Interner Bericht

Computing Discrepancies of Smolyak Quadrature Rules

Karin Frank, Stefan Heinrich

284/96

Fachbereich Informatik

Universität Kaiserslautern · Postfach 3049 · D-67653 Kaiserslautern

Computing Discrepancies of Smolyak Quadrature Rules

Karin Frank, Stefan Heinrich

284/96

Universität Kaiserslautern
AG Numerische Algorithmen
Postfach 30 49
67653 Kaiserslautern
Germany

Mai 1996

Herausgeber: AG Numerische Algorithmen
Leiter: Professor Dr. S. Heinrich

Computing Discrepancies of Smolyak Quadrature Rules

Karin Frank, Stefan Heinrich

Fachbereich Informatik
Universität Kaiserslautern
Postfach 3049, D–67653 Kaiserslautern

24th May 1996

Abstract

In recent years, Smolyak quadrature rules (also called hyperbolic cross points or sparse grids) have gained interest as a possible competitor to number theoretic quadratures for high dimensional problems. A standard way of comparing the quality of multivariate quadrature formulas consists in computing their L_2-discrepancy. Especially for larger dimensions, such computations are a highly complex task. In this paper we develop a fast recursive algorithm for computing the L_2-discrepancy (and related quality measures) of general Smolyak quadratures. We carry out numerical comparisons between the discrepancies of certain Smolyak rules, Hammersley and Monte Carlo sequences.

1 Introduction

Smolyak (1963) developed a special tensor product technique of constructing higher dimensional quadrature formulas and approximation operators from the corresponding one-dimensional objects. Although technically comparatively simple, it leads to almost (up to logarithmic factors) optimal error rates, as soon as the one-dimensional methods involved possess such properties.

By now this technique is known under many different names, as e. g. "Boolean method" (Delvos (1982)), "discrete blending method" (Baszenski et al. (1992)), "hyperbolic cross points" (Babenko (1960)) or "sparse grids" (Zenger (1991)).

In this paper we are concerned with Smolyak quadrature rules. Their optimality properties in certain function spaces were studied e. g. in Smolyak (1963), Temlyakov (1990, 1993), Wasilkowski and Woźniakowski (1995). Numerical experiments for certain families of test functions (see e. g. Griebel et al. (1992), Baszenski and Delvos (1993), Zielinski (1994), Bonk (1994), Novak and Ritter (1996)) demonstrated not only the superiority of Smolyak quadrature rules to ordinary tensor product rules, but also comparable performance with respect to quasi Monte Carlo integration, especially when applied to rather smooth functions in high dimensions ($d \geq 6$).

A quantitative measure of the precision of multivariate quadrature formulas is the L_2-discrepancy. It can be computed explicitly, although at high cost for large dimensions ($O(N^2)$, see Warnock (1972), $O(N(\log N)^d)$, see Heinrich (1995), with N the number of nodes and d the dimension). The L_2-discrepancy is often used to compare the quality of multivariate quadratures such as quasi Monte Carlo methods (low discrepancy sequences) and pseudo Monte Carlo methods (sequences

produced by random number generators). It would therefore be of interest to compute the L_2-discrepancy of Smolyak quadratures. A straight-forward application of the algorithms above would be very costly: In addition to the general complexity of computing the L_2-discrepancy the Smolyak quadratures are defined and computed recursively, so their weights are not given explicitly. Hence one first would have to compute (accumulate) the weights by the help of a special computer program, and then use the general formulas! (As far as we know, nobody has done this till now.) On the other hand, the only essential ingredient of a multidimensional Smolyak rule is one single or a sequence of one-dimensional quadratures. So the question arises if this special structure could lead to significant improvements in computing the discrepancy. This is the topic of the present paper: On the basis of general tensor product properties of the discrepancy we develop a recursive algorithm for computing the discrepancy of general d-dimensional Smolyak quadrature rules. Under some natural assumptions on the number of nodes in the one-dimensional building blocks of the Smolyak quadrature the complexity of our algorithm is

$$O(P \log P + d(\log P)^4)$$

if the one-dimensional quadratures are arbitrary, and

$$O(d(\log P)^4)$$

if the one-dimensional rules are composites of a basis quadrature with $O(1)$ nodes. Here P denotes the largest number of nodes in the one-dimensional rules, and the constants involved in the O-notation do not depend on P and d. (For comparison: The number of nodes N in the full Smolyak quadrature is $N = O(P(\log P)^{d-1})$.)

This allows us to compare discrepancies of Smolyak quadratures with other multivariate methods. But more than this, it enables us to reach exotic numbers of quadrature points (10^{35}). Therefore our algorithm might also be of interest for further theoretical investigations of Smolyak rules. For such numbers of nodes we compare their discrepancy with the (easily exactly computed) average discrepancy of random points.

The paper is organized as follows. We start with an introduction to Smolyak's technique in section 2. The next section discusses the L_2-discrepancy. We have chosen a fairly general approach (relating L_2-discrepancies to reproducing kernels of Hilbert spaces) in view of section 4, where the main tensor product properties of the discrepancies are proved namely in that generality. These results lead to an algorithm which recursively reduces the computation to the one-dimensional case. At this point we specify our considerations to the classical L_2-discrepancy and to the r-smooth counterparts introduced in Paskov (1993). Since the discrepancy of one-dimensional quadratures remains essentially the only costly part of the algorithm, we study this case in section 5. Here the main result is an efficient algorithm for composite quadrature rules. Numerical experiments are contained in section 6.

2 Smolyak quadrature rules

Let $\{Q_n\}_{n=0}^{\infty}$ be a sequence of quadrature rules on $[0,1]$ for continuous functions $f \in C([0,1])$

$$Q_n f \;=\; \sum_{i=0}^{\nu_n} w_i^n \cdot f(x_i^n)\;,$$

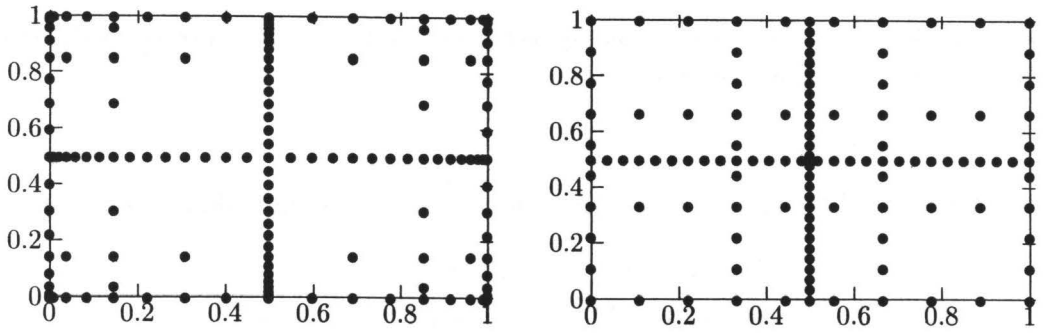

Figure 1: The underlying grid of the Smolyak quadrature based on the Clenshaw–Curtis rule (left, $p = 2$, $n = 4$) and on the trapezoidal rule (right, $p = 3$, $n = 3$).

with $w_i^n \in \mathbb{R}$, $x_i^n \in [0,1]$. Then the standard tensor product quadrature rule for the approximate computation of

$$If = \int_G f(x)\,dx\,,$$

where $G = [0,1]^d$, is

$$U_n^{(d)} f = \left(Q_n \otimes U_n^{(d-1)}\right) f\,,$$

where $U_n^{(1)} = Q_n$, and $Q_n \otimes U_n^{(d-1)}$ is the tensor product of the quadrature formulas, in other words, we apply Q_n with respect to the first variable of f and $U_n^{(d-1)}$ with respect to the remaining $d-1$ variables (see section 4 for a formal definition of the tensor product). Clearly

$$U_n^{(d)} = \sum_{i=0}^{n} (Q_i - Q_{i-1}) \otimes U_n^{(d-1)}\,, \tag{1}$$

with $Q_{-1} \equiv 0$. Now Smolyak's approach consists in the following modification: Set $Q_n^{(1)} = Q_n$ and define $Q_n^{(d)}$ recursively by

$$Q_n^{(d)} = \sum_{i=0}^{n} (Q_i - Q_{i-1}) \otimes Q_{n-i}^{(d-1)}\,, \tag{2}$$

where $Q_{-1} \equiv 0$ again (see Smolyak (1963)).

