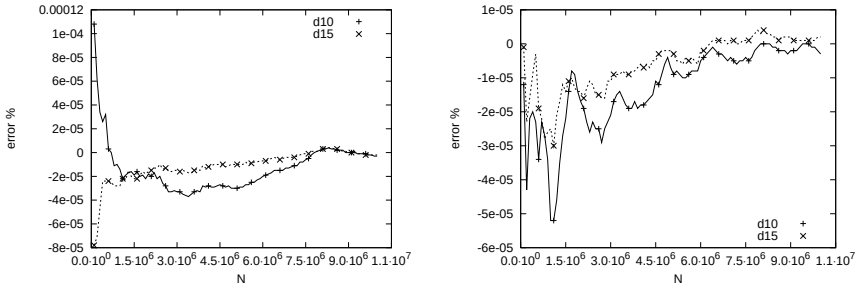


Figure 2.4: Functions max and max$'$ for dimension 10 and 15 with $B = 1000000000$, relative error over N.

Theoretically, we are restricted to functions of class C, practically however the method can be applied to a wider range of functions. Consider the functions

$$f^1 = \prod_{i=1}^{s} \frac{1}{x_i^{0.5}}$$

$$f^2 = \prod_{i=1}^{s} \frac{1}{ln(\frac{1}{x_i})^{0.5}}$$

where f_B^1 and f_B^2 both have non axis parallel jump curves and consequently infinity variation.

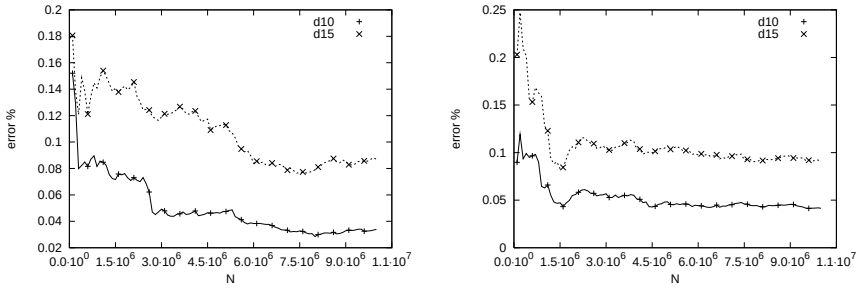
Experimentally, functions f_1 and f_2 can be integrated using our technique, even though their bound representations for this method have infinite variation, c.f. Theorem 2.2.1 condition (b). For a test we used a fixed $B = 1000000000$, which also hints at a serious problem with this method if $f \notin C$. Since the variations of f_B^1 and f_B^2 are infinite we can not obtain a β and thus no optimal bound B using dimension 10 and 15.

Figure 2.5 left hand side shows the results for function f^1 over the number of points N, and the right side shows the results for the integral preserving transformation

$$f'^1 = \prod_{i=1}^{s} \frac{1}{\{5x_i\}^{0.5}}$$

where $\{x\}$ is again the remainder of x.

The figures show that the estimation converges toward a fixed error, this is to be expected since we will by construction always miss $I(\hat{f}_B)$ (see Theorem 2.2.1) since we kept B fixed while it is actually a function of the discrepancy and thus of N. The difference in error between dimension 10 and 15 is a well known phenomenon (curse of dimensionality). However, given that we can somehow obtain the proper bound B for the number of points N used for the integration the error converges even though f^1 (and f_B^1) is of unbounded variation.



f^1 f'^1

Figure 2.5: Functions f^1 and f'^1 for dimension 10 and 15 with $B = 1000000000$, relative error over N.

Keeping the bound B fixed and again using dimensions 10 and 15 we will likely experience problems in the integration when we turn to another function. To illustrate this we used function f^2, and an integral preserving transformation as follows

$$f'^2 = \prod_{i=1}^{s} \frac{1}{ln(\frac{1}{\{5x_i\}})^{0.5}},$$

shown in Fig. 2.6 left and right hand side respectively. When the bound is chosen too low the results usually becomes stable quickly with a high error, stemming again from $\hat{f}_B^2$. In this case the bound was chosen too high, i.e. we would need to use more points N to get to the region where B is optimal. This can be seen from the "overshoots", usually high error rates at the beginning, due to points falling near the jump curve, thus early introducing high values, i.e. close to B, to the estimation. These will usually vanish when the number of points is high enough to get a fine grained sampling of the unit cube but will

stay visible a long time. So while the method works, experimentally, even for functions not in C this poses the problem of estimating a proper B to be used in the integration.

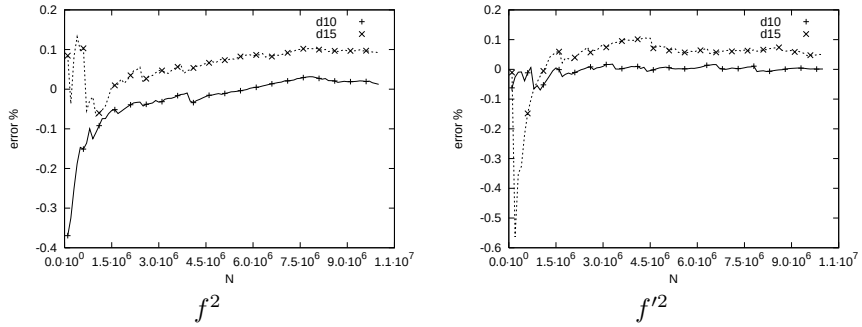


Figure 2.6: Functions f^s and f'^2 for dimension 10 and 15 with $B = 1000000000$, relative error over N.

To illustrate the effect B has on the integration we use a fixed number of points $N = 1000000$ and let B vary. The result of this test, again for functions f^1 and f^2 in dimensions 10 and 15, is given in Fig. 2.7. What can be seen is that the bound depends not only on the number of points but also on the dimension, which is not surprising since it depends on the discrepancy. Also, for f^2, there is an interval of B where the integration holds, while on the other hand we get an increase in error as we move away from that interval. Also, since the bound is depending on the discrepancy, which in turn depends on the number of points and the dimension, the bound is a function of the dimension leading to quite some error in dimension 15 where the approximation was exact for dimension 10.

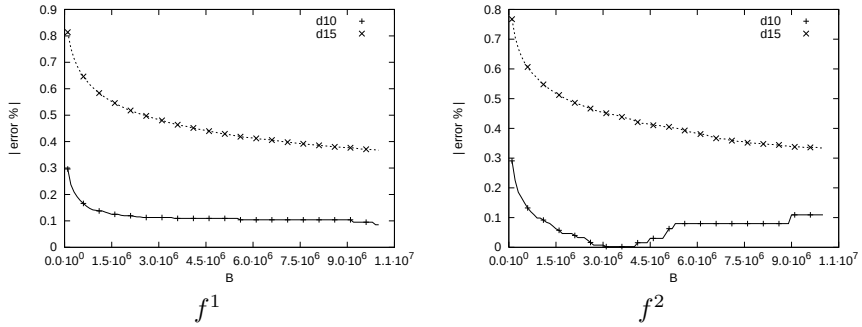


Figure 2.7: Functions f^1 and f^2 for dimension 10 and 15 with $N = 1000000$, relative error over B.

2.4 Conclusion

We have shown that the proposed method can be used to numerically integrate over a rich class of functions $C(\beta, \gamma)$. The method also works experimentally on an even bigger class of functions with the problem that some parameters, i.e. the bound B, can't be chosen specifically for the function. Furthermore, no complicated extensions have to be constructed, the bound can be applied during runtime, and thus the method can be applied to the function directly. Also, the method is not restricted to singularities on the boundary or in the corner. Thus this method is extremely easy to implement and apply, even for non specialists.

However, even if a function is of class C we face the problem that we have to know the number of points beforehand to choose an optimal B. Also, since we need β and γ to choose optimal parameters for the numerical integration the function must be well known. This is theoretically of no importance but practically can prevent (optimal) integration.

Despite these drawbacks this method will be used in experimental setups throughout this work, since other methods have even more severe drawbacks. Specifically, none but two methods can handle singularities in the interior of the unit interval. For the one method the functions has to be transformed to a case where singularities occur only in the origin. For the other method an extension has to be constructed and it is not clear if such an extension can be found for every function. This together with the fact that the construction of an extension is hardly possible for a "user" makes this methods unfit for practical use.

Also note that Klinger methods has some similarities to Zinterhof's method. Klinger removes the nearest point to the origin from the S_N but doesn't reduce the number of points N used. This is essentially the same as setting the function in the elemental interval to 0. Though this is a rather special case, only one singularity in the origin and only one point set to zero, it can still be considered a special case of Zinterhof's results.

Another rather interesting family of methods are those which use corner avoidance properties of sequences. They are even simpler to implement than Zinterhof's method since no modification of the function has to be done and the method can be applied in a straight forward fashion. These methods however entails a restriction on the point sequences which are used, they have to have shown corner avoidance properties. This has to be shown for point sequences prior to use, then however they can be used nearly directly, only the first point has to be dropped for some sequences, since the first point often is in the $(0, 0, \ldots, 0)$-corner. Unfortunately these methods are restricted to functions with singularities in the corner or to functions which can be reduced to such a case. It should also be noted that for singularities in the corner, point sequences with corner avoidance properties and a sufficiently high B Zinterhof's method works similarly.

Chapter 3

Parallelization of QMC Integration Using the GRID

In this chapter we will take a closer look at various point generating sequences in connection with parallelization techniques. It would be beneficial to research and test them separately which unfortunately is not always possible, the study of leaping will make this clear. Also, parametrization requires some special properties of the used sequences and as such can not be applied to just every sequence (in essence we will only apply it to the Zinterhof sequence).

Also it is hardly possible to test all point generating sequences, and as such a couple of sequences which have been repeatedly studied, and are rather well known have been selected. This is done mainly to be able to compare prior results of other authors to the grid specific results. Whenever possible we tried to extend the findings to a general case but again this is often not possible.

3.1 Introduction

GRID environments exhibit challenging properties for numerical integration techniques. This type of computing facility potentially shows extreme heterogeneity in terms of PE speed and network connections, moreover the available computing resources may change in time even during ongoing computations (e.g. new machines may become available or others may get lost due to network problems or maintenance shutdowns, other users may start computations on the same hardware, etc.). Li et al. [29] discuss the use of a GRID service called Integration Service for solving multivariate integration problems, where the PARINT package [9] is used as integration engine.

Therefore, it is not only difficult but impossible to predict the amount of integration nodes required on a single PE under such conditions. Additionally, in practice it is usually not possible to determine a priori the number of integration nodes N necessary to meet a given error requirement. As a consequence, it is of great importance that N may be increased without losing previously calculated function values. It is clear that techniques for providing integration nodes on the single PEs need to be very flexible under these circumstances.

3.1.1 QMC Techniques in GRID Environments

GRID environments exhibit a potentially high heterogeneity in terms of network capacity (i.e. bandwidth and latency) and computing speed (memory capacity, cache sizes, processor speed). In addition to that, these environments are error prone with respect to broken network links or failing PEs. Moreover, additional resources may become available during an ongoing simulation which should be used to optimize resource consumption. As a consequence, the following requirements should be met by a QMC technique employed in a GRID environment:

- Variety in computing speed requires dynamic load balancing capability.
- Variety in network capacity requires load balancing strategies without central organization and a minimal number of control messages exchanged among the computing nodes.
- Failure in hardware resources requires tolerance to lost partial results.
- Additional resources becoming available require a possibility to assign workload to these resources (i.e. by redistributing or redefining workload).

In addition to that, error bounds and computation results should preferably carry over from sequential execution. If the QMC point sets differ between sequential and parallel execution, the quality of the results needs to be investigated thoroughly. Reproducibility is as well an important issue to be considered.

So far, two entirely different strategies have been discussed in literature to employ QMC sequences in parallel and distributed environments.

1. Splitting a given QMC sequence into separately initialized and disjoint parts which are then used independently on the PEs. This strategy comes in two flavors:
   - **Blocking**: p disjoint contiguous blocks of maximal length l of the original sequence are used on the PEs. This is achieved by simply using a different starting point on each PE (e.g., PE_i, $i = 0, \ldots, p-1$, generates the vectors $\mathbf{x}_{il}, \mathbf{x}_{il+1}, \mathbf{x}_{il+2}, \ldots, \mathbf{x}_{il+l-1}$). In case a large number of smaller blocks is used index j is assigned dynamically to PE_i which generates the vectors $\mathbf{x}_j, \mathbf{x}_{j+1}, \ldots, \mathbf{x}_{j+l-1}$ (where j is incremented in steps of size l to avoid overlap).
   - **Leaping**: interleaved streams of the original sequence are used on the PEs. Each PE skips those points consumed by other PEs (*leap-frogging*) (e. g. employing p PEs, PE_i, $i = 0, \ldots, p-1$, generates the vectors $\mathbf{x}_i, \mathbf{x}_{i+p}, \mathbf{x}_{i+2p}, \ldots$).
2. Using inherently independent sequences on the different PEs (denoted as "parametrization" which can be realized for example by randomizations of a given QMC sequences).

Blocking has been suggested in many application focused papers. Mascagni and Karaivanova [32] propose to use disjoint contiguous blocks from Halton, Faure, and Sobol' sequences in the context of solving sparse systems of linear

algebraic equations. Numerical experiments are carried out on a homogeneous cluster using static load distribution. In a second paper [31] the same authors use the suggested techniques for computing extremal eigenvalues, again a QMC sequence is "neatly broken into same-sized subsequences" by blocking. The authors point out that this simple strategy can not be employed in general for all types of simulation settings. Alexandrov et al. [1] use scrambled Sobol' and Halton sequences to solve certain linear algebra systems. They discuss static and dynamic load balancing and point out the importance of efficient dynamic load balancing in GRID environments. Load balancing is done by dynamically distributing chunks (i.e. blocks) of relatively small size to avoid unevenly sized chunks. Techniques for efficiently generating non-adjacent chunks an a single PE are discussed in this paper. Tests are carried out on homogeneous and heterogeneous systems; in the latter case MPICH over Globus-2 GRID software is used. Li and Mascagni [30] propose to extend techniques used in GRID-based Monte Carlo methods, e.g. the *N-out-of-M* scheduling strategy, to QMC sequences by using scrambled quasi random sequences. Furthermore, known statistical properties of MC carry over to scrambled quasi random sequence and thus allowing partial result validation and intermediate value checking. Wan et al. [50] present a parallel strategy for pricing multidimensional American options. In the first stage, the QMC sequence is generated by independently computing equally sized blocks on the PEs using static load distribution. For the second stage two strategies, one being the stochastic mesh method which involves a backward recursion, for data distribution are compared which both correspond to distributing the original sequence in blocks of different size in different manner across the PEs. Tests are conducted on a SGI Onyx machine. Schürer [44] employs equally sized blocks of (t,m,s)-nets on the PEs when comparing QMC integration techniques to adaptive cubature rules. A SGI Power Challenge is used as a test platform. Schmid et. al.[42] have conducted experiments with blocking Niederreiter (t,s)-sequences where large disjoint blocks are used on the PEs. Good reliability of the results has been observed in homogeneous and (simulations of) heterogeneous environments (tests conducted on a SGI Power Challenge). They have also provided theoretical evidence for this good behavior by showing that discrepancy estimates of arbitrary blocks do not degrade as compared to estimates of entire (t,s)-sequences [43].

Leaping has been discussed much more controversial in literature than blocking. Bromley [2] describes a leapfrog parallelization technique to break up the Sobol' sequence into interleaved substreams in an efficient manner. Schmid et. al. have generalized this idea to all types of binary digital (t,s)-sequences [43] in earlier work. Based on these techniques, Li and Mullen [28] use a leapfrog scheme for (t,m,s)-nets to solve financial derivative problems. However, severe problems occur with leapfrog parallelization especially in case of processor speed heterogeneity which results in QMC point sets which do not correspond to sequential computation. Initial results showed that single (t,s)-sequence substreams with leaps of the form 2^n lead to extremely poor numerical integration results whereas this is not the case for leaps of the form 2^{n+1} [42]. Using leaped substreams parallelization in a heterogeneous processor speed environment therefore may lead to severely degraded results as compared to sequential execution when this form of leaping is employed. Different PEs consume a different number of integration nodes and so the poor results of using single substreams are propagated to the parallel results if no synchronization among PEs is performed [12, 42, 43].

Schmid et. al. have also provided theoretical evidence for the observed effects by showing the discrepancy estimated of leaped substreams to be significantly larger as compared to the original sequences [43]. It has also turned out that not only 2^n type substreams are affected by poor quality but these effects occur for many forms of leaps and are highly unpredictable [12, 43].

Parametrization has been proposed as a QMC parallelization strategy by two groups independently. DeDoncker et al. [6, 8, 7] propose randomized (Korobov) lattice and Richtmeyr rules (which are a special type of Weyl sequences), and discuss load distribution strategies for homogeneous and heterogeneous architectures [4]. Results are provided for both, homogeneous and heterogeneous environments, and in both cases result accuracy and execution efficiency was reported to be very well. Ökten and Srinivasan [37] propose to use Halton and scrambled Halton sequences with leaped base sequences on different PEs. Excellent theoretical error estimations are provided and also experimental results for homogeneous as well as for heterogeneous environments exhibit high quality. Parametrization is also compared to blocking and leaping in this work and advantages and disadvantages of the three schemes are analyzed for different application scenarios. Srinivasan [49] confirms the findings of the latter paper and refines the comparison of the three parallelization strategies based on simulation results for pricing financial derivatives.

Based on the requirements for a QMC technique to be useful in GRID environments stated before we try to assess the effectiveness of the three parallel QMC techniques proposed in literature.

Blocking

Two flavors of blocking are discussed. In the first variant, the QMC sequence is partitioned into small blocks which are dynamically distributed among the PEs. Whereas this technique uses QMC node sets almost identical to sequential execution and can handle all types of changing resource scenarios and heterogeneity quite well, it requires the frequent exchange of control messages and is therefore not suited for GRID environments. The validity of this assessment of course depends on the relation between block size and the communication possibilities in the actual GRID environment. In the second blocking variant, one large block is assigned to each PE at the start of the computation. Since the number of QMC points required on each PE is not known a priori the block size needs to be chosen large enough to avoid a PE to exceed the number of available points in its block (exceeding the number would then result in overlap of the blocks which of course degrades the final result). On the other hand, if the blocks have been selected much too large, a significant number of points may not be consumed on slow PEs and the overall point set used exhibits large "gaps" as compared to the sequential case which potentially threatens result accuracy (although the results available so far concerning this effect do not seem to be very severe). Choosing the block size appropriately is therefore a critical issue in this approach. The same considerations of course apply as well if a PE fails. In the case where a specific QMC point set with a limited number of points has been distributed among the available PEs at the start of the computation (e.g. a (t,m,s)-net), handling additional resources is fairly complicated. On the other hand, in the case of using infinite sequences only the next large block in the sequence not being assigned to a PE so far has to be assigned to a PE

which has become available during the computation. Although some of the potential problematic effects (like significant block overlap or large gaps) have not been investigated systematically, the currently available results indicate reliable behavior. The use of large blocks is therefore an interesting option for GRID environments.

Leaping

Contrasting to the blocking case, there is no need to specify a number of points required on each PE since each substream may deliver an infinite number of points in principle. Therefore, there is no danger of running short of points. Also, substream overlap can not occur. In the case of homogeneous environments where care is taken that each PE consumes an equal share of QMC points, a result identical to sequential execution is easily obtained. Note that this is not the case for blocking due to the problems with choosing a good block size (except in case the number of points required is known in advance – which is rarely the case). The situation changes drastically in heterogeneous environments: in case of different PEs consuming a different number of QMC points it has been shown that depending on the type of load imbalance more or less severe degradations in result accuracy are observed. The same considerations (with even more pronounced effects) of course apply as well if a PE fails. When additional resources become available the classical leaping scenario can not handle this situation properly – usually a QMC node set is partitioned into J interleaved substreams if J PEs are available. There is no additional substream available in this scenario. A way to handle this situation is to partition a given QMC point set into $I > J$ substreams in case of J PEs are available. The $I - J$ substreams are not used by default but kept as additional work share in case additional PEs become available. However, neither empirical nor theoretical results are available so far to assess the quality of corresponding results. The use of leaping in GRID environments may be therefore accompanied with problematic side effects which endanger a flexible and transparent use. In addition to that, the most important advantage of leaped substream parallelization as compared to blocking (i.e. in case of synchronized execution the used point set corresponds to the sequential case) does not apply in GRID environments due to the heterogeneity.

Parametrization

The most important difference (and also disadvantage) of parametrization as compared to blocking and leaping is that the QMC point set used in parallel or distributed computation does not correspond to a single (sequentially used) point set. Therefore, the investigation of the results' quality when using this technique is of great importance since it is not clear a priori how results from different point sets will interact in the final result. The findings so far indicate a good quality of the results based on theoretical estimates and empirical tests. Due to the use of de-facto independent QMC point sets on the PEs load balancing at low cost comes for free (each PE generates as much points as it requires locally) and the same is true for reacting to changes with respect to available resources. Therefore, parametrization is a well suited approach for GRID environments provided the quality of the results can be guaranteed. A

disadvantage is that knowledge about the reliability of the results is restricted so far to Halton sequences and (Korobov) lattice rules. An advantage of using randomized QMC techniques in general is that error control techniques like variance reduction are integral parts of this approach. This also holds true for their use in GRID environments.

3.1.2 QMC Node Sets

For generating the Sobol', Halton, Faure and Niederreiter-Xing sequences we use the implementation of the "High-dimensional Integration Library" HIntLib[1]. The descriptions of the Sobol', Faure, and Niederreiter-Xing sequences are taken from [45].

Sobol' Sequence

The Sobol' sequences [47] are digital (t_s, s)-sequences over F_2, where

$$t_s = \sum_{i=1}^{s} (\deg p_i - 1), \tag{3.1}$$

with $p_1 = x \in F_2[x]$ and p_{i+1} denoting the ith primitive polynomial over F_2 ordered by degree.

Sobol' sequences were the first known constructions yielding (t, s)-sequences for arbitrary dimensions s. They were introduced long before the theory of (t, s)-sequences over arbitrary finite fields F_b was established in [34]. However, they only exist for $b = 2$, and even in this case, the resulting t parameter is not optimal for $s > 3$. For $s > 7$, even the Niederreiter sequence, which is equally easy to implement, yields lower t-values.

For $s = 1$ the Sobol' sequence (defined by the polynomial $p_1 = x$) is a $(0, 1)$-sequence identical to the van der Corput sequence in base 2.

We use the implementation of construction 6 in [33].

Faure Sequence

The Faure sequences are digital $(0, s)$-sequences over F_b with b denoting a prime number (original case) or a prime power (general case) greater or equal to s. The case for b prime was shown by Faure [14], the general result is due to Niederreiter [34, Theorem 6.2].

The s infinite generator matrices $C^{(1)}, \ldots, C^{(s)}$ over F_b are defined by $C^{(i)} = (c_{jr}^{(i)})_{j,r>0}$ with

$$c_{jr}^{(i)} = \binom{r}{j} \alpha_i^{r-j}, \tag{3.2}$$

where $\alpha_1, \ldots, \alpha_s$ denote s distinct elements from F_b and the conventions $\alpha^0 = 1$ for all $\alpha \in F_b$ and $\binom{r}{j} = 0$ for $j > r$ are used.

For $\alpha = 1$, the resulting matrix is the infinite Pascal matrix modulo the characteristic of F_b; for $\alpha = 0$, it is the infinite identity matrix. If $s = 1$ and $\alpha_1 = 0$, the resulting $(0, 1)$-sequence is identical to the van der Corput sequence in the same base.

[1] Available at: http://www.cosy.sbg.ac.at/~rschuer/hintlib/

Sequences with the same parameters can also be obtained using Niederreiter sequences or Niederreiter-Xing sequences with rational function fields.

We use the implementation of construction 8 in [33].

Halton Sequence

The construction of the Halton sequence was introduced in [15]. For a dimension $s > 0$, let $b_1, \ldots, b_s$ be integers ≥ 2. Then the Halton sequence in the bases sequence $b_1, \ldots, b_s$ is defined as $x_0, x_1, \ldots$ with

$$x_n = (\Phi_{b_1}(n), \ldots, \Phi_{b_s}(n)) \in I^s, \ \forall n \geq 0, \tag{3.3}$$

where $\Phi_b(n)$ is the radical inverse function. The radical inverse function in base b is defined as

$$\Phi_b(n) = \sum_{i=0}^{\infty} a_i(n) b^{-j-1}, \ \forall n \geq 0, \tag{3.4}$$

where $a_i(n)$ is the ith digit in the digit expansion of n in base b.

Niederreiter-Xing Sequence

In [36] and [51] Niederreiter and Xing develop two methods for creating a digital (t, s)-sequence over F_b based on an algebraic function field with full constant field F_b, genus t, and containing at least $s+1$ rational places. Niederreiter-Xing sequence construction III is constructive, assuming that defining equations for the function field are given and that $s + 1$ rational places are known. We use the implementation of construction 18 in [3].

Weyl Sequence

Weyl sequences are defined by

$$x_n = (\{n\theta_1\}, \{n\theta_2\}, \ldots, \{n\theta_s\}), \ \ n = 1, 2, 3, \ldots$$

where s is the dimension and $\{x\}$ is the fractional part of x. It is well known that a Weyl sequence is uniformly distributed if and only if θ_i are independent irrational numbers. Weyl sequences are used in different variants in literature. An important issue with respect to their quality in terms of uniformity of distribution is the amount or degree of irrationality of the employed starting vector $\Theta = (\theta_1, \ldots, \theta_s)$. Whereas DeDoncker et al. [7] investigate Richtmeyr rules where $\theta_i = \sqrt{p_i}$ with p_i being the i-th prime, we use a special type of Zinterhof sequences [52] where $\theta_i = e^{1/i}$:

$$x_n = (\{ne^{1/1}\}, \ldots, \{ne^{1/s}\}), \ \ n = 1, 2, 3, \ldots, \tag{3.5}$$

These sequences have been shown to exhibit excellent distribution behavior and they excel by their ease of construction and implementation even for non-specialists.

3.1.3 Splitting Feasibility

First let us consider the execution efficiency in a distributed environment of the splitting (leap frogging) and blocking based distribution approach as compared with a synchronized and centralized technique (denoted client/server). In the client/server version we have one PE which generates and distributes the QMC integration points and collects the results (control element CE), and a number of PEs where the function is calculated. For a high number of PEs this model becomes a problem since the only source for new points is the server which can only generate points at a fixed rate. Furthermore, sending the points and results over a network generates a high amount of traffic, thus it is necessary to shift the generation of points from the server to the client. The test is conducted on a heterogenous network consisting of 12 computers (for machine specification see Section 3.1.5). One machine is always setup as CE while the rest are PEs. For leaping the leap size was set to 11 (i.e. the number of PEs) and the block size for the blocking mode was set to 500. The average runtime of five tests was used to calculate the speedup as shown in Table 3.1. We see that for both blocking and leaping the speedup as compared to the client/server version is clearly in a range where splitting can be considered useful.

Table 3.1: Speedup from splitting.

Mode	Speedup	Average Runtime (milliseconds)
client/server	1	6092.2
blocking	3.12	1952.6
leaping	3.27	1861.8

The reason for leaping being slightly better than blocking stems from the fact that in the leaping mode only control messages need to be exchanged while the blocking variant periodically has to request new blocks of QMC nodes. As we have discussed earlier, there is also the possibility to bring the blocking mode closer to leaping by using large blocks.

For the case of parametrization, while not given in the table, we gain the same speedup as with leaping. In terms of distribution this is the same as leaping, to each PE a singe point generating sequence is assigned and no further network messages are necessary for the acquisition of points.

3.1.4 Test Functions

A widely used test function for numerical integration of improper integrals, which was first used by Sobol' in 1973 [48], is

$$t(x) = \frac{1}{x_1^{\alpha_1} \cdots x_s^{\alpha_s}}, \tag{3.6}$$

where s is the dimension and $0 < \alpha_i < 1$, $\forall i \in 1, \ldots, s$. We use a slightly less general version of this test function of the form

$$f(x) = \prod_{i=1}^{s} \frac{1}{x_i^{\alpha}}, \tag{3.7}$$

where s is the dimension and $0 < \alpha < 1$. For $f(x)$ clearly $\int_{(0,1)^s} f(x)dx = \left(\frac{1}{1-\alpha}\right)^s$ and the value of α determines the severity of the singularity, i.e. the gradient of the function (see Fig. 3.1). Additionally to be able to analyze situations where the singularity is inside the interval, as opposed to the origin, we can shift the function by a value $o > 0$ as follows

$$f_o(x) = \prod_{i=1}^{s} \frac{1}{\{x + o\}^\alpha}, \tag{3.8}$$

where $\{x\}$ is the fractional part of x (see Fig. 3.1).

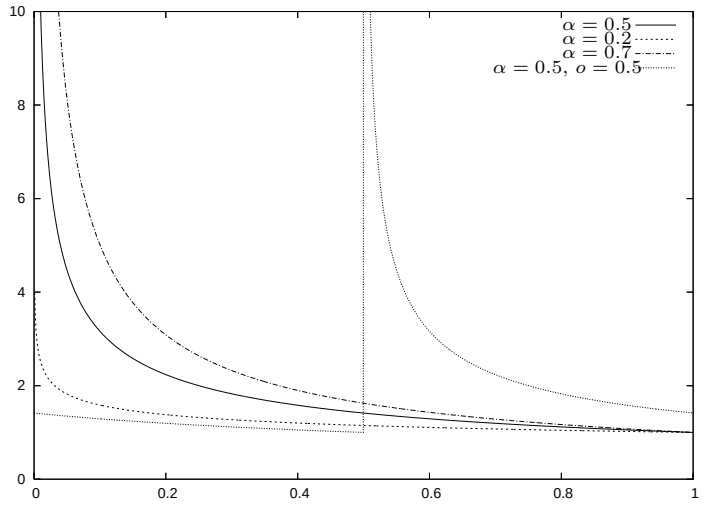


Figure 3.1: Coordinate function of test function with various values of α and shift o as given.

Unless specified otherwise, we use dimension $s = 10$ and no shift in the test function. We evaluate up to $10000000 = 10^7$ integration nodes and set the bound B to 1000000.

This test function is used throughout all experimental evaluations, only the parametrization test use additional functions which are given in Section 3.4.

3.1.5 Test Environment

The hardware platform used in our experiments is a classical heterogeneous cluster architecture. The machines are specified in Table 3.2.

The table contains information concerning the CPU of the machines, and in particular their architecture, speed and memory. The entries in column "stream" is the number of the leaped substream of the QMC sequence used on this machine. "Root" means that no actual integration is done but that this machine is the coordinator of the calculations. In a test on eight machines, machine one is the control node, machine two uses stream 0, machine three uses

Table 3.2: Specification of machines in the heterogenous network.

CPU(arch)	CPU(MHz)	Memory (kB)	stream	machine
AMD Athlon(tm) Processor	1244.719	513852	root	1
AMD Athlon(tm) MP 2800+	2133.468	2064584	stream0	2
AMD Athlon(tm) XP 2800+	2083.121	513852	stream1	3
AMD Athlon(tm) XP 2800+	2083.123	513852	stream2	4
AMD Athlon(tm) XP 2800+	2083.139	255308	stream3	5
AMD Athlon(tm) XP 2800+	2083.134	513852	stream4	6
AMD Athlon(tm) XP 2800+	2083.149	513852	stream5	7
AMD Athlon(tm) XP 2000+	1666.747	255308	stream6	8
AMD Athlon(tm) XP 2000+	1659.642	255308	stream7	9
AMD Athlon(tm) XP 2000+	1659.642	255308	stream8	10
AMD Athlon(tm) XP 2000+	1659.627	255308	stream9	11
AMD Athlon(tm) Processor	1244.732	513852	stream10	12

stream 1, and so on. When for a test the number of machines is stated this table gives information on which machines the streams run on and thus with which relative speed.

In order to simulate a more heterogeneous environment we artificially speed up or slow down single machines to simulate faster or slower machines contributing to the integration process. Due to the potential heterogeneity of GRID environments this speedup is greatly exaggerated; the corresponding machine running faster or slower does so by a factor of 10^3. Additionally, we also compare this extreme case of speedup to a more modest case, where the affected machine consumes twice as much (in case of acceleration) or half the amount (in case of slow down) of QMC points.

3.2 Leaping

As we have seen, with leaping it is possible is to generate separately initialized and disjoint substreams of a given (sequential) sequence of integration nodes. However, there are already a number of shortcomings known for leaped substream parallelization techniques for (t,s)-sequences, e.g. [42, 43, 12].
So here we will cover the following issues:

- So far, only the leaping of Sobol' and Niederreiter (t,s)-sequences has been investigated. Here we additionally treat Faure, Halton, Weyl, and Niederreiter-Xing sequences.

- So far, only the quality differences among differently leaped streams have been investigated systematically. Here we additionally treat the quality differences among equally leaped but differently initialized substreams and the respective implications for the use in distributed computations.

- So far, only a moderate amount of hardware heterogeneity has been investigated. Here we reflect potential GRID properties and additionally treat a case where computing capacity varies by a factor of 10^3 and compare the result to more classical environments.

3.2.1 Single Substream Results

In this section we investigate the integration result accuracy when single leaped substreams are used for generating the integration nodes. The rationale for this is to assess the quality of these substreams – in case of unbalanced load substreams generated on faster PEs will contribute more integration nodes to the overall result than others and consequently their quality will propagate into the result to some extent.

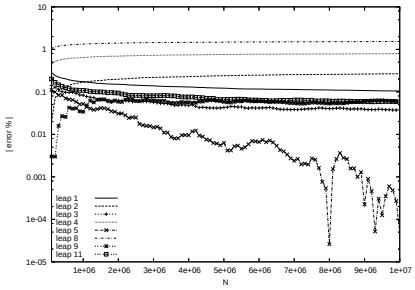
First we focus onto the Sobol' sequence. We aim at showing that the behavior of the Sobol' sequence as documented by Schmid and Uhl [43] can also be seen when integrating functions with singularities. Note that for figures the ordinate gives the absolute value of the error percentage displayed in a logarithmic scale. We investigate the behavior of leaps of the form 2^i (depicted as lines) and 2^i+1 (depicted as lines with points), for $i = 1, \ldots, 6$ (additionally leap 11 is shown for stream 0, see below). For a reference value we display the result of the original sequence (leap 1) with an offset equal to the stream number. Apart from using substreams with different leap values starting with the first point of the sequence ("stream 0"), we also investigate those substreams initialized with the second and third point, respectively ("stream 1" and "stream 2").

The general impression of the results for the Sobol' sequence in Fig. 3.2 confirms the findings of earlier works [42, 43, 12] where stream 0 of Sobol' and binary (t,s)-sequences has shown severe integration result degradation when using leaps of the form 2^i and a lower amount of degradation for leaps of the form 2^i+1, while still giving worse results as compared to the original sequence. However, there are also subtile differences shown in the results. To start with, in case of stream 2 and large leaps we notice that **all** results are superior to the result of the sequential sequence. Leaps of the form $2^i + 1$ do not only perform better as compared to those of the form 2^i but actually do improve the results of the original sequence in almost all cases considered (we notice an even significant improvement in single cases, e.g. stream 0 - leap 5 and leap 33). These effects (and differences to earlier findings) may be due to the fact that we use a test function with singularity located in the origin and that leaping in general reduces the share of points taken from the start of the original sequence (which is known to be of lower quality [25, 41]). It is also interesting to note that there are significant differences among the results of the different streams. The overall trend suggests that result degradations caused by leaping are less severe for streams initialized at some distance from the start of the original sequence.