Often the sequence $\{Q_n\}$ is constructed in the following way. Let Q be an arbitrary quadrature rule on $[0,1]$ and $p \geq 2$ be a natural number. Then Q_n for positive $n \in \mathbb{N}$ denotes the composite quadrature rule of applying Q on p^n subintervals $[\frac{i}{p^n}, \frac{i+1}{p^n}]$, $i = 0, \ldots, p^n - 1$. The number n is called the level of the composite formula Q_n. For multivariate quadrature it is useful to choose Q_0 as the midpoint rule

$$Q_0 f = f(0.5)\,,$$

since otherwise the number of points used by $Q_n^{(d)}$ increases too fast with d.

The grid Γ_n^d employed by (2) is a so-called sparse grid (see fig. 1), in contrast to the regular grid used by (1). If the grids Γ_i^1 are nested, i. e.

$$\Gamma_0^1 \subset \Gamma_1^1 \subset \ldots \subset \Gamma_n^1 ,$$

then so are the grids Γ_i^d for all $d > 1$, and the sparse grid Γ_n^d has the following structure

$$\begin{aligned} \Gamma_n^d &= (\Gamma_0^1 \times \Gamma_n^{d-1}) + (\Gamma_1^1 \times \Gamma_{n-1}^{d-1}) + \ldots + (\Gamma_n^1 \times \Gamma_0^{d-1}) \\ &= (\Gamma_0^1 \times \Gamma_n^{d-1}) + ((\Gamma_1^1 \setminus \Gamma_0^1) \times \Gamma_{n-1}^{d-1}) + \ldots + ((\Gamma_n^1 \setminus \Gamma_{n-1}^1) \times \Gamma_0^{d-1}) . \end{aligned}$$

Since the subsets $(\Gamma_i^1 \setminus \Gamma_{i-1}^1) \times \Gamma_{n-i}^{d-1}$, $i = 0, ..., n$, do not have common points, the cardinality of Γ_n^d can be calculated recursively by

$$|\Gamma_n^d| = \sum_{i=0}^{n} |\Gamma_i^1 \setminus \Gamma_{i-1}^1| \cdot |\Gamma_{n-i}^{d-1}| , \tag{3}$$

where $\Gamma_{-1}^1 = \emptyset$. This formula will be used to calculate the cardinality of the grids in section 6. If the underlying sequence $\{Q_n\}$ consists of composite quadratures as described above, it gives cardinalities $|\Gamma_n^d| = O(n^{d-1}p^n)$ (in contrast to the cardinality of the regular tensor grid $O(p^{nd})$). Relation (3) was used earlier by Paskov (1993) to compute the number of points in a grid arising from Smolyak's quadrature rule using the rectangular rule as the basis quadrature Q.

3 A general notion of discrepancy

Let $G = [0,1]^d$, let $C(G)$ be the space of continuous functions on G and $L_2(G)$ the space of square integrable with respect to the Lebesgue measure functions on G. By $C(G, L_2(G))$, or shortly $C(L_2)$ we shall denote the space of all Borel measurable functions $B(x,t)$ on G^2 with $B(x,\cdot) \in L_2(G)$ for all $x \in G$ and such that $x \to B(x,\cdot)$ defines a continuous mapping from G into $L_2(G)$. The space $C(L_2)$ is endowed with the norm

$$\|B\|_{C(L_2)} = \max_{x \in G} \|B(x,\cdot)\|_{L_2(G)} .$$

For each $B \in C(L_2)$, the integral operator

$$(T_B f)(x) = \int_G B(x,t)\, f(t)\, dt , \quad f \in L_2(G)$$

defines a compact linear operator $T_B : L_2(G) \to C(G)$. The function B is the starting point of our general definition of discrepancy.

Let $Q = ((x_1, v_1), \ldots, (x_M, v_M))$ be an array defining a quadrature formula

$$Qf = \sum_{i=1}^{M} v_i\, f(x_i)$$

for any $f \in C(G)$. Given B as above and $t \in G$, we set

$$e(t) = \int_G B(x,t)\, dx - \sum_{i=1}^{M} v_i\, B(x_i, t) . \tag{4}$$

Observe that $e(t)$ is defined for almost all $t \in G$ and belongs to $L_2(G)$. The L_2-discrepancy with respect to the function B is the mean square error of integrating $B(\cdot, t)$ by the help of Q:

$$D_B(Q) \;=\; \Big(\int_G e(t)^2 dt\Big)^{1/2} . \tag{5}$$

Clearly, for $B(x,t) = \chi_{[0,t)}(x)$ with $\chi_{[0,t)}(x) = \chi_{[0,\tau_1)}(\xi_1) \cdot \ldots \cdot \chi_{[0,\tau_d)}(\xi_d)$ and $x = (\xi_1, \ldots, \xi_d)$, $t = (\tau_1, \ldots, \tau_d)$ we obtain the usual L_2-discrepancy (see Niederreiter (1978, 1992)), while for $B(x,t) = \frac{1}{(r!)^d}(t-x)_+^r$ with $(t-x)_+^r = (\tau_1 - \xi_1)_+^r \cdot \ldots \cdot (\tau_d - \xi_d)_+^r$ and $a_+ = a$ if $a > 0$ and $a_+ = 0$ otherwise, we get the r-smooth L_2-discrepancy introduced in Paskov (1993). Both will be studied in the sequel, but first we return to the case of general $B \in C(L_2)$ and derive some expressions for $D_B(Q)$ which will be used later on. Put

$$(IB)(t) \;=\; \int_G B(x,t)\, dx$$

and

$$(QB)(t) \;=\; \sum_{i=1}^{M} v_i\, B(x_i, t) .$$

Then IB and QB belong to $L_2(G)$, and

$$D_B(Q)^2 \;=\; (IB, IB) - 2(IB, QB) + (QB, QB) . \tag{6}$$

Set furthermore

$$K(x,y) \;=\; \int_G B(x,t) B(y,t)\, dt\,, \quad (x, y \in G) . \tag{7}$$

It follows from (4) and (5) that

$$D_B(Q)^2 \;=\; \int_{G^2} K(x,y)\, dx\, dy - 2\sum_{i=1}^{M} v_i \int_G K(x, x_i)\, dx + \sum_{i,j=1}^{M} v_i v_j K(x_i, x_j) . \tag{8}$$

Hence $D_B(Q)$ depends on B only through K. Relation (8) generalizes Warnock's formula (Warnock (1972)), which we recover for $B(x,t) = \chi_{[0,t)}(x)$ and

$$K(x,y) \;=\; \int_G \chi_{[0,t)}(x) \chi_{[0,t)}(y)\, dt \;=\; \prod_{i=1}^{d} (1 - \max(\xi_i, \eta_i)) , \tag{9}$$

with $x = (\xi_1, \ldots, \xi_d)$ and $y = (\eta_1, \ldots, \eta_d)$. To motivate our general definition, we show, that it can be interpreted as the worst case error of Q over some Hilbert space of functions. Let

$$H(K) \;=\; \{h \in C(G) \,:\, h = T_B f \text{ for some } f \in L_2(G)\}$$

be endowed with the norm

$$\|h\|_{H(K)} \;=\; \inf\{\|f\|_{L_2(G)} \,:\, h = T_B f\} .$$

Using the Schmidt representation of T_B (considered as an operator in $L_2(G)$),

$$T_B = \sum_i \lambda_i^{1/2} g_i \otimes h_i$$

with λ_i being the non-zero eigenvalues (repeated according to their multiplicity) and (g_i) and (h_i) the corresponding orthonormal systems of eigenfunctions of $T_B^* T_B$ and $T_B T_B^* = T_K$, respectively, we obtain for $h \in H(K)$,

$$\|h\|_{H(K)}^2 = \sum_i \lambda_i^{-1} (h, h_i)^2 ,$$

where $(\cdot, \cdot)$ stands for the scalar product of $L_2(G)$ and the sums over i are finite or infinite, depending on whether T_B has a finite or infinite number of non-zero eigenvalues. Hence $H(K)$ is a Hilbert space. In fact, it is the reproducing kernel Hilbert space generated by K, see Aronszajn (1950) (it follows from (7) that K is continuous, symmetric and non-negative definite). In the case of $B(x,t) = \frac{1}{(r!)^d}(t-x)_+^r$ (as well as in many other cases, see Ritter et al. (1995)), $H(K)$ is a Sobolev space of functions $f \in L_2(G)$ satisfying

$$\left\| \frac{\partial^{(r+1)d} f(\xi_1, \ldots, \xi_d)}{\partial \xi_1^{r+1} \ldots \partial \xi_d^{r+1}} \right\|_{L_2(G)} < \infty$$

and certain boundary conditions (see Temlyakov (1990), Paskov (1993)). The worst case error of the quadrature Q on $H(K)$ is given by

$$e(Q, H(K)) = \sup_{\substack{h \in H(K) \\ \|h\|_{H(K)} \le 1}} |Ih - Qh| ,$$

where $Ih = \int_G h(x)\, dx$. Hence

$$\begin{aligned} e(Q, H(K))^2 &= \sup_{\substack{f \in L_2(G) \\ \|f\|_{L_2(G)} \le 1}} |(I-Q) T_B f|^2 \\ &= \|T_B^* (I-Q)\|_{L_2(G)}^2 \\ &= \int_G \left(\int_G B(x,t)\, dx - \sum_{i=1}^M v_i B(x_i, t) \right)^2 dt \\ &= D_B(Q)^2 . \end{aligned}$$

Consequently, $D_B(Q)$ is the worst case error of Q over the unit ball of the reproducing kernel Hilbert space $H(K)$. We obtain a general analogue of Zaremba's inequality (see Zaremba (1968)):

$$|Ih - Qh| \le D_B(Q) \|h\|_{H(K)} , \tag{10}$$

which is sharp in the sense that for each Q there exists an $h \in H(K)$, $h \neq 0$ such that equality is attained.

Note that we derived K from the given function B via (7), and obtained a symmetric non-negative definite continuous K. Conversely, for each such K we can find a $B \in C(L_2)$ such that (7) holds,

i. e. all relations discussed above are true for a general (continuous) reproducing kernel. Indeed, by Mercer's Theorem (see Werner (1995), Th. VI.4.2)

$$K(x,y) \;=\; \sum_i \lambda_i h_i(x) h_i(y)$$

with $\lambda_i \geq 0$ and $h_i \in C(G)$ as above. The convergence is uniform in $C(G^2)$. Setting

$$B(x,t) \;=\; \sum_i \lambda_i^{1/2} h_i(x) h_i(t)$$

we get $B \in C(L_2)$ (the series converges in the norm of $C(L_2)$), and representation (7) follows.

Let us finally mention the well-known equality of worst-case quadrature error over $H(K)$ and average error with respect to Borel measures (in particular Gaussian measures) with covariance kernel K. We refer to Sacks and Ylvisaker (1970) for details. Hence the discrepancy can also be interpreted as average integration error. Woźniakowski (1991) was the first to observe this for the classical L_2-discrepancy and its relation to the Wiener sheet measure, while corresponding measures for the case $B(x,t) = \frac{1}{(r!)^d}(t-x)_+^r$ are discussed in Paskov (1993).

4 A recursive algorithm for tensor product quadrature rules

As in the previous section we let $B \in C(G, L_2(G))$ and $Q = ((x_1, v_1), \ldots, (x_M, v_M))$ be a quadrature formula on $G = [0,1]^d$. According to (6),

$$D_B(Q)^2 \;=\; (IB, IB) \;-\; 2(IB, QB) \;+\; (QB, QB)\,.$$

Our algorithm will use this decomposition and treat the summands separately. For this purpose, we set

$$C_B \;=\; (IB, IB) \;\;,\;\; F_B(Q) \;=\; (IB, QB)\,,$$

and, with $R = ((y_1, w_1), \ldots, (y_N, w_N))$ another quadrature formula on G,

$$S_B(Q,R) \;=\; (QB, RB)\,.$$

So we have

$$D_B(Q)^2 \;=\; C_B - 2F_B(Q) + S_B(Q,Q)\,. \tag{11}$$

We will be concerned with the discrepancy of sums and of tensor products of quadrature formulas. Therefore, we now investigate the behaviour of C_B, $F_B(Q)$ and $S_B(Q,R)$, if Q, R are sums or tensor products of quadrature formulas. Let first Q, R be sums of quadrature formulas

$$Q \;=\; \sum_{i=1}^{p} Q_i \;\;,\;\; R \;=\; \sum_{j=1}^{q} R_j\,.$$

Then basic properties of the scalar product imply

$$\begin{aligned} F_B(Q) &= \sum_{i=1}^{p} F_B(Q_i)\,, \\ S_B(Q,R) &= \sum_{i=1}^{p}\sum_{j=1}^{q} S_B(Q_i,R_j)\,. \end{aligned}$$