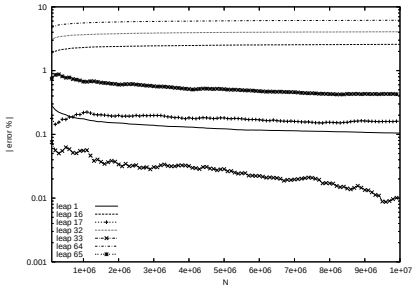
Fig. 3.3 shows the results of the identical experimental setup applied to Halton sequences. A very different behavior is displayed.

Clearly, the observation that leaps of the form $2^i + 1$ perform better than leaps of the form 2^i, as seen in the Sobol' sequence, doesn't hold for the Halton sequence at all. For streams 0 and 2 the leaps of the form 2^i are better, while for stream 1, with exception of 2^{17} and 2^{65}, the streams of the form $2^i + 1$ are better. Overall, leaping has a disastrous effect on the integration results which are severely degraded in almost all cases (only stream 1 - leap 33 and stream 2 - leaps 16,32,64 improve the result of the original sequence). Note that again the results differ a lot among the different streams, especially for larger leaps.

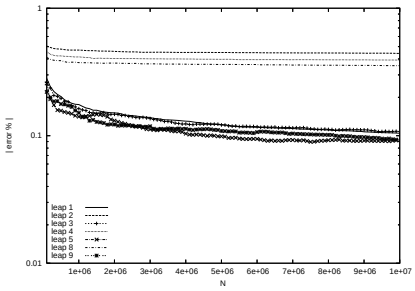
Fig. 3.4 applies the test scenario to the Niederreiter-Xing sequence. Again, significantly different behavior with respect to splitting sensitivity may be observed.



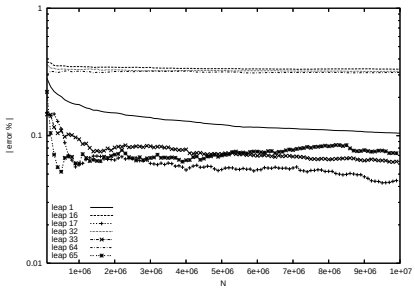
Stream 0, leaps 2–9 and 11



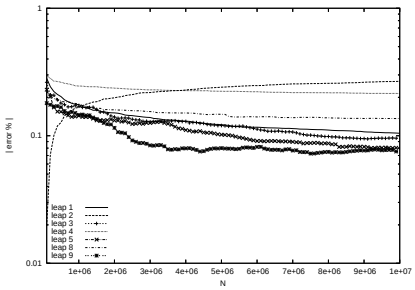
Stream 0, leaps 16–65



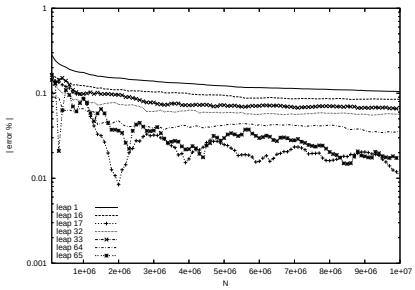
Stream 1, leaps 2–9



Stream 1, leaps 16–65

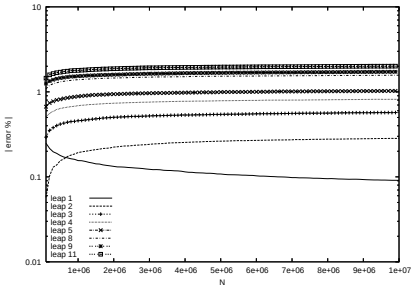


Stream 2, leaps 2–9

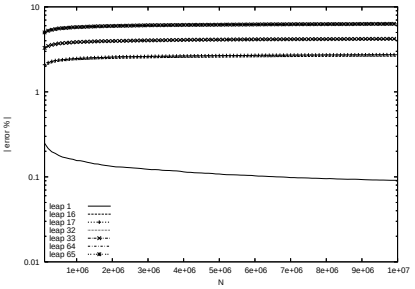


Stream 2, leaps 16–65

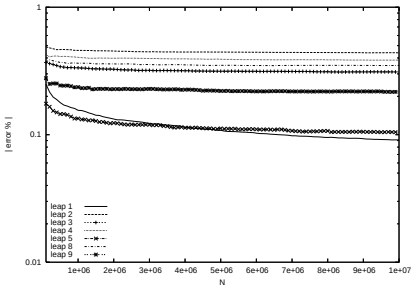
Figure 3.2: Stream behavior for the Sobol' Sequence, streams zero to two.



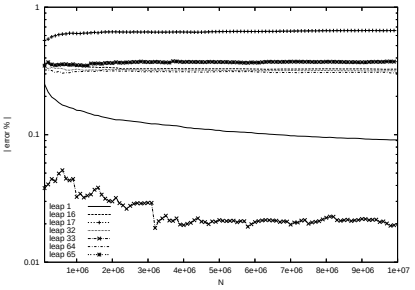
Stream 0, leaps 2–9 and 11



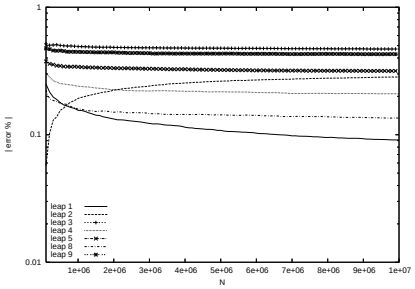
Stream 0, leaps 16–65



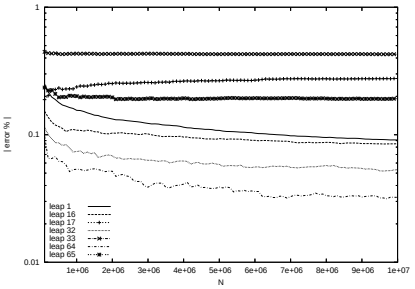
Stream 1, leaps 2–9



Stream 1, leaps 16–65

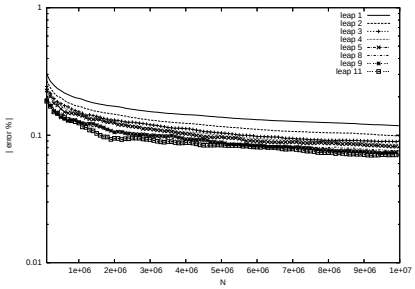


Stream 2, leaps 2–9

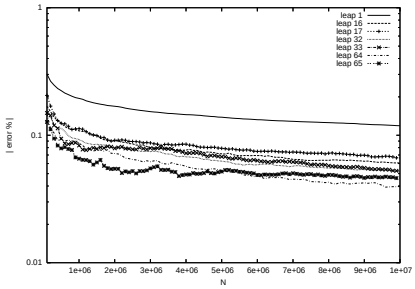


Stream 2, leaps 16–65

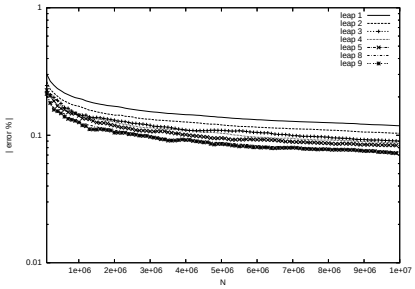
Figure 3.3: Stream behavior for the Halton Sequence, streams zero to two.



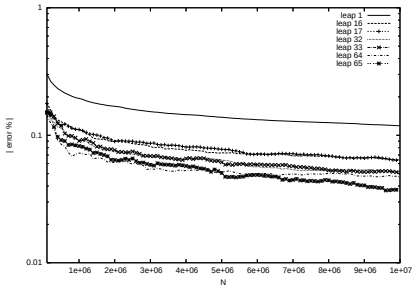
Stream 0, leaps 2–9 and 11



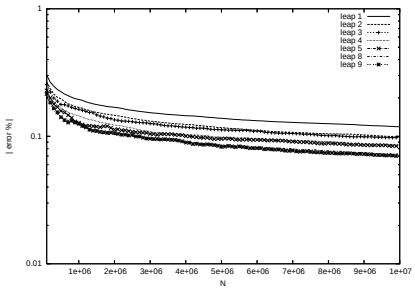
Stream 0, leaps 16–65



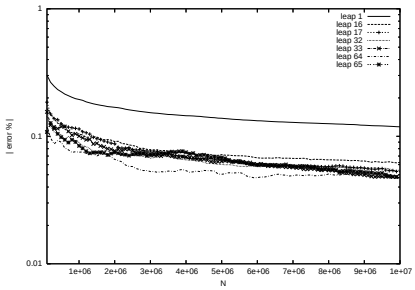
Stream 1, leaps 2–9



Stream 1, leaps 16–65



Stream 2, leaps 2–9

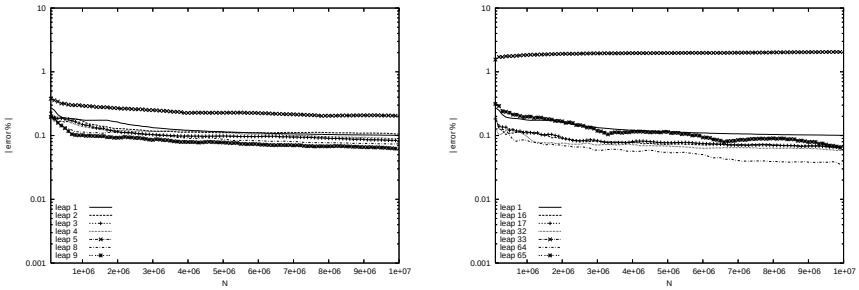


Stream 2, leaps 16–65

Figure 3.4: Stream behavior for the Niederreiter-Xing Sequence, streams zero to two.

An almost opposite behavior to the Halton sequence is found for the Niederreiter-Xing sequence. All leap forms considered improve the results of the original sequence and for larger leaps the improvement is seen to a higher extent. There is no clear relation between leaps of the form 2^i and $2^i + 1$ as seen for Sobol' or Halton sequences.

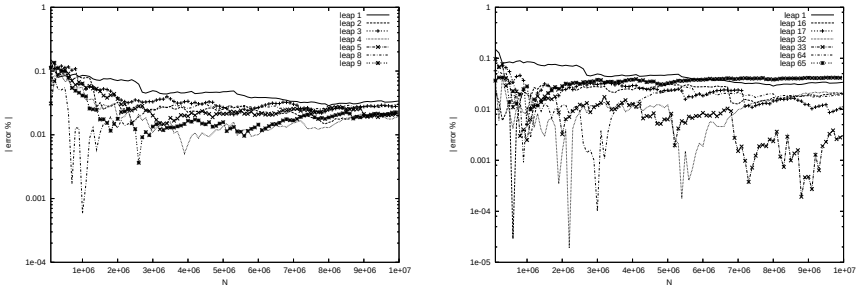
For Faure and Zinterhof sequences we restrict our investigations to stream 0. Fig. 3.5 shows that for the Faure sequence result degradation also occurs but less frequent and on an irregular basis (only leaps 5 and 32 exhibit worse results as compared to the original sequence); in most cases we note a moderate improvement. This is somewhat surprising since the Faure sequence is a very specialized form of Niederreiter-Xing sequences and was expected to behave similarly. This result suggests that also the Niederreiter-Xing sequences might exhibit degraded results under leaping for some specifically selected parameters.



Stream 0, leaps 2–9 Stream 0, leaps 16–65

Figure 3.5: Stream zero behavior for the Faure Sequence.

Finally Fig. 3.6 shows the results of leaping applied to the Zinterhof sequence. Similar to Niederreiter-Xing sequences, we do not find result degradations but improvements as a result of leaping and there is no systematic and significant difference among leaps of different form.



Stream 0, leaps 2–9 Stream 0, leaps 16–65

Figure 3.6: Stream zero behavior for the Zinterhof sequence.

We have seen that the different types of QMC sequences react differently to the use of single substreams. Whereas Niederreiter-Xing and Zinterhof sequences turn out to be very stable and give even slightly better results than the original sequences, Sobol', Halton, and Faure sequences show degraded integration results for some settings. Whereas the effects are somewhat structured and

occur frequently in the case of the Sobol' sequence, the contrary is true for the Faure sequence. The Halton sequence is least suited for leaping and delivers almost consistently significantly worse results as compared to the original. We have also observed that different initialization (different streams) but equal leap form also may lead to significantly different behavior.

Furthermore the behavior of the leaped substreams when used for integrating improper integrals depends also on the position of the singularity. When shifting the singularity from the origin into the unit interval (e.g. in the following example a shift of 0.5 – see Fig. 3.1), the results differ significantly to the original case. Fig. 3.7 shows the results for stream 2 and large leaps of the Halton sequence (see lower right graph of Fig. 3.3 for the original test function).

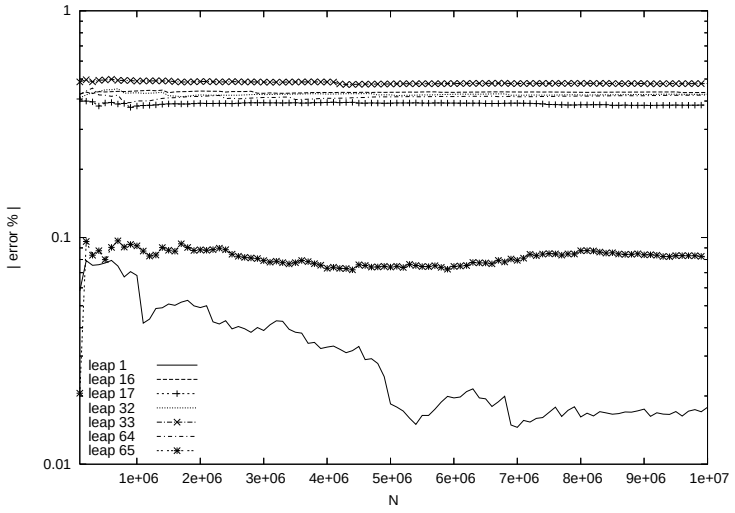


Figure 3.7: Stream two, leaps one and 16–65 of the Halton sequence using the function $f_{0.5}(x)$.

Whereas 2^i leaps improve the Halton sequence results for the original test function, all forms of leaps degrade the results for the shifted version. A possible explanation of this effect is that most QMC sequences avoid the origin in an L-shaped or hyperbolic shaped region [16, 38]. Therefore we are dealing with the more benevolent case when the singularity is situated in the origin. Thus in the general case one can expect further deterioration of the stream when the singularity is moved into the interval.

In the following section, we will investigate the possible impact of the behavior of single substreams on integration results when leaping based QMC point set distribution is used in GRID environments.

3.2.2 Multiple Substream Results

Now we turn to results concerning integration accuracy when using the whole point sequence with leaping as opposed to single streams. To start we again use 11 PEs and consequently a leap factor of 11. Fig. 3.8 displays the results for the Sobol' sequence. The client/server integration result in the figure serves

as the reference value for all subsequent tests as well. For leaping, we use a standard non-synchronized leaping approach (denoted as "leap" in the plots) and the labels "one-fast" and "one-slow" denote the leaping scenarios with one artificially slowed or speed up PE by a factor of 10^3 (applied to stream 0 in either case).

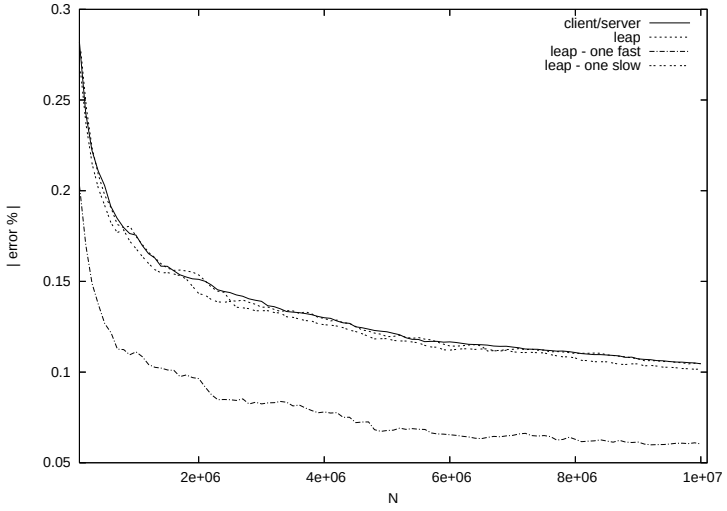


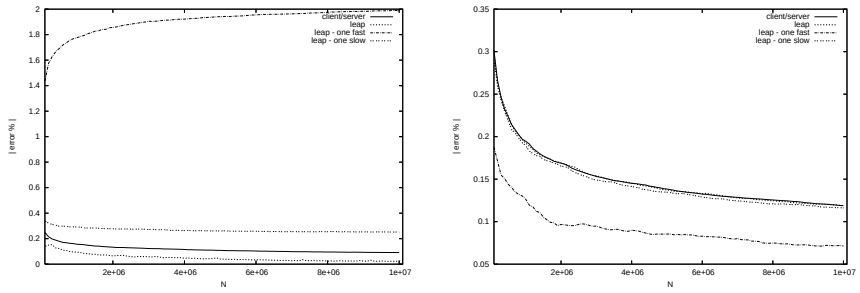
Figure 3.8: Comparison of leaping for the Sobol' sequence utilizing 12 machines.

We see that slowing down one PE significantly and employing leaping without precautions on systems with moderate heterogeneity does not change the integration result in this case. However, when speeding up one PE the integration result is improved. This result corresponds well to the results of single substream investigations where stream 0 of leap 11 also shows better behavior as compared to the original sequence (compare Fig. 3.2 top left graph). Note that speeding up one node by a factor of 10^3 means that the 10 slower nodes only contribute 1% of all QMC integration nodes which explains the propagation of the single substream behavior into the final result. Slowing down one node by the same amount (which in fact means that one stream is missing almost entirely) on the other hand obviously does not affect the result at all.