In the following we consider tensor products of quadrature formulas. Let $d = d_1 + d_2$, $d_1, d_2 \geq 1$, and $G = G_1 \times G_2$, with $G_1 = [0,1]^{d_1}$, $G_2 = [0,1]^{d_2}$. Let further Q_1, Q_2 be quadrature formulas on G_1, G_2, respectively,

$$Q_1 = ((x_{1i_1}, v_{1i_1}))_{i_1=1}^{M_1}\ ,\ Q_2 = ((x_{2i_2}, v_{2i_2}))_{i_2=1}^{M_2}\ ,$$

and $Q = Q_1 \otimes Q_2 = ((x_i, v_i) : i = (i_1,i_2),\ i_1 = 1,...,M_1,\ i_2 = 1,...,M_2)$ their standard tensor product

$$x_i = (x_{1i_1}, x_{2i_2})\ ,\ v_i = v_{1i_1} \cdot v_{2i_2}\,.$$

We write I_1, I_2 for the integral over G_1, G_2, respectively. We assume

$$B(x,t) = B_1(x_1,t_1) \cdot B_2(x_2,t_2)$$

where $x = (x_1,x_2)$, $t = (t_1,t_2)$, and $B_1 \in C(G_1, L_2(G_1))$, $B_2 \in C(G_2, L_2(G_2))$. We have

$$QB(\cdot,t) = \sum_{i=(1,1)}^{(M_1,M_2)} v_i B(x_i,t) = Q_1B_1(\cdot,t_1) \cdot Q_2B_2(\cdot,t_2)\,.$$

Using these observations we can conclude

$$\begin{aligned} C_B &= \int_{G_1\times G_2} \left(\int_{G_1\times G_2} B_1(x_1,t_1)B_2(x_2,t_2)\,dx_1dx_2 \right)^2 dt_1dt_2 \\ &= \int_{G_1} \left(\int_{G_1} B_1(x_1,t_1)dx_1 \right)^2 dt_1 \cdot \int_{G_2} \left(\int_{G_2} B_2(x_2,t_2)dx_2 \right)^2 dt_2 \\ &= C_{B_1} \cdot C_{B_2}\ ; \end{aligned}$$

$$\begin{aligned} F_B(Q) &= \int_{G_1\times G_2} \left(\int_{G_1\times G_2} B_1(x_1,t_1)B_2(x_2,t_2)\,dx_1dx_2\, Q_1B_1(\cdot,t_1)Q_2B_2(\cdot,t_2) \right) dt_1dt_2 \\ &= \int_{G_1}\int_{G_1} B_1(x_1,t_1)\,dx_1\, Q_1B_1(\cdot,t_1)\,dt_1 \cdot \int_{G_2}\int_{G_2} B_2(x_2,t_2)\,dx_2\, Q_2B_2(\cdot,t_2)\,dt_2 \\ &= (I_1B_1, Q_1B_1)_{L_2(G_1)} \cdot (I_2B_2, Q_2B_2)_{L_2(G_2)} \\ &= F_{B_1}(Q_1) \cdot F_{B_2}(Q_2)\,. \end{aligned}$$

Finally, we want to consider $S_B(Q,R)$ under this aspect. Let $R = R_1 \otimes R_2$ also be a tensor product of two quadrature formulas R_1, R_2 on G_1, G_2, respectively, with

$$R_1 = (y_{1j_1}, w_{1j_1})_{j_1=1}^{N_1}\ ,\ R_2 = (y_{2j_2}, w_{2j_2})_{j_2=1}^{N_2}\ .$$

Then similar transformations give

$$\begin{aligned} S_B(Q,R) &= \int_G \left(\sum_{i=(1,1)}^{(M_1,M_2)} v_i B(x_i,t) \sum_{j=(1,1)}^{(N_1,N_2)} w_j B(y_j,t) \right) dt \\ &= \int_{G_1} \sum_{i_1=1}^{M_1} \sum_{j_1=1}^{N_1} v_{1i_1} w_{1j_1} B_1(x_{1i_1},t_1) B_1(y_{1j_1},t_1)\, dt_1 \cdot \\ &\quad \cdot \int_{G_2} \sum_{i_2=1}^{M_2} \sum_{j_2=1}^{N_2} v_{2i_2} w_{2j_2} B_2(x_{2i_2},t_2) B_2(y_{2j_2},t_2)\, dt_2 \\ &= (Q_1 B_1, R_1 B_1)_{L_2(G_1)} \cdot (Q_2 B_2, R_2 B_2)_{L_2(G_2)} \\ &= S_{B_1}(Q_1,R_1) \cdot S_{B_2}(Q_2,R_2)\,. \end{aligned}$$

Now we are able to describe the recursive algorithm for the computation of the discrepancy of Smolyak quadrature rules. For the rest of this section we make the following assumptions. We fix a one-dimensional $B^{(1)} \in C([0,1], L_2([0,1]))$ and let, for $d > 1$, $B^{(d)} \in C([0,1]^d, L_2([0,1]^d))$ be defined by

$$B^{(d)}(x,t) = \prod_{k=1}^{d} B^{(1)}(\xi_k, \tau_k)\,,$$

where $x = (\xi_1, ..., \xi_d)$ and $t = (\tau_1, ..., \tau_d)$. Let $\{Q_n\}$ be a sequence of arbitrary quadrature formulas on $[0,1]$. As introduced in section 2, the Smolyak quadrature rule on $[0,1]^d$, $d \geq 2$, satisfies the recursion (2)

$$Q_n^{(d)} = \sum_{i=0}^{n} R_i \otimes Q_{n-i}^{(d-1)}\,,$$

where $R_i = Q_i - Q_{i-1}$, $Q_{-1} \equiv 0$ and $Q_n^{(1)} = Q_n$. We fix a maximal level n_{max} and apply a recursion over d to calculate all quantities $F_{B^{(d)}}(Q_n^{(d)})$ and $S_{B^{(d)}}(Q_m^{(d)}, Q_n^{(d)})$ for $m,n = 0,1,\ldots,n_{max}$.

The recursion starts from the univariate terms $F_{B^{(1)}}(Q_n)$, $S_{B^{(1)}}(Q_m, Q_n)$. Their computation will be discussed in the next section. From these terms we get immediately

$$\begin{aligned} F_{B^{(1)}}(R_n) &= F_{B^{(1)}}(Q_n) - F_{B^{(1)}}(Q_{n-1}) \\ S_{B^{(1)}}(R_m,R_n) &= S_{B^{(1)}}(Q_m,Q_n) - S_{B^{(1)}}(Q_{m-1},Q_n) - S_{B^{(1)}}(Q_m,Q_{n-1}) + S_{B^{(1)}}(Q_{m-1},Q_{n-1}). \end{aligned}$$

Using the relations

$$\begin{aligned} F_{B^{(d)}}(Q_n^{(d)}) &= \sum_{i=0}^{n} F_{B^{(d)}}(R_i \otimes Q_{n-i}^{(d-1)}) \\ &= \sum_{i=0}^{n} F_{B^{(1)}}(R_i) \cdot F_{B^{(d-1)}}(Q_{n-i}^{(d-1)}) \end{aligned}$$

and

$$\begin{aligned} S_{B^{(d)}}(Q_m^{(d)}, Q_n^{(d)}) &= S_{B^{(d)}}\left(\sum_{i=0}^{m} R_i \otimes Q_{m-i}^{(d-1)}, \sum_{j=0}^{n} R_j \otimes Q_{n-j}^{(d-1)}\right) \\ &= \sum_{i=0}^{m}\sum_{j=0}^{n} S_{B^{(d)}}\left(R_i \otimes Q_{m-i}^{(d-1)}, R_j \otimes Q_{n-j}^{(d-1)}\right) \\ &= \sum_{i=0}^{m}\sum_{j=0}^{n} S_{B^{(1)}}(R_i, R_j) \cdot S_{B^{(d-1)}}\left(Q_{m-i}^{(d-1)}, Q_{n-j}^{(d-1)}\right) \end{aligned} \tag{12}$$

we get the terms $F_{B^{(d)}}(Q_n^{(d)})$ and $S_{B^{(d)}}(Q_m^{(d)}, Q_n^{(d)})$ $(m, n = 0, 1, \ldots, n_{max})$ in all dimensions. Finally, the discrepancy $D_B(Q_n^{(d)})$ is given by (11), where obviously

$$C_{B^{(d)}} = (C_{B^{(1)}})^d .$$

Each dimension step takes $O(n_{max}^4)$ operations. Observe that n_{max} is the number of levels, so usually $n_{max} = O(\log P)$, where P is the number of nodes in the quadrature $Q_{n_{max}}$. Since (12) is, in fact, a discrete convolution, one could apply the FFT to reduce the effort to $O(n_{max}^2 \log n_{max})$. But in all of our computations there was no need of doing this, since the main effort had to be spent in the one-dimensional case.

5 Fast computation in one dimension

In order to get the algorithm of section 4 started, we have to compute a certain number of terms of the form $F_{B^{(1)}}(Q)$ and $S_{B^{(1)}}(Q, R)$ for univariate quadratures $Q = ((x_1, v_1), ..., (x_M, v_M))$ and $R = ((y_1, w_1), ..., (y_N, w_N))$ on $[0, 1]$. This depends, of course, on the concrete form of $B^{(1)}$. In this chapter we develop algorithms for the classical L_2-discrepancy and its r-smooth generalizations mentioned before. Thus we let r be a non-negative integer and put

$$B^{(1)}(x, t) = B_r^{(1)}(x, t) = \frac{1}{r!}(t - x)_+^r .$$

In the case $r = 0$, this is understood as $B_0^{(1)}(x, t) = (t - x)_+^0 = \chi_{[0,t)}(x)$. From now on this choice is fixed, and we indicate the dependence on the parameter r by the subscript: C_r, F_r, S_r. A direct calculation gives

$$\begin{aligned} C_r(Q) &= ((r+1)!)^{-2}(2r+3)^{-1}, \\ F_r(Q) &= (r!)^{-2}(r+1)^{-1}\sum_{i=1}^{M}\sum_{j=0}^{r}\binom{r}{j} v_i\,(-x_i)^j\,\frac{1 - x_i^{2r+2-j}}{2r+2-j} \end{aligned} \tag{13}$$

and

$$S_r(Q, R) = (r!)^{-2}\sum_{i_1=1}^{M}\sum_{i_2=1}^{N}\sum_{j_1, j_2=0}^{r}\binom{r}{j_1}\binom{r}{j_2} v_{i_1} w_{i_2} (-x_{i_1})^{j_1} (-y_{i_2})^{j_2}\,\frac{1 - \max(x_{i_1}, y_{i_2})^{2r-j_1-j_2+1}}{2r - j_1 - j_2 + 1} . \tag{14}$$

First we discuss the case of general quadratures, later on we shall study composite quadratures. For an arbitrary quadrature, the direct computation of $F_r(Q)$ by (13) takes $O(M)$ operations. The direct computation of $S_r(Q,R)$ by (14) would require $O(MN)$ operations. Here one could use the algorithm proposed by Heinrich (1995), which would need $O(M \log M + N \log N)$ operations. But in the one-dimensional case there is, in fact, no need of this recursion. The following simple algorithm needs not more than $O(M \log M + N \log N)$ operations, either: First we order the node sets of Q and R (let us mention that the node sets of many quadrature rules are ordered by their definition). Now assuming $x_1 \le x_2 \le \ldots \le x_M$ and $y_1 \le y_2 \le \ldots \le y_N$, we determine for each $i_1 = 1, \ldots, M$ an index $\nu(i_1)$ such that $x_{i_1} \ge y_{i_2}$ for all $i_2 \le \nu(i_1)$ and $x_{i_1} < y_{i_2}$ for $i_2 > \nu(i_1)$. Using this property we can rewrite the direct formula (14) as follows

$$S_r(Q,R) \;=\; \frac{1}{(r!)^2} \sum_{j_1,j_2=0}^{r} \binom{r}{j_1}\binom{r}{j_2} \frac{(-1)^{j_1+j_2}}{2r-j_1-j_2+1} \cdot \gamma_r(j_1,j_2)$$

where

$$\begin{aligned}
\gamma_r(j_1,j_2) \;&=\; \sum_{i_1=1}^{M}\sum_{i_2=1}^{N} v_{i_1}\, w_{i_2}\, x_{i_1}^{j_1}\, y_{i_2}^{j_2}\, (1-\max(x_{i_1},y_{i_2})^{2r-j_1-j_2+1}) \\
&=\; \sum_{i_1=1}^{M} v_{i_1}\, x_{i_1}^{j_1}\, (1-x_{i_1}^{2r-j_1-j_2+1}) \sum_{i_2=1}^{\nu(i_1)} w_{i_2}\, y_{i_2}^{j_2} \\
&\quad + \sum_{i_1=1}^{M} v_{i_1}\, x_{i_1}^{j_1} \sum_{i_2=\nu(i_1)+1}^{N} w_{i_2}\, y_{i_2}^{j_2}\, (1-y_{i_2}^{2r-j_1-j_2+1})\,. \qquad (15)
\end{aligned}$$

Both sums in (15) can be computed in $O(M+N)$ operations, if the inner sums are added up successively.

Now we turn to composite quadrature rules. Let Q_n be the composite quadrature formula which is constructed by applying a basis quadrature $Q = ((x_0,v_0),\ldots(x_q,v_q))$ on p^n subintervals $[\frac{i-1}{p^n}, \frac{i}{p^n}]$, $i = 1,\ldots,p^n$, with $p > 1$ some natural number:

$$Q_n f \;=\; \sum_{i=1}^{p^n}\sum_{k=0}^{q} \frac{v_k}{p^n} \cdot f\left(\frac{i-1+x_k}{p^n}\right)\,.$$

In the following we will make use of this structure of Q_n to transform the direct formulas for $F_r(Q_n)$ and $S_r(Q_m,Q_n)$ into a faster computable form. Therefore we have to introduce some technical means.

In the transformed formulas terms like $\sum_{i=1}^{m} i^k$ or $\sum_{n=1}^{\nu} k^{(n)}$ will occur, where $k^{(n)}$ is the factorial polynomial

$$k^{(n)} \;=\; k(k-1)\cdot\ldots\cdot(k-n+1)\,.$$

To calculate these sums in an efficient way we need the so-called Stirling numbers of the first and second kinds. These are defined recursively, the Stirling numbers of the first kind $\sigma_i^{(n)}$ $(n = 1,2,\ldots; i = 0,1,\ldots,n)$ by

$$\begin{aligned}
\sigma_0^{(n)} \;&=\; 0\,,\; \sigma_n^{(n)} \;=\; 1\,,\; n = 1,2,\ldots\,, \\
\sigma_i^{(n+1)} \;&=\; \sigma_{i-1}^{(n)} - n\cdot\sigma_i^{(n)}\,,\; n = 1,2,\ldots\,,\; i = 1,\ldots,n\,,
\end{aligned}$$

and the Stirling numbers of the second kind $s_i^{(n)}$ $(n = 1, 2, ...; i = 0, 1, ..., n)$ by

$$\begin{aligned} s_0^{(n)} &= 0\,,\ s_n^{(n)} = 1\,,\ n = 1, 2, \dots\,, \\ s_i^{(n+1)} &= s_{i-1}^{(n)} + i \cdot s_i^{(n)}\,,\ n = 1, 2, \dots\,,\ i = 1, ..., n\,. \end{aligned}$$