Fig. 3.9 (left plot) shows the results of the same setting applied to the Halton sequence. We again observe a significant deviation from the client/server result only in case of speeding up one PE, and the deviation is quite severe. This behavior relates to the single substream behavior which shows the same for the single stream 0 (compare Fig. 3.3 top left graph).

Also in the case of the Niederreiter-Xing sequence the single substream behavior of stream 0 of leap 11 (see Fig. 3.4 top left graph) is propagated into the result of the distributed execution with one faster PE (one-fast case) whereas the two other leaping variants are not affected (Fig. 3.9 (right plot)).

As a consequence of these results, changing the leap factor is expected to potentially change the overall result significantly. This is confirmed in Fig. 3.10 where we use 8 PEs (resulting in a leap factor 8) and the Sobol' sequence. Again



Halton Niederreiter-Xing

Figure 3.9: Comparison of leaping for the Halton and Niederreiter-Xing sequences utilizing 12 machines.

in perfect accordance to the results seen before speeding up stream 0 affects the integration result, in this case a severe degradation is observed (as it is expected from a single substream behavior). However, when speeding up stream 0 by a factor of 2 only (denoted as "leap-one fast-double") the integration result is even improved as compared to the client/server reference which does not at all correspond to the former result. Slowing down one PE by a factor of 10^3 results in some degradation in this setting which is also true (but with minor significance) for slowing down by a factor of 2.

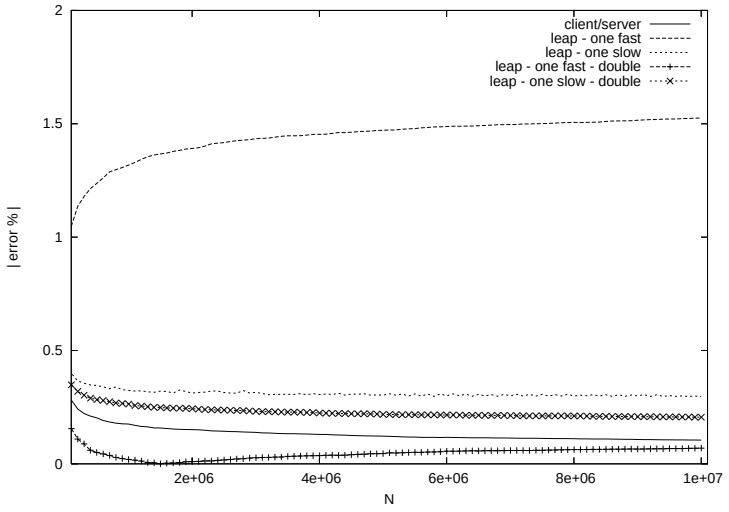


Figure 3.10: Influence of artificial speedup on the results in the case of the Sobol' sequence using nine machines.

When we change the number of machines employed to 10 machines (9 PEs, resulting in leap factor 9), we note in Fig. 3.11 that both types of speeding up stream zero improve the result. Whereas this is to be expected in principle based on the single substream results, the amount of improvement when speeding up by a factor of 10^3 exceeds the improvement of single substream execution.

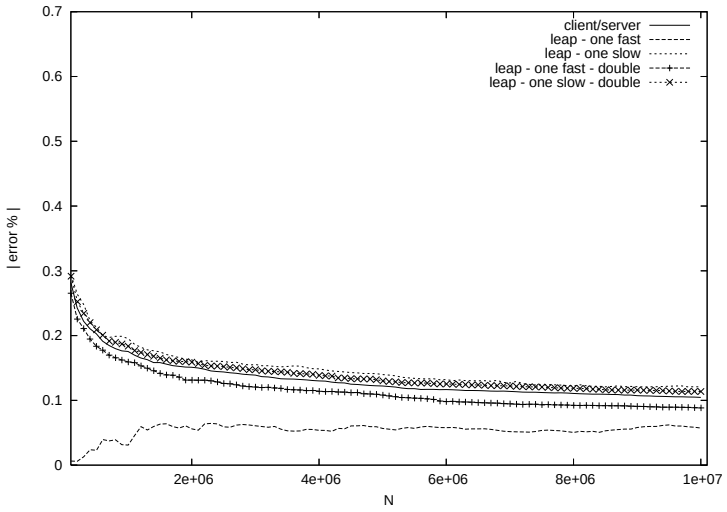


Figure 3.11: Influence of artificial speedup on the results in the case of the Sobol' sequence using ten machines.

Finally we investigate the influence of which stream is affected by a slow down or speed up in distributed execution. Using again 8 PEs but speeding up stream 8 in this case (as opposed to stream 0 as shown in Fig. 3.10) is shown to deliver degraded results as well in the case of the Sobol' sequence (Fig. 3.12), however, the degradation is much less severe in this case. The Niederreiter-Xing sequence does exhibit result improvement again. Note also that for the client/server result the Sobol' sequence is superior to the Niederreiter-Xing sequence.

The results of leaping applied in our experimental setting may be summarized and interpreted as follows:

- Properties of the single substreams are propagated to integration results using multiple substreams in distributed execution to some extent.
- Systems exhibiting moderate heterogeneity or systems with a single failing PE are not severely affected by this phenomenon.
- Systems with a large variety in processing speed may produce very unreliable and poor results when leaping is applied. In most cases these effects can be predicted by analyzing single substream behavior, but not all numerical effects may be explained in this manner sufficiently.

When turning to GRID environments this means that due to the potentially extreme heterogeneity leaping can not be used without extreme care in such environments. While leaping turns out to be able to handle single failing PEs well (the case of "one-slow" in the results), the sensitivity towards different processing speeds is of course a significant problem when QMC node sets with low quality leaped substreams are used. In this case only investigation of single substream behavior and corresponding parameter selection (e.g. choosing leap

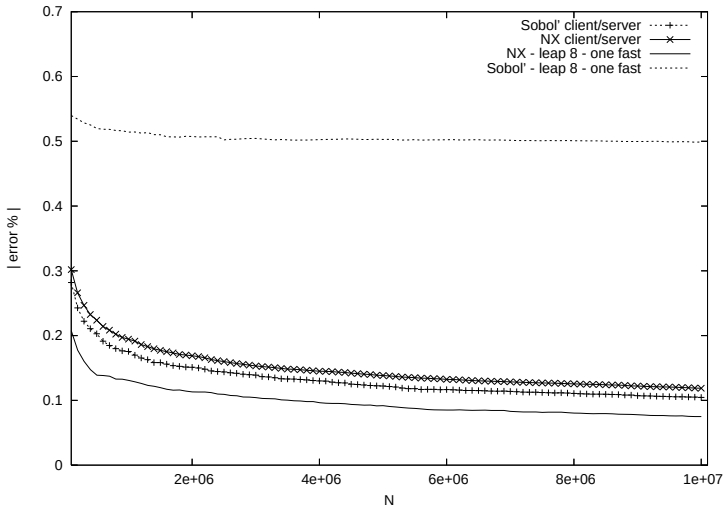


Figure 3.12: Comparison of leaping for the Sobol' and Niederreiter-Xing sequences utilizing nine machines and modifying stream 8.

factors not equal to the number of participating PEs) **or** synchronization among PEs helps out. Since both approaches do not really contribute to facilitate the use of leaping in an efficient and transparent way, other strategies should be used under such conditions.

3.3 Blocking

The distribution of points via blocking entails a network overhead, this overhead can be reduced by using bigger blocks. The basic idea is to use blocks so big that no PE has to request another block. The first test was conducted under the assumption that only one block is used per PE, this can lead to the case where gaps, i.e.: the blocks were chosen too big, or overlaps, i.e.: the blocks were chosen to small, can occur. This is more or less a test of the big block scheme under set conditions in order to understand the influence of gaps and overlaps.
Additionally, we try to ascertain the influence of spare streams when using a leaping point distribution. Spare streams are kept to be able to utilize PEs which become available during computation.

For each of the point sets the following test where executed:
Gap: For this test big blocks were used which were laid out such that there is an unused gap between the blocks. The gap is 20% the size of a block.
Overlap: For the overlap test big blocks are used with a size set to generate less than the overall number of points and no acquisition of new blocks. This resulted in an overlap where about 30% of one block overlaps the following block.
Streamsave: For this test small blocks were used with a point size of 75 and between blocks there is a gap with size 25. This test can also be interpreted

as using synchronized streams (leaping) where the last 25 of the 100 overall streams are reserved for nodes which become available during computation. The synchronization of the streams was used to get a better grasp of the influence of missing streams, this is somewhat artificial since leaping performs somewhat erratic in a heterogenous environment as we have seen earlier.

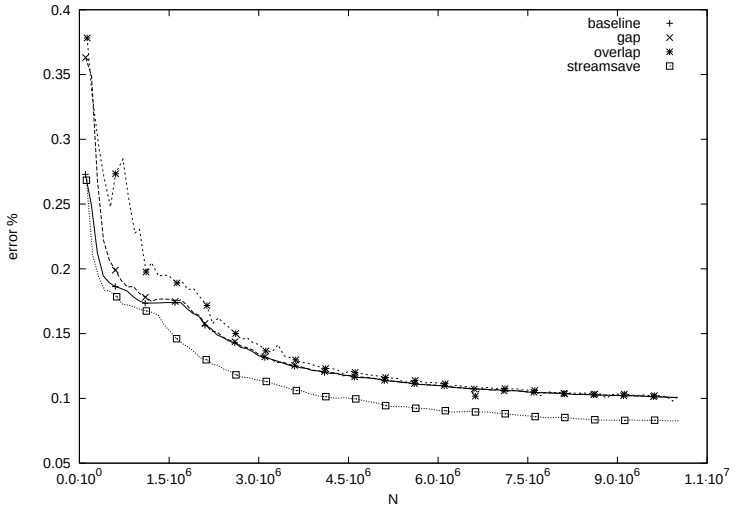


Figure 3.13: Comparison of the Faure sequence regarding overlap, gaps and stream save.

Additionally, a baseline run, calculation of exactly N contiguous points, is given for comparison. The results from the tests are given in Fig. 3.13 to Fig. 3.17, the range of x axis is the same for all graphics but the range of the y axis is chosen so as to discern the effects of a single graph instead of keeping the same range for comparison.

Overall for these test all point sets except Halton behave well, meaning toward higher N they are smooth and differences between the other cases and baseline become negligible and the streamsave case does exceptionally well, even with Halton.

However, apart form the overall poor performance of Halton for gaps and overlaps another effect can be discerned. All sequences, even the otherwise well behaved Niederreiter/Xing sequence show some fluctuation of the results. When comparing this to the baseline run for each sequence we see that Niederreiter/Xing-, Zinterhof- and Faure sequences behave basically the same as the baseline for high N while showing a bit of fluctuation for lower values of N. On the other hand, the Sobol function shows this behavior for all but the highest values of N where it finally becomes smooth like the baseline.

These erratic behavior can become problematic during actual integration, since in an application case the fluctuation of the result would be used as a stop criterion, i.e. if the result stabilizes the calculation is considered finished. The fluctuation as exhibited by the Sobol sequence can lead to an calculation of more points than are necessary, and thus more time consumption.

From these results big blocks should only be used when no gaps or overlaps

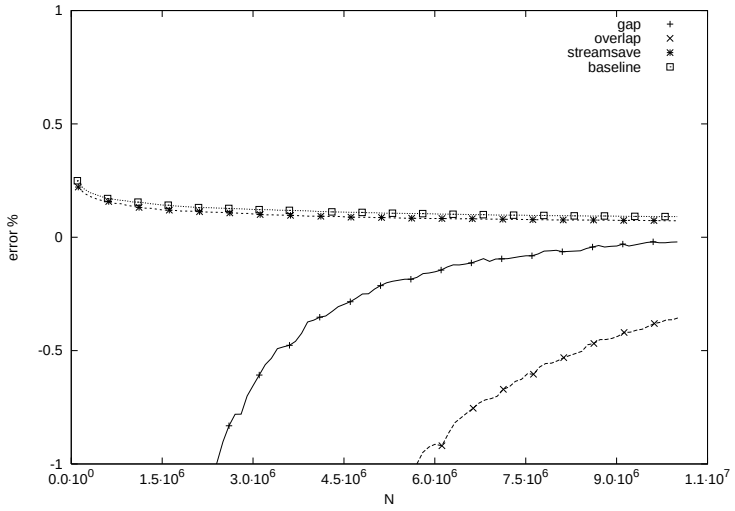


Figure 3.14: Comparison of the Halton sequence regarding overlap, gaps and stream save.

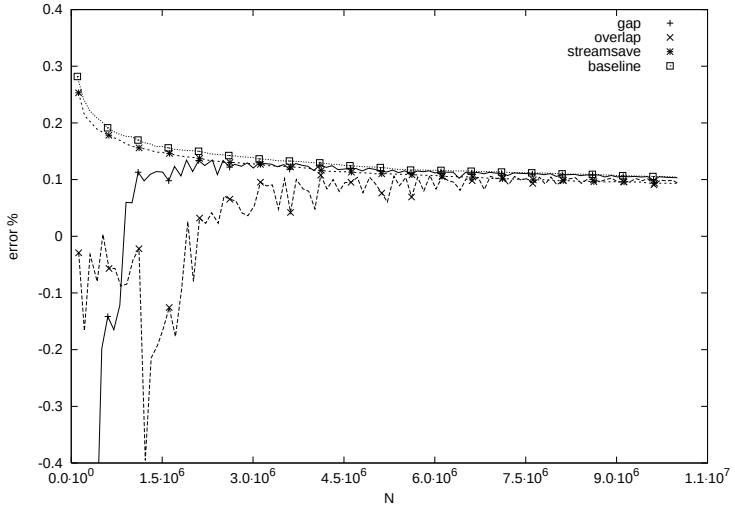


Figure 3.15: Comparison of the Sobol' sequence regarding overlap, gaps and stream save.

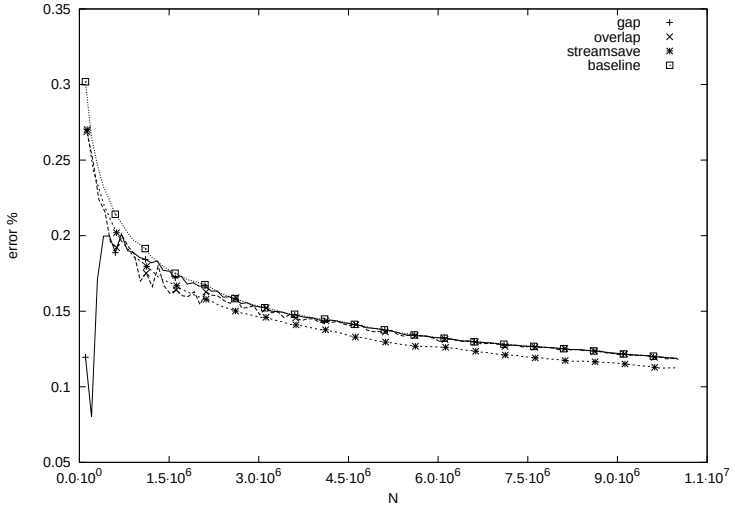


Figure 3.16: Comparison of the Niederreiter/Xing sequence regarding overlap, gaps and stream save.

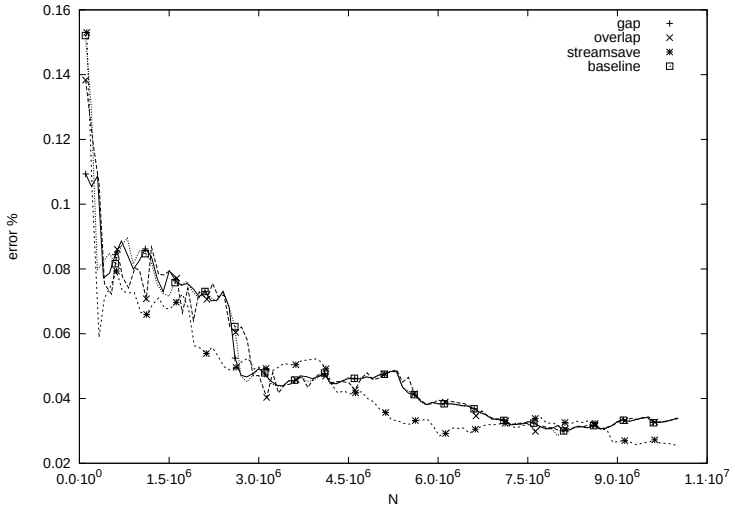


Figure 3.17: Comparison of the Zinterhof sequence regarding overlap, gaps and stream save.

occur or when there is a certainty that a high number of points N has to be used. Also even when there is a certainty for high N a knowledge of the point set is necessary to prevent usage of point sets which show effects like Halton for our test function.

We have dealt with big blocks under rather specific conditions, now we will look at a comparison of big blocks and small blocks. In oder to not simply repeat the above big block cases we will, as suggested in literature, set the block size for big blocks to a high number (in this case equal to N). This will result in the case where each machine uses up about 10% of it's assigned points. In contrast to this we will use a block size of 500 for the small block case. While both are rather extreme they merely serve to get a notion of each end of the spectrum of possible block sizes. The results are given in Fig. 3.18 through Fig. 3.22, again the scale of the x axis is kept constant and the y axis is scaled to give a good view of the results.

The first thing which becomes immediately evident is that parallelization with small blocks is virtually the same as the baseline run, which is a sequential execution of the integration, a result in the next chapter will give a mathematical explanation of this result.

With big blocks two effects are noteworthy. First, different from the above experiments all point sequences generate better results with big blocks, compare Fig. 3.14 and Fig. 3.19. This emphasizes the problem that we do not have knowledge about how the sequences behave under blocking with big blocks, furthermore, we can not infer from one experiment how a given sequence will perform for a different experiment. Second, for all results concerning big blocks the results is dissimilar to the baseline, e.g. sequential run. Thus, we can infer from a sequential run no more information as from a different execution of the big block mode.