The following well-known properties (see e. g. Abramowitz, Stegun (1972)) will be used in our transformation. If $k, n \geq 1$,

$$k^{(n)} = \sum_{i=1}^{n} \sigma_i^{(n)} k^i\,, \tag{16}$$

$$k^n = \sum_{i=1}^{n} s_i^{(n)} k^{(i)}. \tag{17}$$

It is easily checked by induction that for $m, n \geq 1$

$$\sum_{j=1}^{m} j^{(n)} = (n+1)^{-1}(m+1)^{(n+1)}\,.$$

This together with (17) gives

$$\begin{aligned} \sum_{j=1}^{m} j^n &= \sum_{i=1}^{n} \frac{s_i^{(n)}}{i+1} \cdot (m+1)^{(i+1)} \\ &= (m+1)m\left[\frac{1}{2} + (m-1)\left[\dots + (m-n+1) \cdot \frac{s_n^{(n)}}{n+1}\right]\dots\right]. \end{aligned} \tag{18}$$

The $n_{max} + 1$ terms $F_r(Q_n)$ will be treated as follows:

$$\begin{aligned} F_r(Q_n) &= \frac{(r!)^{-2}}{(r+1)} \sum_{i=1}^{p^n} \sum_{k=0}^{q} \sum_{j=0}^{r} \binom{r}{j} \frac{(-1)^j}{2r+2-j} \frac{v_k}{p^n} \left(\frac{i-1+x_k}{p^n}\right)^j \left[1 - \left(\frac{i-1+x_k}{p^n}\right)^{2r+2-j}\right] \\ &= \frac{(r!)^{-2}}{(r+1)} \sum_{j=0}^{r} \binom{r}{j} \frac{(-1)^j}{2r+2-j} \sum_{k=0}^{q} \frac{v_k}{p^n} \cdot \varphi_r^n(j,k)\,, \end{aligned}$$

where

$$\begin{aligned} \varphi_r^n(j,k) &= \sum_{i=1}^{p^n} \left[p^{-nj}(i-1+x_k)^j - p^{-n(2r+2)}(i-1+x_k)^{2r+2}\right] \\ &= p^{-nj} \sum_{l=0}^{j} \binom{j}{l} x_k^{j-l} \sum_{i=1}^{p^n} (i-1)^l - p^{-n(2r+2)} \sum_{l=0}^{2r+2} \binom{2r+2}{l} x_k^{2r+2-l} \sum_{i=1}^{p^n} (i-1)^l\,. \end{aligned}$$

Using (18) we can calculate $\varphi_r^n(j,k)$ for each $j = 0, ..., r$, $k = 0, ..., q$ and $n = 0, ..., n_{max}$ in $O(r^2)$ operations independent of k and n. Hence, each $F_r(Q_n)$ takes $O(r^3 q)$ operations, and all those terms require $O(r^3 q n_{max})$ operations.

The same principle works also for the terms $S_r(Q_n, Q_m)$. However, the presence of the maximum introduces some more technical difficulties. As in the case of arbitrary quadrature formulas we will split the double sum over the points, expressing the maximum explicitly, and reduce the influence of the number of points step by step to sums of type (18), which can be computed efficiently.

We rewrite $S_r(Q_m, Q_n)$ in a more convenient form

$$S_r(Q_m, Q_n) = \frac{1}{(r!)^2} \sum_{j_1,j_2=0}^{r} \binom{r}{j_1}\binom{r}{j_2} \frac{(-1)^{j_1+j_2}}{2r-j_1-j_2+1} \sum_{k_1=0}^{q}\sum_{k_2=0}^{q} \frac{v_{k_1}}{p^m}\frac{v_{k_2}}{p^n}\, \psi_r^{m,n}(j_1,j_2,k_1,k_2)\,,$$

where the main complexity of the task is concentrated in the term $\psi_r^{m,n}$ with

$$\begin{aligned}\psi_r^{m,n}(j_1,j_2,k_1,k_2) &= \sum_{i_1=1}^{p^m}\sum_{i_2=1}^{p^n} \left(\frac{i_1-1+x_{k_1}}{p^m}\right)^{j_1} \left(\frac{i_2-1+x_{k_2}}{p^n}\right)^{j_2} \cdot \\ &\quad \cdot \left(1-\max\left(\frac{i_1-1+x_{k_1}}{p^m}, \frac{i_2-1+x_{k_2}}{p^n}\right)^{2r-j_1-j_2+1}\right). \end{aligned} \tag{19}$$

Without loss of generality we assume $m \le n$. Define for $i_1 = 1, ..., p^m$

$$\nu(i_1) = p^{n-m}(i_1-1) + 1 + \lfloor p^{n-m} x_{k_1} - x_{k_2} \rfloor$$

where $\lfloor a \rfloor$ denotes the largest integer not exceeding a. Note that $\nu(i_1)$ is a linear function in i_1. Clearly $i_2 \le \nu(i_1)$ if and only if

$$\frac{i_1-1+x_{k_1}}{p^m} \ge \frac{i_2-1+x_{k_2}}{p^n}\,.$$

It follows from (19) that

$$\begin{aligned}&\psi_r^{m,n}(j_1,j_2,k_1,k_2) \\ &\quad = \sum_{i_1=1}^{p^m} \left(\frac{i_1-1+x_{k_1}}{p^m}\right)^{j_1} \cdot \sum_{i_2=1}^{p^n} \left(\frac{i_2-1+x_{k_2}}{p^n}\right)^{j_2} \\ &\qquad - \sum_{i_1=1}^{p^m} \left(\frac{i_1-1+x_{k_1}}{p^m}\right)^{j_1} \cdot \sum_{i_2=1}^{p^n} \left(\frac{i_2-1+x_{k_2}}{p^n}\right)^{2r-j_1+1} \\ &\qquad - \sum_{i_1=1}^{p^m}\sum_{i_2=1}^{\nu(i_1)} \left(\frac{i_1-1+x_{k_1}}{p^m}\right)^{j_1} \left(\frac{i_2-1+x_{k_2}}{p^n}\right)^{j_2} \left[\left(\frac{i_1-1+x_{k_1}}{p^m}\right)^{\varrho} - \left(\frac{i_2-1+x_{k_2}}{p^n}\right)^{\varrho}\right], \end{aligned} \tag{20}$$

where $\varrho = 2r - j_1 - j_2 + 1$.