Adding this to the findings from the previous experiment, blocking with big blocks should be discouraged if the behavior of the point sequence regarding big blocks is unknown.

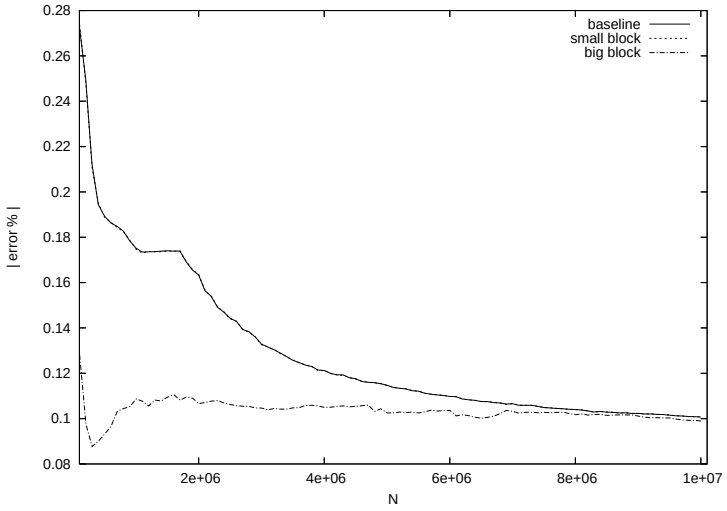


Figure 3.18: Comparison of the Faure sequence regarding big and small block sizes.

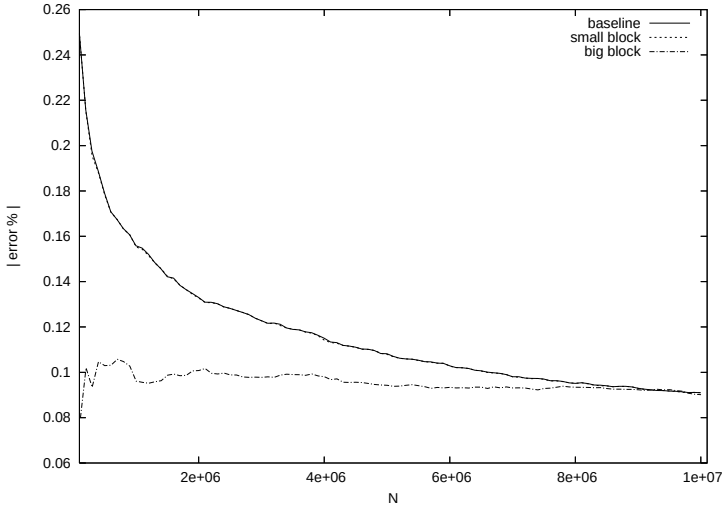


Figure 3.19: Comparison of the Halton sequence regarding big and small block sizes.

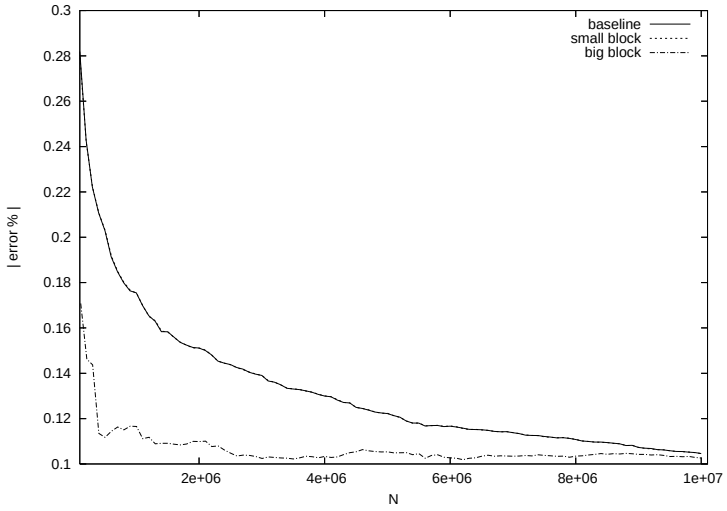


Figure 3.20: Comparison of the Sobol' sequence regarding big and small block sizes.

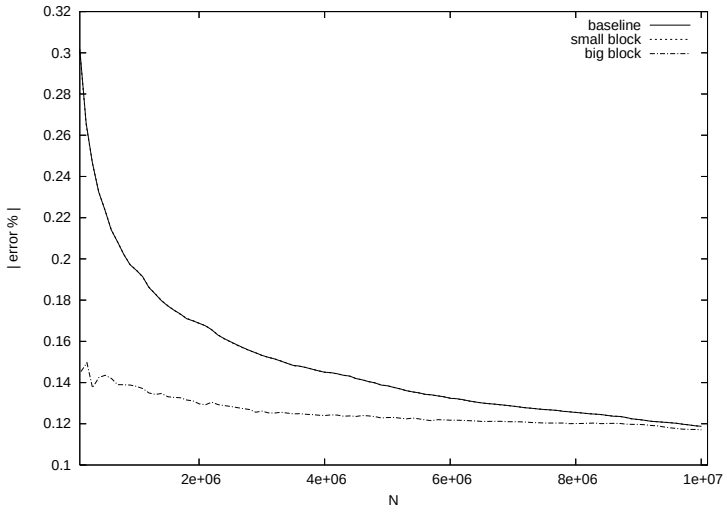


Figure 3.21: Comparison of the Niederreiter/Xing sequence regarding big and small block sizes.

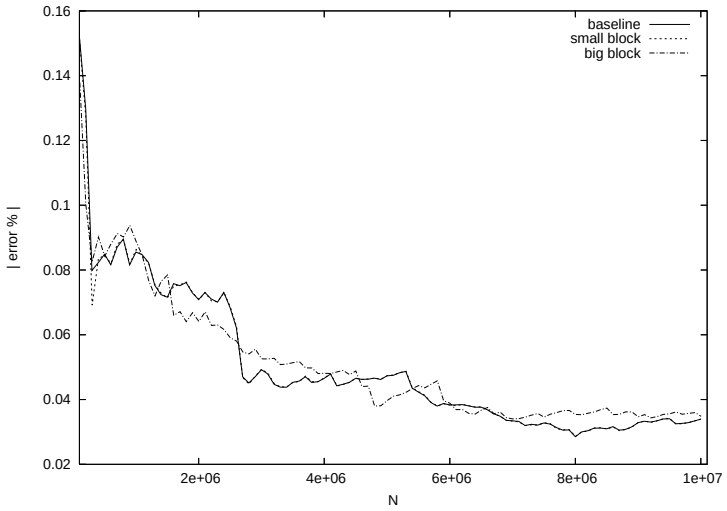


Figure 3.22: Comparison of the Zinterhof sequence regarding big and small block sizes.

3.4 Parametrization

For parametrization we need a number of independent point generating sequences, one per PE. There is a method to utilize the Zinterhof sequence for parametrization which we use. While this method is discussed in a bit more detail in the next chapter it is important to anticipate some results here in order to see if the theoretic results are reflected in the experimental evaluation.

Recall that for the Zinterhof sequence we set $\theta_i = e^{1/i}$ and consequently:

$$x_n = (\{ne^{1/1}\}, \ldots, \{ne^{1/s}\}) \;\; n = 1, 2, 3, \ldots. \tag{3.9}$$

For a parametrization approach however we need P s-dimensional sequences which are independent. This can be achieved by splitting a Ps-dimensional Zinterhof sequence into P s-dimensional sequences by splitting the generator $\vec{\theta}$. Let $\vec{\theta} = (e^{1/1}, \ldots, e^{1/s}, \ldots, e^{1/(P-1)s+1}, \ldots, e^{1/Ps})$ be the generator of the Ps-dimensional sequence than we can obtain the P s-dimensional generators $\vec{\theta_1}, \ldots, \vec{\theta_P}$ as follows

$$\vec{\theta_1} = (e^{1/1}, \ldots, e^{1/s}),$$

$$\vdots$$

$$\vec{\theta_P} = (e^{1/(P-1)s+1}, \ldots, e^{1/Ps}).$$

Let $f_1 = (\{n\theta_1\}, \ldots, \{n\theta_s\})$, …, $f_P = (\{n\theta_{P-1}s + 1\}, \ldots, \{n\theta_{Ps}\})$, $n = 1, \ldots, N$ and let I_P be the $P \times P$ unit matrix with entries $e_{ii} = 1$ and $e_{jk} = 0$ for $j \neq k$ and $i, j, k = 1, \ldots, P$. Let D_N^{Ps} denote the discrepancy for the Ps-dimensional sequence $(\{n\theta_1\}, \ldots, \{n\theta_s\}, \ldots, \{n\theta_{P-1}s + 1\}, \ldots, \{n\theta_{Ps}\})$, $n = 1, \ldots, N$. Then the covariance and correlation matrix for the sequences $f_1, \ldots, f_P$ can be given as

$$\operatorname{cov}(f_1, \ldots, f_P) = \frac{s}{12} \cdot I_P + \mathrm{O}(D_N^{Ps})$$

$$\operatorname{cor}(f_1, \ldots, f_P) = I_P + \mathrm{O}(D_N^{Ps}).$$

And for almost all Weyl sequences we can write

$$\operatorname{cov}(f_1, \ldots, f_P) = \frac{s}{12} \cdot I_P + \mathrm{O}(N^{\epsilon-1}),$$

and

$$\operatorname{cor}(f_1, \ldots, f_P) = I_P + \mathrm{O}(N^{\epsilon-1}), \tag{3.10}$$

where the estimation holds especially for generators of the type $\vec{\theta} = (e^{r_1}, \ldots, e^{r_{Ps}})$ where $r_i \in \mathbb{Q}$ and $r_j \neq r_k$ for $j \leq k$ with $i, j, k = 1, \ldots, Ps$.

This gives us an unlimited number of sequences which can be used for parametrization and a notion of the correlation. What we do not yet have is a estimation of the error. Since parametrization does not conform to a single well known point generating sequence we do not have a discrepancy estimate.

There is however a result concerning convergence rate by Ökten, et.al. [37] as follows.

For P PEs the estimation of the integral is

$$I'_N(f) = \frac{1}{N} \sum_{n=1}^{N} f(x_n) = \sum_{i=1}^{P} \frac{c_i}{N} \frac{1}{c_i} \sum_{n=1}^{c_i} f(x_n^{(i)}) = \sum_{i=1}^{P} \frac{c_i}{N} I'^{(i)}(f),$$

where $N = c_1 + \cdots + c_P$ and the ith sequence contributes the points $x_1^{(i)}, \ldots, x_{c_i}^{(i)}$ to the overall sequence $x_i, \ldots, x_N$. Now if the sequences are independent $I'_N(f)$ becomes an unbiased estimator for the integral $I(f)$ with

$$\begin{aligned} \mathrm{Var}\left[I'_N(f) - I(f)\right] &= E\left|I'_N(f) - I(f)\right|^2 = \\ &= \sum_{i=1}^{P} \left(\frac{c_i}{N}\right)^2 E\left|I'^{(i)}(f) - I(f)\right|^2 \leq \\ &\leq (V(f))^2 \sum_{i=1}^{P} \left(\frac{c_i}{N}\right)^2 (D^*_{c_i})^2, \end{aligned}$$

which for equal speeds simplifies to

$$\mathrm{Var}\left[I'_N(f) - I(f)\right] = (D^*_{N/P})^2 (V(f))^2 / P. \tag{3.11}$$

This gives us enough knowledge about the Zinterhof sequence under parametrization that we can go over to the experimental evaluation.

Two things are apparently of interest, one this the handling of GRID specific problems, e.g. failing machines. The other is the effect on the integration if a higher number of PEs is used. This second problem stems from equation (3.11) where on the one hand the variance is reduced when the number P of PEs rises, and on the other hand the discrepancy, upon which the variance also depends, is increased since the discrepancy is a function of N and an increasing P means a decreasing number of points per PE. Furthermore, we can see from equation (3.10) that for an increasing number of PEs the correlation raises, where low correlation means independent sequences.

To conduct these experiments we use the following functions, all given for dimension s, unless noted otherwise $s = 10$:

$$\begin{aligned} fc1: \quad f(\vec{x}) &= \prod_{i=1}^{s} \frac{1}{x_i^{0.5}}, \\ fc2: \quad f(\vec{x}) &= \prod_{i=1}^{s} \ln(\frac{1}{x_i})^{0.5-1}, \\ mult: \quad f(\vec{x}) &= x_1 x_2 \cdots x_s, \\ gus: \quad f(\vec{x}) &= \sqrt{\frac{45}{4s}} \left(\sum_{i=1}^{s} x_i^2 - \frac{s}{3}\right), \\ hus: \quad f(\vec{x}) &= \prod_{i=i}^{s} \left(x_i^3 - \frac{1}{4} + 1\right). \end{aligned}$$

Note that fc1 is the same function we used with leaping and blocking, it is only repeated here to give a complete overview.

In Fig. 3.23 and Fig. 3.24 we study the influence of failing nodes under parametrization. As reference we use the baseline of sequential Zinterhof sequence of dimension 10. The parametrization as such has no baseline so it's useful to compare it to a similar baseline to estimate the overall performance. The *parametrization* plot is the normal uninfluenced parametrization in a heterogeneous environment. The case of *one-fast* means that one PE did nearly

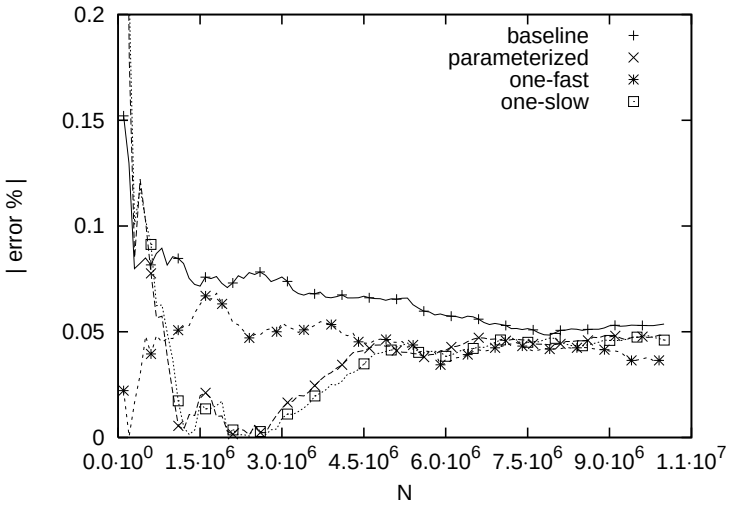


Figure 3.23: Comparison of function fc1 under parametrization in dimension $s = 10$.

all the work, i.e. it was 10^3 times faster than the other PEs, this can happen if a lot of PEs fail along the way of if the come available only later during the calculation. The case *one-slow* is the other end of the spectrum, here one machine fails or becomes available only close to the end of the calculation. Overall we used 12 PEs for both tests with dimension 10, the number of points used is given at the x-axis and the percentage of the error in relation to the exact value of the integral is given at the y-axis.

What can clearly be seen is that the parametrization performs comparable to regular sequential integration, sometimes worse (Fig. 3.24) and sometimes better (Fig. 3.23), but in any case it generates an error of the same magnitude. In the plots it can also be seen that the influence of failing nodes, be it one or many, is negligible. While for a low N it could become problematic, e.g. *one-fast* in Fig. 3.23, since the error is unduly high, but as soon as a moderate number of points is used this behavior vanishes. The same results show for the bounded functions mult (Fig.3.25), gus (Fig.3.26) and hus (Fig.3.27) for dimensions 10, 15 and 20 respectively. All tests show that the usually problematic case of failing machines has little influence in the overall result.

However, the convergence rate can be reduced under parametrization, Fig. 3.26 is an extreme example. While it is clear that for higher dimensions the convergence rate is reduced, the difference between the baseline and the parametrization results in this case is quite huge.

We also want to look at the behavior of the integration when we use more PEs. For this test we used PEs with equal speeds so that we can concentrate on the effects of the changing number of PEs rather than the heterogeneity of the network. Even though we used PEs with similar speed we did not synchronize so a slight discrepancy of points calculated per PE is given, however the deviation was small. We kept the number of points N set to 5000000 and 10000000 and dimension $s = 10$ for functions fc1 and fc2 (Fig. 3.28) and mult, hus and gus

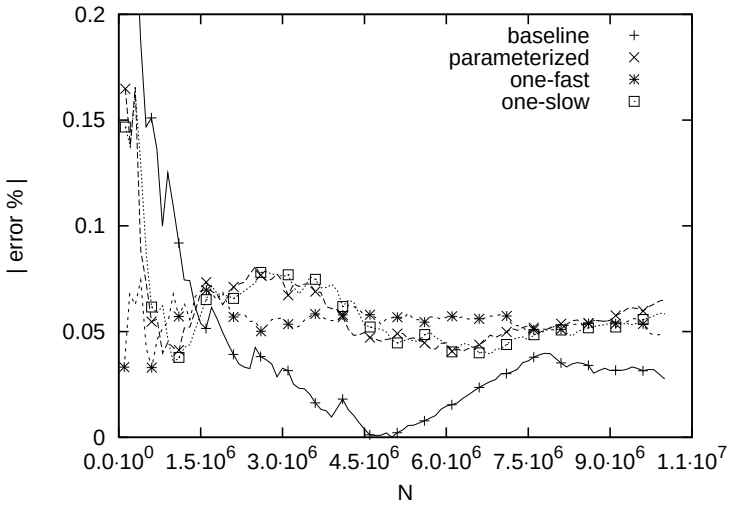


Figure 3.24: Comparison of function fc2 under parametrization in dimension $s = 10$.

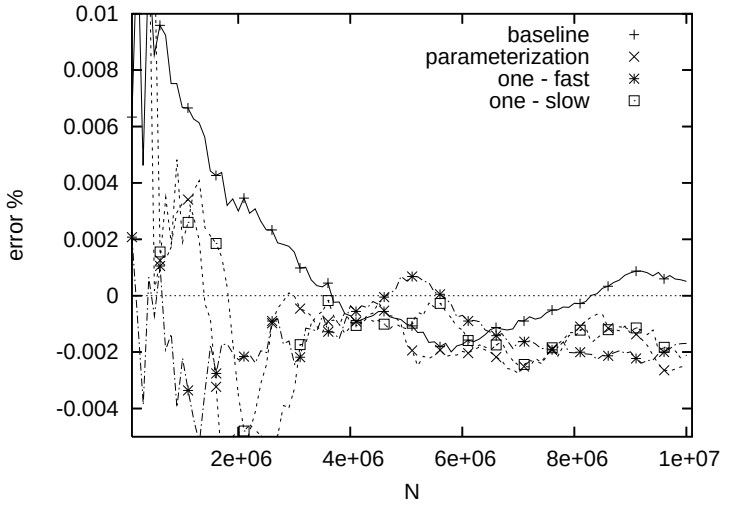


Figure 3.25: Comparison of function mult under parametrization in dimension $s = 10$.

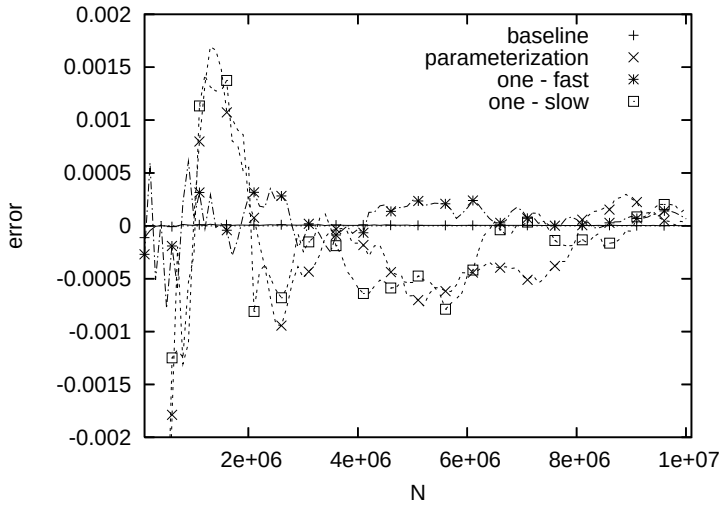


Figure 3.26: Comparison of function gus under parametrization in dimension $s = 15$.

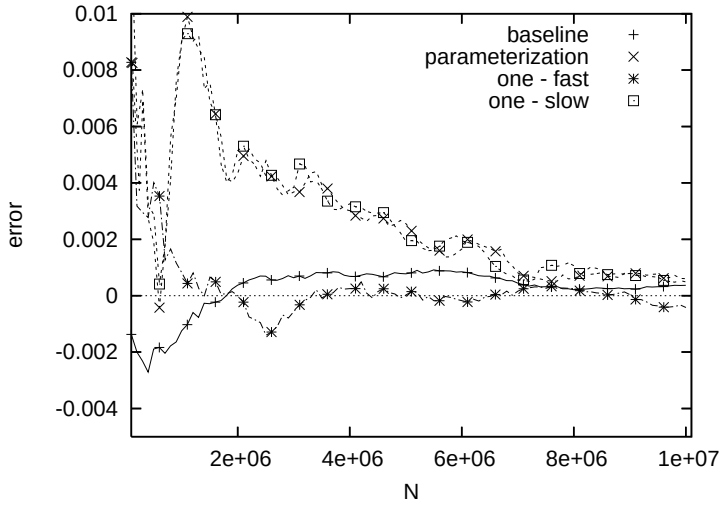
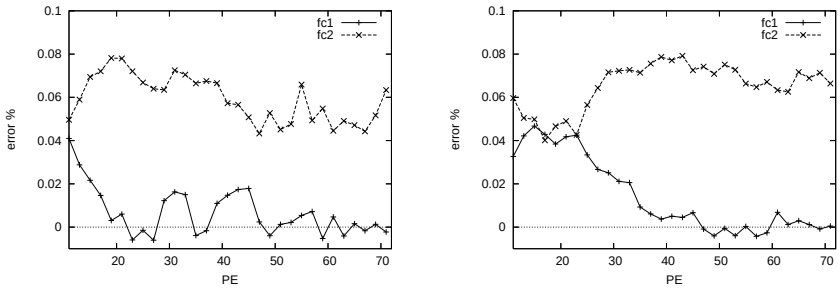
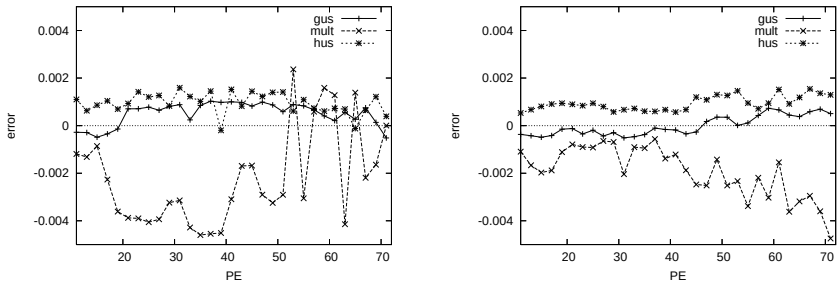


Figure 3.27: Comparison of function hus under parametrization in dimension $s = 20$.



$N = 5000000$ $N = 10000000$

Figure 3.28: Comparing the integration error of functions fc1 and fc2 when the number of PEs is increased under parametrization for a fixed N.



$N = 5000000$ $N = 10000000$

Figure 3.29: Comparing the integration error of functions hus, gus and mult when the number of PEs is increased under parametrization for a fixed N.

(Fig. 3.29). Please note that in Fig. 3.29 the value of mult is a relative error rather than an absolute error, it was still included in this figure for reasons of ordinate scale. The number of machines used was in the range of 12–72, one machine was used as a root element, so the actual number of PEs went from 11 to 71, which is given on the abscissa. On the ordinate the error is displayed, for the functions gus and hus this is the absolute error, since the integral of these functions is 0 and we can not give a relative error. For the functions mult, fc1 and fc2 the errors are given as a percentage of the integral value of the function.

Overall what we see is that the error either stays roughly the same in most cases. However, in the case of *fc1* it even decreases and in the case of *mult* for N=10000000 it increases. Furthermore, a higher number of PEs can possibly degrade the result, and since it can not be predicted for which functions this will happen this is a severe drawback.

3.5 Conclusion

We have studied leaping, blocking and parametrization in some detail as well as various point generating sequences, the following list summarizes the findings.

- **Leaping**
  Leaped substreams of different types of QMC point sets have turned out to behave quite differently. Whereas Sobol', Halton, and Faure sequences exhibit single leaped substreams with extremely low quality, Niederreiter-Xing and Zinterhof sequences are shown to behave very stable under leaping in our experiments.

  On systems with a large amount of heterogeneity in terms of processing speed, the properties of single substreams may be reflected in the final result (which is of course a problem in case of substreams with extremely low quality like Halton, Sobol', and Faure sequences). This effect does not occur in systems with moderate heterogeneity and in systems with single failing PEs.

  Consequently, using leaped substreams of QMC sequences on single PEs as a general strategy in GRID environments can not be recommended. Our results however indicate that certain types of sequences may be employed in this scenario without taking specific precautions (Niederreiter-Xing and Zinterhof sequences) and we have demonstrated that a moderate amount of heterogeneity does not lead to severe result degradation, even in scenarios where low quality substreams are used.

- **Blocking**
  Small block sizes can nearly achieve the same speedup as leaping when disregarding network load. Big blocks on the other hand have the potential of the same speedup and low network use as leaping. However, big blocks can lead to a higher number of processed points when the block size is not chosen carefully. In fact a good choice of block size for big block is nearly impossible since the amount of points N used for the integration is not known a priori, and even if it is known the acquisition of new nodes during runtime can result in gaps.

  Overall the usage of small blocks with a proper method of preventing delay due to a lack of points while requesting a new block is preferable, unless the streams for leaping are synchronized or the behavior of the point set concerning gaps or overlaps is known.

- **Parametrization**
  We discussed the properties of parametrization with regard to the GRID, and made a point that of all parallelization schemes it can handle the GRID environment best. Additionally, since the Zinterhof sequence is computationally efficient (and extraordinarily simple to implement) we are able to utilize the GRID to it's maximum extent for a given task without further overhead. We showed, theoretically and experimentally, that the proposed parametrization scheme for the Zinterhof sequence is sound. However, the experiments and the error estimation, i.e. a probabilistic error bound instead of a discrete, show that the convergence rate of the numerical problem can decrease. Additionally, an increase in the number of PEs can lead to a increase in the integration error. This is a serious drawback since computational power in the GRID is primarily derived from the utilization of a huge number of machines.

Even though leaping and parametrization have distinct advantages blocking is overall the best method. And under the assumption of blocking we have seen that while small block sizes work well for all point sequences this is not the case with big blocks. Furthermore, several point sequences show defects that make them inapt for the application in the GRID. From all point sequences only Niederreiter/Xing and Zinterhof sequences have shown persistent fitness. Only these two should be used, at least from the sequences which were considered here. The Niederreiter/Xing showed at times a better behavior regarding fluctuation of the results, the Zinterhof sequence on the other hand is faster to calculate and easier to implement, so either is a fair choice.

Chapter 4

Theoretic Results for Zinterhof Sequences in the GRID

We have seen that when comparing this sequence to others they have either the drawback of being computationally expensive (e.g. Niederreiter/Xing sequence) or have some kind of problematic defect (e.g. Sobol' sequence). In contrast, the Zinterhof sequence behaves well and is computationally inexpensive. What remains is to look at it theoretically instead of experimentally, which we will do in this chapter.

4.1 Fundamentals

This chapter deals mainly with the Zinterhof sequence which was defined in Section 3.1.2. In the following a slightly different view on numerical integration will be presented. We will use a Fourier series expansion because this makes it easier to compute a comparison. The rational why we only look at a comparison is to give a difference in error as compared to the original sequence as opposed to calculating the discrepancy for blocking or leaping, which is rather difficult.

4.1.1 A Different View on Numerical Integration

Consider for dimension s the Fourier series expansion of the function $f(\vec{x})$ to be numerically integrated

$$f(\vec{x}) = \sum_{m_1,\ldots,m_s=-\infty}^{\infty} C(\vec{m}) e^{2\pi i(m_1 x_1 + \cdots + m_s x_s)}, \tag{4.1}$$

with the integration error

$$R_N(\vec{\theta}) = \frac{1}{N}\sum_{n=1}^{N} f(n\vec{\theta}) - \int_0^1 \cdots \int_0^1 f(\vec{x}) dx_1, \ldots, dx_s \tag{4.2}$$

where $\vec{x} = (x_1, \ldots, x_s)$, $\vec{m} = (m_1, \ldots, m_s)$ and $\vec{\theta} = (\theta_1, \ldots, \theta_s)$.

For absolute convergent Fourier series the error is

$$R_N(\vec{\theta}) = \sum_{\substack{m_1,\ldots,m_s=-\infty \\ \vec{m}\neq 0}}^{\infty} C(\vec{m}) \frac{1}{N} \sum_{n=1}^{N} e^{2\pi i(\theta_1 m_1 + \cdots + \theta_s m_s)n} = \sum_{\vec{m}\neq\vec{0}} C(\vec{m}) S_N(\vec{\theta}) \ .$$

Thus to determine the quality of the integration method we have to estimate $S_N(\vec{\theta})$. Clearly $\theta_1, \ldots, \theta_s$ must be rational independent unless $\theta_1 m_1 + \cdots + \theta_s m_s \in \mathbb{Z}$ and thus $S_N(\vec{\theta}) = 1$. Furthermore, by using Weyl's criterion, we know that for independent irrational numbers $\vec{\theta}$ it holds that

$$\lim_{N\to\infty} S_N(\vec{\theta}) \to 0 \quad \forall \vec{m} \in \mathbb{Z}^s \backslash \{0\}.$$

Since $S_N(\vec{\theta})$ is a geometric series we can write

$$S_N(\vec{\theta}) = \frac{1}{N} e^{2\pi i(m_1\theta_1 + \cdots + m_s)} \frac{1 - e^{2\pi i(m_1\theta_1 + \cdots + m_s)N}}{1 - e^{2\pi i(m_1\theta_1 + \cdots + m_s)}}.$$

For the rational independent $\theta_1, \ldots, \theta_s$ with the equality $e^{ix} = \cos(x) + i\sin(x)$ and the basic approximation $|\sin(\pi x)| \geq 2 \ll x \gg$, where $\ll x \gg$ is the distance of x to the nearest integer, we can approximate

$$|S_N(\vec{\theta})| \leq 1/N \frac{1}{2 \ll m_1\theta_1 + \cdots + m_s\theta_s \gg} \ .$$

Consider for $\alpha > 1$ the class $E_\alpha^s(C) = \{f(\vec{x}) : |C(\vec{m})| \leq \frac{C}{||\vec{m}||^\alpha}\}$ then

$$|R_N(\vec{\theta})| \leq 1/N \sum_{\vec{m}\neq\vec{0}} \frac{C}{||\vec{m}||^\alpha} \frac{1}{2 \ll m_1\theta_1 + \cdots + m_s\theta_s \gg}$$

where $||\vec{m}|| = \prod_{i=1}^{s} \max(1, |m_i|)$.

Now for $\theta_1 = e^{r_1}, \ldots, \theta_s = e^{r_s}$, $r_i \neq r_j$ for $i \neq j$, $r_i \in \mathbb{Q}$ the subsequent result follows from an approximation by A. Baker (c.f. [23]):

$$\ll m_1\theta_1 + \ldots + m_s\theta_s \gg \geq \frac{C(\vec{\theta})}{||\vec{m}||\psi(\vec{m})} \ ,$$

where $\psi(\vec{m})$ weakly converges towards ∞ for $||\vec{m}|| \to \infty$.

Since there is no irrational vector $\vec{\theta}$ such that for all $\vec{m} \ll m_1\theta_1 + \cdots + m_s\theta_s \gg \geq \frac{C(\vec{\theta})}{||\vec{m}||}$ holds, we obtain the final error approximation for $\alpha > 2$ (Zinterhof provides the same error magnitude even for $\alpha > 1$ [52])

$$|R_N(\vec{\theta})| \leq 1/N \sum_{\vec{m}\neq\vec{0}} \frac{C||\vec{m}||}{||\vec{m}||^\alpha} \frac{\psi(\vec{m})}{2C(\vec{\theta})} \ .$$

4.2 Leaping

For estimating the integration error resulting from using leaped Zinterhof sequences, we replace $\vec{\theta}$ in equation (4.2) by $L\theta_1, \ldots, L\theta_s$ for leap size $L \in \mathbb{N}$. Then instead of $S_N(\vec{\theta})$ we have

$$S_N(L\vec{\theta}) = \frac{1}{N} \sum_{n=1}^{N} e^{2\pi i(Lm_1\theta_1 + \cdots + Lm_s\theta_s)n}.$$

By analogy to the general case we can approximate the integration error, however this approximation is worse since instead of $\vec{m}$ we now have $L\vec{m}$ in all formulas. Thus with $||L\vec{m}|| = \prod_{i=1}^{s} \max(1, |Lm_i|) \leq L^s ||\vec{m}||$ and $\psi(L\vec{m})$ instead of $\psi(\vec{m})$ we get

$$|R_N(L\vec{\theta})| \leq 1/N \sum_{\vec{m}\neq\vec{0}} \frac{L^s C||\vec{m}||}{||\vec{m}||^\alpha} \frac{\psi(L\vec{m})}{2C(\vec{\theta})} \ .$$

Considering that $\psi(\vec{m})$ grows only logarithmically for $\vec{\theta} = (\theta_1^{r_1}, \ldots, \theta_s^{r_s})$ and likewise for $\psi(L\vec{m})$ the difference of $\psi(\vec{m})$ to $\psi(L\vec{m})$ plays hardly any role. Thus the error approximation for leaping with leap size L is worse by the factor L^s than the error approximation for the unleaped sequence. This indicates a potentially significant deterioration of the results independent of the specific leap value, as opposed to leaped (t,s)-sequences where only substreams of leap size 2^n have been shown to have poor discrepancy [43].

4.3 Blocking

Again, consider the Fourier series given in Equation (4.1) and the error given in Equation (4.2) with the same parameters.

Then we have for $f \in E_\alpha^s(C)$ with $\alpha > 3/2$ and $x_1, \ldots, x_N \in I^s := [0,1]^s$ the approximation ([10, Theorem 1.35])

$$\left| \frac{1}{N} \sum_{n=1}^{N} f(x_n) - \int_{I^s} f(\vec{x}) dx_1, \ldots, dx_s \right| \leq C \left(\frac{4\alpha - 4}{2\alpha - 3} \right)^{s/2} F_N(x_n) \qquad (4.3)$$

where $F_N(x_n)$ is the diaphony of $x_1, \ldots, x_N$.

It is known [53] that for the Zinterhof sequence the estimation of the diaphony

$$F_N(n\vec{\theta}) = \mathrm{O}(1/N^{1-\epsilon}), \qquad (4.4)$$

for $\epsilon > 0$ holds, since $\vec{\theta}$ is of the form $\theta_1 = e^{r_1}, \ldots, \theta_s = e^{r_s}$ where the $r_i \in \mathbb{Q} \, \forall i = 1, \ldots, s$ are rationally independent.