The first two parts of the sum (20) can be computed in $O(r^2)$ operations using again (18). The last part must be handled separately by applying successively (18), (17) and changing the order of summation while the sum over the points is not reduced to the basis form computable by (18). To give an idea of this procedure, we perform just the first steps:

$$\begin{aligned}&\sum_{i_1=1}^{p^m}\sum_{i_2=1}^{\nu(i_1)} \left(\frac{i_1-1+x_{k_1}}{p^m}\right)^{j_1} \left(\frac{i_2-1+x_{k_2}}{p^n}\right)^{j_2} \left[\left(\frac{i_1-1+x_{k_1}}{p^m}\right)^{\varrho} - \left(\frac{i_2-1+x_{k_2}}{p^n}\right)^{\varrho}\right] \\ &\quad = \frac{1}{p^{m(2r-j_2+1)}p^{nj_2}}\,\Sigma_1 - \frac{1}{p^{mj_1}p^{n(2r-j_1+1)}}\,\Sigma_2\,, \end{aligned}$$

and

$$\begin{aligned}
\Sigma_1 &= \sum_{i_1=1}^{p^m}\sum_{i_2=1}^{\nu(i_1)}(i_1-1+x_{k_1})^{2r-j_2+1}(i_2-1+x_{k_2})^{j_2} \\
&= \sum_{i_1=1}^{p^m}(i_1-1+x_{k_1})^{2r-j_2+1}\sum_{l_2=0}^{j_2}\binom{j_2}{l_2}x_{k_2}^{j_2-l_2}\sum_{i_2=1}^{\nu(i_1)}(i_2-1)^{l_2} \\
&= \sum_{l_2=0}^{j_2}\binom{j_2}{l_2}x_{k_2}^{j_2-l_2}\sum_{i_1=1}^{p^m}(i_1-1+x_{k_1})^{2r-j_2+1}\left(\sum_{\alpha=1}^{l_2}\frac{s_\alpha^{(l_2)}}{\alpha+1}\nu(i_1)^{(\alpha+1)}+\nu(i_1)\cdot\delta_{l_20}\right) \\
&= \sum_{l_2=0}^{j_2}\binom{j_2}{l_2}x_{k_2}^{j_2-l_2}\sum_{i_1=1}^{p^m}(i_1-1+x_{k_1})^{2r-j_2+1}\left(\sum_{\alpha=1}^{l_2}\frac{s_\alpha^{(l_2)}}{\alpha+1}\sum_{\beta=1}^{\alpha+1}\sigma_\beta^{(\alpha+1)}\nu(i_1)^\beta+\nu(i_1)\cdot\delta_{l_20}\right) \\
&= \sum_{l_2=0}^{j_2}\binom{j_2}{l_2}x_{k_2}^{j_2-l_2}\left(\sum_{\alpha=1}^{l_2}\frac{s_\alpha^{(l_2)}}{\alpha+1}\sum_{\beta=1}^{\alpha+1}\sigma_\beta^{(\alpha+1)}\sum_{i_1=1}^{p^m}(i_1-1+x_{k_1})^{2r-j_2+1}\nu(i_1)^\beta\right. \\
&\qquad \left.+\delta_{l_20}\cdot\sum_{i_1=1}^{p^m}(i_1-1+x_{k_1})^{2r-j_2+1}\nu(i_1)\right)
\end{aligned}$$

where $\delta_{l_20}=1$ if $l_2=0$ and $\delta_{l_20}=0$ otherwise. Since $\nu(i_1)^\beta$ is a polynomial of degree β in i_1, we can use the binomial formula again to transform this multiple sum into a form, where the influence of the sum over the points is reduced to sums of type (18), with exponents not greater than $2r-j_2+1+\beta\le 2r+2$. Analoguously we treat Σ_2. This leads to very large expressions, which nevertheless can be computed in a number of operations depending on the smoothness r as $O(r^6)$, but not on the parameters of the quadrature q,n,m. Thus we derived an algorithm which is able to calculate all $(n_{max}+1)^2$ terms in $O(q^2r^8n_{max}^2)$ operations.

Now we can analyze the complexity of the whole process of computing $D_B(Q_n^{(d)})$ (for $B(x,t)=B_r^{(d)}(x,t)=\frac{1}{(r!)^d}(t-x)_+^r$). We fix $r\ge 0$. Assume that there are reals $p>1$, $c_1,c_2>0$ such that the number of nodes P_n in the one-dimensional quadratures Q_n satisfies

$$c_1p^n\le P_n\le c_2p^n\,.$$

This is a natural assumption for Smolyak quadratures. Fix n_{max} and denote $P=P_{n_{max}}$. Obviously, $n_{max}=O(\log P)$.

Let us first consider the case of arbitrary quadratures Q_n. The ordering of the nodes of Q_n, $n=0,...,n_{max}$ needs $O(P\log P)$ operations and the computation of $F_r(Q_n)$ $(n=0,...,n_{max})$ takes $O(P)$ operations. The calculation of $S_r(Q_n,Q_m)$ for fixed n and $m=0,...,n$ can be accomplished in $O(nP_n)$ and that of $S_r(Q_n,Q_m)$ for $m=0,...,n$, $n=0,...,n_{max}$ in $O(n_{max}P)=O(P\log P)$ operations. Each dimension step costs $O((\log P)^4)$, so we get

$$O(P\log P+d(\log P)^4)$$

as the resulting complexity, where the constant in the O-notation depends only on c_1,c_2,p and r.

In the case that Q_n is the composite of a basis quadrature, the terms $F_r(Q_n)$ and $S_r(Q_m,Q_n)(m,n=0,...,n_{max})$ can be computed in $O(n_{max}^2)=O((\log P)^2)$ operations. Together with the dimension

step we get the complexity

$$O(d(\log P)^4)$$

where this time the constant depends on c_1, c_2, p, r and q — the number of points in the basic rule.

6 Numerical results

The following numerical experiments shall serve two different purposes: Firstly, we compare the discrepancies of Smolyak quadrature rules with those of Monte Carlo and quasi Monte Carlo quadratures with usual parameters, and secondly, we want to demonstrate the power of the new algorithm by calculating the discrepancies for extremely large node sets or very high dimension. Throughout this section, we fix the function

$$B(x,t) = B_r^{(d)}(x,t) = \frac{1}{(r!)^d}(t-x)_+^r\,,\ (x,t \in G)\,,$$

and we write D_r for $D_{B_r^{(d)}}$. As already mentioned, for $r = 0$ we get the usual L_2-discrepancy and for $r \geq 1$ the r-smooth L_2-discrepancy introduced by Paskov (1993).

All implementations were carried out on a workstation HP 9000/712/60 in C^{++}. Since cancellation proved to be a real problem due to the similarity of the terms $C_r(Q)$, $F_r(Q)$ and $S_r(Q,Q)$ in (11) for large M, in all calculations quadruple precision was used.