The definition of the diaphony F_N for a general s-dimensional sequence $\vec{x_1}, \ldots, \vec{x_N}$ is

$$F_N^2(\vec{x_n}) = \frac{1}{N} \sum_{i,j=1}^{N} H_2(\vec{x_i} - \vec{x_j}), \qquad (4.5)$$

with

$$H_2(\vec{x}) = \prod_{i=1}^{s} h_2(x_i) - 1,$$

and h_2 being the normed Bernoulli polynomial of degree 2,

$$h_2 = 1 + 2\pi^2 \left(\{x\}^2 - \{x\} + \frac{1}{6} \right),$$

where $\{x\}$ is the fractional part of x.

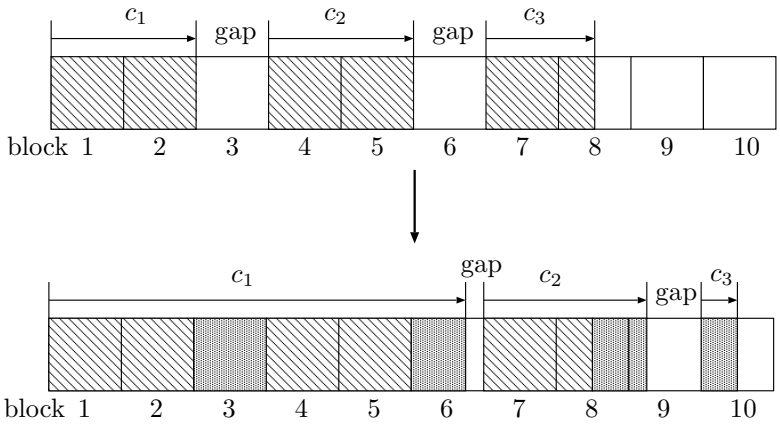


Figure 4.1: Growth of subsequences with small blocks.

The diaphony F_N of the sequence $x_1, \ldots, x_N$ is translation invariant, which follows directly from Equation (4.5) where we get $H_2((a + \vec{x_i}) - (a + \vec{x_j})) = H_2(\vec{x_i} - \vec{x_j})$, thus for any $a = (a_1, \ldots, a_s)$

$$F_N(x_n) = F_N(a + x_n)$$

holds.

For the Zinterhof sequence we can choose $a = x_B = (B\theta_1, \ldots, B\theta_s)$ such that we obtain

$$F_N(x_n) = F_N(x_B + x_n) = F_N(x_{B+n})$$

where $n = 1, \ldots, N$.

Thus when using the error approximation (4.3) we see that we can use an arbitrary block of length N instead of the first N points without deterioration of the integration error. Note that this corresponds well to an earlier result on (t,s)-sequences where it was shown that discrepancy estimates of arbitrary blocks do not degrade as compared to estimates of entire (t,s)-sequences [43].

Now, similar to [37], let us consider the general case of blocking with block size b where new blocks are handed out as requested ("small blocks"). The classical blocking scheme, which we call "big blocks", is essentially a subset of this general case. When using p PEs we have always p continuous subsets, each subset of points ends where a block is still unfinished. So we have p sequences each generating an approximation of the integral I

$$I'^i = \frac{1}{c_i} \sum_{\lambda(c_i)} f(x_i)$$

where $i = 1, \ldots, p$, c_i is the number of vectors in sequence i and $\lambda(c_1)$ is the set of indices of vectors of the original Zinterhof sequence which generates sequence i and the numbering be such that $c_p \leq c_{p-1} \leq \cdots \leq c_1$ holds. Figure 4.1 illustrates this for three PEs, when an PE finishes with block 3 the former c_1 and c_2 collapse to form the new c_1. Also when one block is finished another block is assigned and an PE starts to work on it, this forms a new sequence c_3.

Now with N the total number of points, we get

$$I'_N = \frac{1}{N} \sum_{\lambda(N)} f(x_i) = \sum_{i=1}^{p} \frac{c_i}{N} \frac{1}{c_i} \sum_{\lambda(c_i)} f(x_i) = \sum_{i=1}^{p} \frac{c_i}{N} I'^i,$$

which gives us the overall estimate from the estimates of the individual sequences.

We can consider the error

$$E_N(f) = |I'_N - I(f)| \leq \sum_{i=1}^{p} \frac{c_i}{N} |I'^i - I(f)| \leq \sum_{i=1}^{p} \frac{c_i}{N} D^*_{c_i} V(f).$$

When looking at blocking with a small block size, i.e. not the big block scenario, it is clear that the first sequence grows continuously as more intermediate blocks are finished and likewise new sequences are introduced at the end with a very small c_p. From the above error estimate we see that the weighted average of the discrepancies is used, but since c_1 continually grows for $N \to \infty$ we get $c_i/c_1 \to 0$ for $1 < i \leq p$. Since $c_p \leq c_i \leq c_2$ for $i = 3, \ldots, p-1$, we get

$$E_N(f) \leq V(f)(\frac{c_1}{N} D^*_{c_1} + \frac{(p-1)c_2}{N} D^*_{c_p}).$$

For very big N the error estimation thus becomes approximately

$$E_N(f) \leq V(f) D^*_{c_1}$$

where $c_1 \approx N$.

Clearly, the smaller the blocks are, the faster they become insignificant and the faster the first sequence grows. For big blocks we have a problem since unlike blocking with small blocks no sequence becomes insignificant and for the error we can only get the general error estimate. Given a homogenous environment where $c_1 = \cdots = c_p$ we get only

$$E_N(f) \leq V(f) D^*_{N/p}$$

which shows no advantage over using a single machine.

4.4 Parameterization

We already know the definition of the Zinterhof sequence, for a parameterization approach however we need P s-dimensional sequences which are independent. This can be achieved by splitting a Ps-dimensional Zinterhof sequence into P s-dimensional sequences by splitting the generator $\vec{\theta}$. Let $\vec{\theta} = (e^{1/1}, \ldots, e^{1/s}, \ldots, e^{1/(P-1)s+1}, \ldots, e^{1/Ps})$ be the generator of the Ps-dimensional sequence than we can obtain the P s-dimensional generators $\vec{\theta_1}, \ldots, \vec{\theta_P}$ as follows

$$\vec{\theta_1} = (e^{1/1}, \ldots, e^{1/s}),$$

$$\vdots$$

$$\vec{\theta_P} = (e^{1/(P-1)s+1}, \ldots, e^{1/Ps}).$$

What we now need is assurance that the $\vec{\theta_1}, \ldots, \vec{\theta_P}$ are independent and that the independence is enough to grant a higher convergence rate that just using a single s-dimensional sequence.

Concerning the convergence rate, Ökten, et.al. [37] has given the following. For P PEs the estimation of the integral is

$$I'_N(f) = \frac{1}{N}\sum_{n=1}^{N} f(x_n) = \sum_{i=1}^{P}\frac{c_i}{N}\frac{1}{c_i}\sum_{n=1}^{c_i} f(x_n^{(i)}) = \sum_{i=1}^{P}\frac{c_i}{N} I'^{(i)}(f),$$

where $N = c_1 + \cdots + c_P$ and the ith sequence contributes the points $x_1^{(i)}, \ldots, x_{c_i}^{(i)}$ to the overall sequence $x_i, \ldots, x_N$. Now if the sequences are independent $I'_N(f)$ becomes an unbiased estimator for the integral $I(f)$ with

$$\begin{aligned}\mathrm{Var}\left[I'_N(f) - I(f)\right] &= E\left|I'_N(f) - I(f)\right|^2 = \\ &= \sum_{i=1}^{P}\left(\frac{c_i}{N}\right)^2 E\left|I'^{(i)}(f) - I(f)\right|^2 \leq \\ &\leq (V(f))^2 \sum_{i=1}^{P}\left(\frac{c_i}{N}\right)^2 (D^*_{c_i})^2,\end{aligned}$$

which for equal speeds simplifies to

$$\mathrm{Var}\left[I'_N(f) - I(f)\right] = (D^*_{N/P})^2 (V(f))^2 / P.$$

The only thing left to do is to assure the independence of this parameterization approach.

It is well known [53] that the Weyl sequence

$$n(\theta_1, \ldots, \theta_s, \theta_{s+1}, \ldots, \theta_{2s}, \ldots, \theta_{(P-1)s+1}, \ldots, \theta_{Ps})$$

fulfills the sequential test $D_N = \mathrm{O}(1/N^{1-\epsilon})$ and a diaphony test of $F_N = \mathrm{O}(1/N^{1-\epsilon})$ for almost all choices $\vec{\theta} = (\theta_1, \ldots, \theta_{Ps})$ and especially for $\vec{\theta}$ with θ_i of the form $\theta_i = e^{r_i}$ where $r_i \in \mathbb{Q}$,$r_i \neq r_j \neq 0$,$1 \leq i, j \leq Ps$.

Independence of events is usually defined as follows: Let (Ω, α, P) be a probability space over the set Ω with the σ-algebra α of events with probability P. The events $E, F \in \alpha$ are independent if

$$P(E \cap F) = P(E) \cdot P(F).$$

Now let $E \sqsubseteq [0,1)^{2s}$,$F \sqsubseteq [0,1)^{2s}$ such that

$$E = \{(x_1, \ldots, x_{2s}) | \alpha_i \leq x_i < \beta_i,\ i = 1, \ldots, s,\ 0 \leq x_j < 1,\ j = s+1, \ldots, 2s\},$$

$$F = \{(x_1, \ldots, x_{2s}) | 0 \leq x_i < 1,\ i = 1, \ldots, s,\ \alpha_j \leq x_j < \beta_j,\ j = s+1, \ldots, 2s\}$$

and

$$E \cap F = \{(x_1, \ldots, x_{2s}) | \alpha_i \leq x_i < \beta_i,\ i = 1, \ldots, s,\ \alpha_j \leq x_j < \beta_j,\ j = s+1, \ldots, 2s\}.$$

Thus, for the Lebesgue measure $P = \lambda$ being the natural probability on $I^{2s} := [0,1)^{2s}$,

$$P(E \cap F) = P(E) \cdot P(F)$$

holds. So the events E and F are independent in the sense of the Lebesgue measure. On the other hand holds for almost all Weyl sequences $n\vec{\theta}$ and especially for those Weyl sequences with non zero different rational logarithms

$$D_N = \sup_J \left| \frac{A_N(J)}{N} - \lambda(I) \right| = \mathrm{O}(1/N^{1-\epsilon}).$$

So we get

$$P(E) = \frac{A_N(E)}{N} + \mathrm{O}(N^{\epsilon-1}),$$

$$P(F) = \frac{A_N(F)}{N} + \mathrm{O}(N^{\epsilon-1})$$

and

$$P(E \cap F) = \frac{A_N(E \cap F)}{N} + \mathrm{O}(N^{\epsilon-1}),$$

with generally different O-constants and $\epsilon > 0$.

Consequently, we get

$$\frac{A_N(E)}{N} \cdot \frac{A_N(F)}{N} = \frac{A_N(E \cap F)}{N} + \mathrm{O}(N^{\epsilon-1}),$$

which means that the arbitrary intervals $E, F \in I^s$ are independent up to an error $\mathrm{O}(N^{\epsilon-1})$ with relative frequencies $A_N(.)/N$.

It is well known that the independence properties of events are hard to prove, thus we will turn to another measure of independence, covariance and correlation. The covariance and correlation for two (pseudo-) random variables x, y are given as follows

$$\mathrm{cov}(x, y) = E(X - E(x)) \cdot (Y - E(y))$$

and

$$\mathrm{cor}(x, y) = \frac{\mathrm{cov}\, x, y}{(\mathrm{cov}\, x, x \cdot \mathrm{cov}\, y, y)^{1/2}}.$$

Now for two (pseudo-) random sequences $(x_n)_{n\geq 1}, (y_n)_{n\geq 1}$, $x_n, y_n \in I^s$ we consider the empirical correlation and covariance as

$$\mathrm{cov}((x_n)_{n=1}^N, (y_n)_{n=1}^N) := \frac{1}{N}\sum_{i=1}^N x_i \cdot y_i - \left(\frac{1}{N}\sum_{i=1}^N x_i\right) \cdot \left(\frac{1}{N}\sum_{i=1}^N y_i\right),$$

$$\mathrm{cor}((x_n)_{n=1}^N, (y_n)_{n=1}^N) := \frac{\mathrm{cov}((x_n)_{n=1}^N, (y_n)_{n=1}^N)}{[\mathrm{cov}((x_n)_{n=1}^N, (x_n)_{n=1}^N) \cdot \mathrm{cov}((y_n)_{n=1}^N, (y_n)_{n=1}^N)]^{1/2}},$$

where $x_n \cdot y_n$, denotes in a natural way the scalar product of the vectors x_n and y_n.

We considered only the pairwise independence of events E, F and of two Weyl sequences respectively. We give now some theoretical results concerning the covariance and correlation coefficient (and the correlation matrix respectively) of multidimensional sequences of nodes.

Let us start with a $2s$-dimensional sequence $x_k := (x_1^k, \ldots, x_s^k, x_{x+1}^k, \ldots, x_{2s}^k)$, $k = 1, \ldots, N$. We split this $2s$-dimensional sequence into two s-dimensional

sequences $x'_k := (x_1^k, \ldots, x_s^k)$ and $x''_k := (x_{x+1}^k, \ldots, x_{2s}^k)$, $k = 1, \ldots, N$. We are interested in the covariance and correlation behavior of these sequences. The mathematical expectation of a function $f(x) \mapsto y$, $I^{2s} \to \mathbb{R}, \mathbb{C}$ is

$$E(f) = \int_{I^{2s}} f(x)dx = \int_0^1 \cdots \int_0^1 f(x_1, \ldots, x_{2s})dx_1 \ldots dx_{2s}.$$

If f, g are independent functions then $E(f \cdot g) = E(f) \cdot E(g)$ holds and clearly $\operatorname{cov}(f, g) = E(f \cdot g) - (E(f) \cdot E(g)) = 0$.

Let

$$\sigma^2(f) = E(|f|^2) - |E(f)|^2,$$

$$\sigma^2(g) = E(|g|^2) - |E(g)|^2,$$

then for $\sigma^2(f) \neq \sigma^2(g) \neq 0$ the correlation coefficient of f and g can be given as

$$\operatorname{cor}(f, g) = \frac{\operatorname{cov}(f, g)}{\sigma(f) \cdot \sigma(g)}.$$

Given a finite collection of functions $f_1, \ldots, f_P$, the (hermitian) covariance and correlation matrices are defined by the entries $c_{ij} = \operatorname{cov}(f_i, f_j)$ and $r_{ij} = \operatorname{cor}(f_i, f_j)$ respectively.

Now we consider the random vector $(x_1, \ldots, x_s, x_{s+1}, \ldots, x_{2s})$ and split it up into $x' = (x_1, \ldots, x_s)$ and $x'' = (x_{s+1}, \ldots, x_{2s})$. So we get the important special case

$$\operatorname{cov} x', x'' = E(x_1 x_{s+1} + \cdots x_s x_{2s}) - E(x') \cdot E(x'').$$

We can easily calculate these values as

$$E(x_1 x_{s+1} + \cdots x_s x_{2s}) = \int_0^1 \cdots \int_0^1 (x_1 x_{s+1} + \cdots x_s x_{2s})dx_1 \ldots x_{2s} = \frac{s}{4},$$

$$E(x_1, \ldots, x_s) = E(x_{s+1}, \ldots, x_{2s}) = (\frac{1}{2}, \ldots, \frac{1}{2})$$

and consequently

$$\operatorname{cov}(x', x'') = \frac{s}{4} - (\frac{1}{2}, \ldots, \frac{1}{2}) \cdot (\frac{1}{2}, \ldots, \frac{1}{2}) = 0.$$

For the variances $\sigma^2(x')$ and $\sigma^2(x'')$ we get

$$\sigma^2(x') = E(x_1^2 + \cdots + x_s^2) - E(x_1, \ldots, x_s)^2 = \frac{s}{3} - \frac{s}{4} = \frac{s}{12},$$

and likewise $\sigma^2(x'') = s/12$. So the random points x' and x'' are non constant uncorrelated random variables because

$$\operatorname{cor}(x', x'') = \frac{\operatorname{cov}(x', x'')}{\sigma(x') \cdot \sigma(x'')} = 0.$$

We consider now the covariance and the correlation of sequences $(x_1^k, \ldots, x_s^k, x_{s+1}^k, \ldots, x_{2s}^k)$, $k = 1, \ldots, N$ where $x' = (x_1^k, \ldots, x_s^k) \in I^s$ and $x'' =$

$(x_{s+1}^k, \ldots, x_{2s}^k) \in I^s$. When we replace the expectation E by the mean of the N values we get

$$\begin{aligned} \operatorname{cov}(x', x'') &= \frac{1}{N}\sum_{k=1}^{N}(x_1^k x_{x+1}^k + \cdots + x_s^k x_{2s}^k) - \\ &- \left(\frac{1}{N}\sum_{k=1}^{N}(x_1^k, \ldots, x_s^k)\right) \cdot \left(\frac{1}{N}\sum_{k=1}^{N}(x_{s+1}^k, \ldots, x_{2s}^k)\right). \end{aligned}$$

We continue by applying number theoretical methods to this problem. It is clear that the covariance contains sums of the types

$$\frac{1}{N}\sum_{k=1}^{N} x_i^k x_{s+i}^k$$

and

$$\frac{1}{N}\sum_{k=1}^{N} x_i^k.$$

The variation in the sense of Hardy and Krause of the functions xy and x itself is easily computed as $V_2(x \cdot y) = 1$, $V_1(x) = 1$. The expectations $E(xy)$ and $E(x)$ are easily computed as

$$E(xy) = \int_0^1 \int_0^1 xy\, dxdy = \frac{1}{4}$$

and

$$E(x) = \frac{1}{2}.$$

By the Koksma-Hlawka inequality we get

$$\frac{1}{N}\sum_{k=1}^{N} x_i^k x_{s+i}^k = \frac{1}{4} + R_N(i,2),\ i = 1, \ldots, s$$

and

$$\frac{1}{N}\sum_{k=1}^{N} x_i^k = \frac{1}{4} + R_N(i,1),\ i = 1, \ldots, s$$

where for the errors $R_N(i,1)$ and $R_N(i,2)$ the approximation

$$|R_N(i,1)| = \mathrm{O}(D_N)$$

and

$$|R_N(i,2)| = \mathrm{O}(D_N)$$

holds.

Consequently, for the covariance we get

$$\begin{aligned}\operatorname{cov}(x', x'') &= \frac{s}{4} + \sum_{i=1}^{s} R_N(i,2) - \left[\left(\frac{1}{2}, \dots \frac{1}{2}\right) + (R_N(1,1), \dots, R_N(s,1))\right] \cdot \\ &\quad \cdot \left[\left(\frac{1}{2}, \dots \frac{1}{2}\right) + (R_N(s+1,1), \dots, R_N(2s,1))\right] = \\ &= \frac{s}{4} + \sum_{i=1}^{s} R_N(i,2) - \frac{s}{4} + \frac{1}{2}\left(\sum_{i=1}^{s}(R_N(i,1) + R_N(s+i,1))\right) + \\ &\quad + \sum_{i=1}^{s} R_N(i,1) \cdot R_N(s+i,1).\end{aligned}$$

And finally we get the estimation

$$|\operatorname{cov}(x', x'')| = \mathrm{O}(D_N + D_N^2),$$

where the O-constant is depending only on the dimension s of the sequences x', x'', $k = 1, \dots, N$. So in essence the covariance of the two sequences only depends on the distribution of the sequences and on the dimension. Consequently, the better distributed the $2s$-dimensional sequence is the less correlated the two resulting s-dimensional sequences will be.

Since we have now an estimation of the covariance we can proceed to the correlation

$$\operatorname{cor}(x', x'') = \frac{\operatorname{cov}(x', x'')}{\sigma(x') \cdot \sigma(x'')},$$

it remains only to compute the variances σ.

We will only look at $\sigma^2(x')$, since the same results will hold for $\sigma^2(x'')$,

$$\sigma^2(x') = \frac{1}{N}\sum_{k=1}^{N}((x_1^k)^2 + \dots + (x_s^k)^2) - \left(\frac{1}{N}\sum_{k=1}^{N}(x_i^k, \dots, x_s^k)\right)^2.$$

Because of $E(x^2) = 1/3$ and $E(x) = 1/2$ we get

$$\frac{1}{N}\sum_{k=1}^{N}(x_i^k)^2 = \frac{1}{3} + S(i),\ i = 1, \dots, s$$

and

$$\frac{1}{N}\sum_{k=1}^{N} x_i^k = \frac{1}{2} + T(i),\ i = 1, \dots, s.$$

The Koksma-Hlawka inequality leads to an estimation of the remainder terms $|S(i)| \leq \mathrm{O}(D_N)$ and $|T(i)| \leq \mathrm{O}(D_N)$ and thus to the result

$$\sigma^2(x') = \frac{s}{12}\sum_{i=1}^{s}(S(i) - T(i) - T(i)^2) = \frac{s}{12} + \mathrm{O}(D_N),$$

where for both $\sigma^2(x')$ and $\sigma^2(x'')$ the O-constant only depends on the dimension s. Combining these results we get

$$\operatorname{cor}(x', x'') = \mathrm{O}(D_N),$$

where the O-constant depends only on the dimension s.

Skipping the other calculations for pairwise comparisons for $P > 2$ (since they consist only of an almost word for word repetition of the previous arguments) we can state the following.

Let $f_1 = (\{n\theta_1\}, \ldots, \{n\theta_s\})$, ..., $f_P = (\{n\theta_{(P-1)s+1}\}, \ldots, \{n\theta_{Ps}\})$, $n = 1, \ldots, N$ and let I_P be the $P \times P$ unit matrix with entries $e_{ii} = 1$ and $e_{jk} = 0$ for $j \neq k$ and $i, j, k = 1, \ldots, P$. Let D_N^{Ps} denote the discrepancy for the Ps-dimensional sequence $(\{n\theta_1\}, \ldots, \{n\theta_s\}, \ldots, \{n\theta_{(P-1)s+1}\}, \ldots, \{n\theta_{Ps}\}), n = 1, \ldots, N$. Then the covariance and correlation matrix for the sequences $f_1, \ldots, f_P$ can be given as

$$\mathrm{cov}(f_1, \ldots, f_P) = \frac{s}{12} \cdot I_P + \mathrm{O}(D_N^{Ps})$$

$$\mathrm{cor}(f_1, \ldots, f_P) = I_P + \mathrm{O}(D_N^{Ps}).$$

And for almost all Weyl sequences we can write

$$\mathrm{cov}(f_1, \ldots, f_P) = \frac{s}{12} \cdot I_P + \mathrm{O}(N^{\epsilon-1}),$$

and

$$\mathrm{cor}(f_1, \ldots, f_P) = I_P + \mathrm{O}(N^{\epsilon-1}),$$

where the estimation holds especially for generators of the type $\vec{\theta} = (e^{r_1}, \ldots, e^{r_{Ps}})$ where $r_i \in \mathbb{Q}$ and $r_j \neq r_k$ for $j \leq k$ with $i, j, k = 1, \ldots, Ps$.

We can clearly see that the Zinterhof sequences fulfill the later requirements and thus the estimation holds.

4.5 Conclusion

Overall, we have shown that Zinterhof sequences are well suited for numerical integration in GRID environments. Whereas the error estimation for leaped substreams suggests worse integration errors as compared to sequential usage, experimental results do not indicate this.

For the case of using contiguous blocks for integration the theoretical prediction suggesting behavior equal to the sequential case is supported by experimental results. Also note that the results concerning blocking was not specific for the Zinterhof sequence, a behavior close to the original sequence can be expected of all sequences when blocking with small block sizes is used, under the assumption that we can compute the discrepancy of an arbitrary block.

While the suggested parametrization scheme works in principle, we have already seen that in application it seems to do not as well as blocking and leaping, especially when it comes to utilizing a high number of PEs.

In summary we have shown that the Zinterhof sequence can be used for parameterization but care has to be taken. For blocking with small block sizes all sequences, in the above sense, can be used and results which parallel a sequential execution can be expected. For leaping we have shown a difference in the error which is not reflected in the experiments. While experiments can only be used to ascertain that an error is indeed minimal but never (except when all possibilities are taken into account) that the given error rate is to high, it is a strong suggestion that we have not obtained the minimal error.

Bibliography

[1] V. Alexandrov, E. Atanassov, and I. Dimov, *Parallel quasi Monte Carlo methods for linear algebra problems*, Monte Carlo Methods and Applications, 10 (2004), pp. 213–219.

[2] B. Bromley, *Quasirandom number generators for parallel Monte Carlo algorithms*, Journal of Parallel and Distributed Computing, **38** (1996), pp. 101–104.

[3] A. T. Clayman, K. M. Lawrence, G. L. Mullen, H. Niederreiter, and N. J. A. Sloane, *Updated tables of parameters of* (t, m, s)*-nets*, Journal of Combinatorial Designs, 7 (1999), pp. 381–393.

[4] L. Cucos and E. deDoncker, *Distributed QMC algorithms: new strategies for and performance evaluation*, in Proceedings of the High Performance Computing Symposium 2002 (HPC'02)/Advanced Simulation Techniques Conference, 2002, pp. 155–159.

[5] E. de Doncker and Y. Guan, *Error bounds for integration of singular functions using equidistributed sequences*, Journal of Complexity, 19 (2003), pp. 259–271.

[6] E. deDoncker, A. Genz, and M. Ciobanu, *Parallel compuation of multivariate normal probabilities*, Computing Science and Statistics, 30 (1999).

[7] E. deDoncker, R. Zanny, M. Ciobanu, and Y. Guan, *Asynchronous quasi-Monte Carlo methods*, in Proceedings of the High Performance Computing Symposium 2000 (HPC'00), 2000, pp. 130–135.

[8] ——, *Distributed quasi-Monte Carlo methods in a heterogeneous environment*, in Proceedings of the Heterogeneous Computing Workshop 2000 (HCW'2000), IEEE Computer Society Press, 2000, pp. 200–206.

[9] E. deDoncker, R. Zanny, and K. Kaugars, *Distributed numerical integration algorithms and applications*, in Proceedings of the 4th World Multiconference on Systemics, Cybernetics, and Informatics (SCI'00), 2000, pp. 244–249.

[10] M. Drmota and R. Tichy, *Sequences, discrepancies and applications*, vol. 1651 of Lect. Notes in Math., Springer-Verlag, 1997.

[11] K. Entacher, P. Hellekalek, and P. L'Ecuyer, *Quasi-Monte Carlo node sets from linear congruential generators*, in Monte Carlo and Quasi-Monte Carlo Methods 1998, H. Niederreiter and J. Spanier, eds., Springer, 2000, pp. 188–198.

[12] K. Entacher, T. Schell, W. C. Schmid, and A. Uhl, *Defects in parallel Monte Carlo and quasi-Monte Carlo integration using the leap-frog technique*, Parallel Algorithms and Applications, 18 (2003), pp. 27–47.

[13] G. Evans, *Practical numerical integration*, Wiley, Chichester, 1993.

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

[15] J. H. Halton, *On the efficiency of certain quasi-random sequences of points in evaluating multi-dimension integrals*, Numer. Math., 2 (1960), pp. 84–90. Berichtigung, ibid., (1960), p. 196.

[16] J. Hartinger, R. Kainhofer, and V. Ziegler, *On the corner avoidance properties of various low-discrepancy sequences*, (2005). in press.

[17] J. Hartinger and R. F. Kainhofer, *Non-uniform low-discrepancy sequence generation and integration of singular integrands*, in Proceedings of MC2QMC2004, Juan-Les-Pins France, June 2004, H. Niederreiter and D. Talay, eds., Springer Verlag, June 2005.

[18] J. Hartinger, R. F. Kainhofer, and R. F. Tichy, *Quasi-monte carlo algorithms for unbound, weighted integration problems*, Journal of Complexity, 20 (2004), pp. 654–668.

[19] E. Hlawka and R. Mück, *Über eine transformation von gleichverteilten folgen*, Computing, 9 (1972), pp. 127–138.

[20] H. Hofbauer, A. Uhl, and P. Zinterhof, *Quasi Monte Carlo Integration in GRID Environments: Further Leaping Effects*, Parallel Processing Letters, 16 (2006), pp. 285–311.

[21] ———, *Quasi-Monte Carlo Integration on GRIDS: Using blocked Substreams*, in Proceedings of the 1st Austrian Grid Symposium, J. Volkert, T. Fahringer, D. Kranzlmüller, and W. Schreiner, eds., vol. 210 of books@ocg.at, Schloss Hagenberg, Austria, 2006, Austrian Computer Society, pp. 219–233.

[22] ———, *Parameterization of Zinterhof Sequences for GRID-based QMC Integration*, in Proceedings of the 2nd Austrian Grid Symposium, J. Volkert, T. Fahringer, D. Kranzlmüller, and W. Schreiner, eds., books@ocg.at, Innsbruck, Austria, 2007, Austrian Computer Society. To appear.

[23] H. L. Keng and W. Yuan, *Applications of Number Theory to Numerical Analysis*, Springer Verlag, Science Press, 1981.

[24] B. Klinger, *Numerical Integration of Singular Integrands Using Low-Discrepancy Sequences*, Computing, 59 (1997), pp. 223–236.

[25] L. Kocis and W. Whiten, *Computational investigations of low-discrepancy sequences*, ACM Transactions on Mathematical Software, 23 (1997), pp. 266–294.

[26] A. Krommer and C. Überhuber, *Numerical Integration on Advanced Computer Systems*, vol. 848 of Lecture Notes in Computer Science, Springer, Berlin, 1994.

[27] ——, *Computational integration*, SIAM, Philadelphia, 1998.

[28] J. Li and G. Mullen, *Parallel computing of a quasi-Monte Carlo algorithm for valuing derivatives*, Parallel Computing, 26 (2000), pp. 641–653.

[29] S. Li, K. Kaugars, and E. deDoncker, *Grid-based numerical integration and visualization*, in Sixth International Conference on Computational Intelligence and Multimedia Applications (ICCIMA'05), IEEE Computer Society Press, 2005, pp. 260–265.

[30] Y. Li and M. Mascagni, *Grid-based Quasi-Monte Carlo applications*, Monte Carlo Methods and Appl., 11 (2005), pp. 39–55.

[31] M. Mascagni and A. Karaivanova, *A parallel quasi-Monte Carlo method for computing extremal eigenvalues*, in Monte Carlo and Quasi-Monte Carlo Methods 2000, K. T. Fang, F. J. Hickernell, and H. Niederreiter, eds., Springer-Verlag, 2002, pp. 369–380.

[32] ——, *A parallel quasi-Monte Carlo method for solving systems of linear equations*, in The 2002 International Conference on Computational Science - ICCS 2002, P. Sloot et al., eds., vol. 2330 of Lecture Notes in Computer Science, Springer Verlag, Berlin, Germany, May 2002, pp. 598–608.

[33] G. L. Mullen, A. Mahalanabis, and H. Niederreiter, *Tables of (t, m, s)-net and (t, s)-sequence parameters*, in Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing, H. Niederreiter and P. J.-S. Shiue, eds., vol. Lecture Notes in Statistics of 106, Springer-Verlag, 1995, pp. 58–86.

[34] H. Niederreiter, *Point sets and sequences with small discrepancy*, Monatshefte für Mathematik, 104 (1987), pp. 273–337.

[35] H. Niederreiter, *Random Number Generation and Quasi-Monte Carlo Methods*, no. 63 in CBMS-NSF Series in Applied Mathematics, SIAM, Philadelphia, 1992.

[36] H. Niederreiter and C. Xing, *Low-discrepancy sequences and global function fields with many rational places*, Finite Fields and Their Applications, 2 (1996), pp. 241–273.

[37] G. Ökten and A. Srivivasan, *Parallel quasi-Monte Carlo methods on a heterogeneous cluster*, in Monte Carlo and Quasi-Monte Carlo Methods 2000, K. T. Fang, F. J. Hickernell, and H. Niederreiter, eds., Springer-Verlag, 2002, pp. 406–421.

[38] A. B. Owen, *Halton Sequences Avoid the Origin*, tech. rep., Stanford University, 2004.

[39] ——, *Multidimensional variations for quasi-Monte Carlo*, tech. rep., Stanford University, January 2004.

[40] ——, *Quasi-Monte Carlo for integrands with point singularities at unknown locations*, tech. rep., Stanford University, 2004.

[41] I. Radović, I. M. Sobol, and R. F. Tichy, *Quasi-monte carlo methods for numerical integration: Comparison of different low discrepancy sequences*, Monte Carlo Methods and Appl., **2** (1996), pp. 1–14.

[42] W. C. Schmid and A. Uhl, *Parallel quasi-Monte Carlo integration using (t,s)-sequences*, in Parallel Computation. Proceedings of ACPC'99, P. Zinterhof, M. Vajtersic, and A. Uhl, eds., vol. 1557 of Lecture Notes on Computer Science, Springer-Verlag, 1999, pp. 96–106.

[43] ——, *Techniques for parallel quasi-Monte Carlo integration with digital sequences and associated problems*, Mathematics and Computers in Simulation, 55 (2001), pp. 249–257.

[44] R. Schürer, *Parallel high-dimensional integration: quasi-Monte Carlo versus adaptive cubature rules*, in The 2001 International Conference on Computational Science - ICCS 2001, V. N. Alexandrov, J. J. Dongarra, B. A. Juliano, R. S. Renner, and C. J. K. Tan, eds., vol. 2073 of Lecture Notes in Computer Science, San Francisco, CA, USA, May 2001, Springer Verlag, Berlin, Germany, pp. 1262–1271.

[45] R. Schürer and W. C. Schmid, *Mint - the database of optimal (t, m, s)-net parameters*. online: `http://mint.sbg.ac.at/`.

[46] R. Schürer and A. Uhl, *An evaluation of adaptive numerical integration algorithms on parallel systems*, Parallel Algorithms and Applications, 18 (2003), pp. 13–26.

[47] I. Sobol', *On the distribution of points in a cube and the approximate evaluation of integrals*, U.S.S.R. Computational Mathematics and Mathematical Physics, 7 (1967), pp. 86–112.

[48] ——, *Calculation of improper integrals using uniformly distributed sequences*, Soviet Math Dokl., 14 (1973), pp. 734–738.

[49] A. Srinivasan, *Parallel and distributed computing issues in pricing financial derivatives through quasi-Monte Carlo*, in Proceedings of the International Parallel & Distributed Processing Symposium 2002 (IPDPS'02), Fort Lauderdale, FL, USA, Apr. 2002, IEEE Computer Society Press.

[50] J. Wan, K. Lai, A. Kolkiewicz, and K. Tan, *A parallel quasi Monte Carlo approach to pricing multidimensional americal options*, International Journal of High Performance Computing and Networking, (2006). To appear.

[51] C. Xing and H. Niederreiter, *A construction of low-discrepancy sequences using global function fields*, Acta Arithmetica, 71 (1995), pp. 87–102.

[52] P. Zinterhof, *Einige zahlentheoretische methoden zur numerischen Quadratur und Interpolation*, Sitzungsberichte der Österreichischen Akademie der Wissenschaften, math.-nat.wiss. Klasse Abt. II, 177 (1969), pp. 51–77.

[53] ——, *Über einige Abschätzungen bei der Approximation von Funktionen mit Gleichverteilungsmethoden*, Sitzungsber. Österr. Akad. Wiss. Math.-Natur. Kl. II, **185** (1976), pp. 121–132.

[54] ——, *High dimensional improper integration procedures*, in Proceedings of Contributions of the 7th International Scientific Conference, C. E. F. Technical University of Košice, ed., Hroncova 5, 04001 Košice, SLOVAKIA, May 2002, TULIP, pp. 109–115.