As far as Monte Carlo integration is concerned, we do not use any concrete random number generator, but calculate the square mean of the L_2-discrepancy. Let $Qf = \frac{1}{M}\sum_{i=1}^M f(\zeta_i)$, with ζ_i being independent, uniformly distributed on $G = [0,1]^d$ random variables. Then the square mean $(\mathbb{E}D_r(Q)^2)^{1/2}$ is given by

$$\begin{aligned}
\mathbb{E}D_r(Q)^2 &= \mathbb{E}\int_G \left(IB_r^{(d)}(\cdot,t) - \frac{1}{M}\sum_{i=1}^M B_r^{(d)}(\zeta_i,t)\right)^2 dt \\
&= \frac{1}{M}\int_G \mathbb{E}\left(IB_r^{(d)}(\cdot,t) - B_r^{(d)}(\zeta_1,t)\right)^2 dt \\
&= \frac{1}{M}\int_G \left(\int_G B_r^{(d)}(x,t)^2 dx - \left(\int_G B_r^{(d)}(x,t)dx\right)^2\right) dt \qquad (21)\\
&= \frac{1}{M(r!)^{2d}}\int_G \left(\int_G (t-x)_+^{2r} dx - \left(\int_G (t-x)_+^r dx\right)^2\right) dt \\
&= \frac{1}{M(r!)^{2d}}\left(\left(\frac{1}{(2r+1)(2r+2)}\right)^d - \left(\frac{1}{(r+1)^2(2r+3)}\right)^d\right).
\end{aligned}$$

In the special case $r = 0$ we have (compare Tezuka (1995))

$$(\mathbb{E}D_0(Q)^2)^{1/2} = \left(\frac{1}{M}(2^{-d} - 3^{-d})\right)^{1/2}.$$

The Smolyak quadratures taking part in the comparison are denoted by TR, NC4 and CC. As the names reflect, TR bases on the trapezoidal rule, NC4 on the Newton–Cotes formula of degree 4

and CC on the Clenshaw–Curtis rule (which was also considered by Novak and Ritter (1996)). In all three quadratures Q_0 is chosen as the midpoint rule

$$Q_0 f = f(0.5),$$

because otherwise the number of grid points would increase exponentially in d. Hence, the sequence (Q_n) in TR is defined as

$$\begin{aligned} Q_0 f &= f(0.5) \\ Q_n f &= \frac{1}{2^{n+1}} \sum_{i=1}^{2^n} \left[f\left(\frac{i-1}{2^n}\right) + f\left(\frac{i}{2^n}\right) \right], \; n \geq 1, \end{aligned} \tag{22}$$

in NC4 as

$$\begin{aligned} Q_0 f &= f(0.5) \\ Q_1 f &= \frac{1}{6}[f(0) + 4f(0.5) + f(1)] \\ Q_n f &= \frac{2^{2-n}}{90} \sum_{i=1}^{2^{n-2}} \left[7f\left(\frac{i-1}{2^{n-2}}\right) + 32f\left(\frac{i-\frac{3}{4}}{2^{n-2}}\right) + 12f\left(\frac{i-\frac{1}{2}}{2^{n-2}}\right) + 32f\left(\frac{i-\frac{1}{4}}{2^{n-2}}\right) + 7f\left(\frac{i}{2^{n-2}}\right) \right] \end{aligned} \tag{23}$$

for $n \geq 2$, and in CC as

$$Q_0 f = f(0.5), \; Q_n f = \sum_{i=0}^{2^n} a_i^{(n)} f(x_i^{(n)}), \; n \geq 1, \tag{24}$$

where for $n \geq 1$:

$$\begin{aligned} x_i^{(n)} &= \frac{1}{2}\left(1 - \cos\frac{\pi i}{2^n}\right), \; i = 0, ..., 2^n, \\ a_i^{(n)} &= 2^{-n}\left(1 - \frac{\cos(\pi i)}{2^{2n}-1} - 2\sum_{k=1}^{2^{n-1}-1} \frac{1}{4k^2-1} \cos\frac{2\pi k i}{2^n}\right), \; i = 1, ..., 2^n - 1 \\ a_0^{(n)} &= a_{2^n}^{(n)} = \frac{1}{2(2^{2n}-1)}, \end{aligned}$$

(see Brass (1977)). This definition guarantees that the Smolyak quadrature $Q_n^{(d)}$ uses the same number of grid points, whether the underlying sequence of one-dimensional quadratures is (22), (23) or (24).

In tables 1 and 2 the L_2-discrepancies of these Smolyak quadratures are compared with the square mean $(\mathbb{E}D_r(Q)^2)^{1/2}$ of the L_2-discrepancy of Monte Carlo integration and the L_2-discrepancy of quasi Monte Carlo integration using the Hammersley sequence.

The number of nodes in tables 1 and 2 is limited by the poor performance of the direct algorithm, which must be used for the quasi Monte Carlo quadrature and takes $O(dr^2M^2)$ operations. Therefore, we restricted ourselves to $M \approx 10^4$ nodes. On the other hand, in order to compare only quadratures with the same number of nodes, this number must be governed in each row of the

d	M	MC	Hammersley	TR	NC4	CC
5	19313	1.185e-03	1.564e-04	5.144e-03	1.759e-02	6.161e-03
10	41265	1.524e-04	8.164e-05	2.094e-02	4.287e-02	1.165e-02
15	40001	2.759e-05	4.276e-05	2.546e-02	3.133e-02	4.934e-03
20	11561	9.081e-06	1.102e-04	1.303e-02	1.299e-02	2.034e-03
50	5101	4.173e-10	2.345e-04	2.029e-06	2.021e-06	7.540e-07
100	20201	6.249e-18	5.761e-05	2.589e-13	2.584e-13	1.057e-13

Table 1: L_2-discrepancies for $r = 0$ and $M \approx 10^4$.

tables by the number of grid points used by the Smolyak quadratures, which can be calculated by (3). We tried to choose the parameter n_{max} in such a way that M is as near as possible to 10^4.

Table 1 shows that from dimension 15 on the discrepancy of the quasi Monte Carlo quadrature is at a standstill while the other quadratures are still decreasing steadily. For dimensions $d \geq 50$ even the Smolyak quadratures have a lower discrepancy than the Hammersley sequence although they are rather designed for higher smoothness. If the integrand has some smoothness properties (see table 2) then the Smolyak quadratures have a lower discrepancy than the Hammersley sequence already for $d \geq 10$ (Clenshaw–Curtis even for $d \geq 5$).

In fig. 2 and fig. 3 we fixed the dimension $d = 10$ and compared the L_2-discrepancies of the three Smolyak quadratures with the expected value of the L_2-discrepancy of Monte Carlo integration for various smoothnesses and a number of grid points up to 10^{10}. In all diagrams, axes are logarithmically scaled, except the abscissa in fig. 5.

From the diagrams as well as from the tables it can be seen that Monte Carlo integration is the best choice if the function is not smooth ($r = 0$). However, fig. 4 allows us to conjecture that for very large point sets ($M > 10^{35}$) even for $r = 0$ the Smolyak quadratures will beat Monte Carlo integration.

In the case of higher smoothness the discrepancies of the Smolyak quadratures NC4 and CC become smaller than the discrepancies of Monte Carlo integration already for moderate dimensions ($d \geq 10$, see table 2) and moderate numbers of points (see fig. 3).

d	M	MC	Hammersley	TR	NC4	CC
5	6993	7.581e-08	6.092e-08	6.263e-08	3.186e-10	8.145e-10
10	8801	4.284e-13	3.605e-11	5.199e-13	3.948e-13	3.951e-13
15	5021	3.595e-18	3.503e-14	7.480e-19	8.865e-19	8.865e-19
20	11561	1.502e-23	8.493e-18	9.220e-25	9.516e-25	9.516e-25

d	M	MC	Hammersley	TR	NC4	CC
5	6993	1.954e-14	6.164e-14	2.115e-13	3.644e-15	3.524e-15
10	8801	2.847e-26	3.036e-23	9.143e-27	9.832e-27	9.832e-27
15	5021	6.161e-38	2.750e-32	1.004e-39	1.004e-39	1.004e-39
20	11561	6.635e-50	6.172e-42	1.006e-52	1.006e-52	1.006e-52

Table 2: L_2-discrepancies for $M \approx 10^4$ and higher smoothness: $r = 2$ (top), $r = 4$ (bottom).

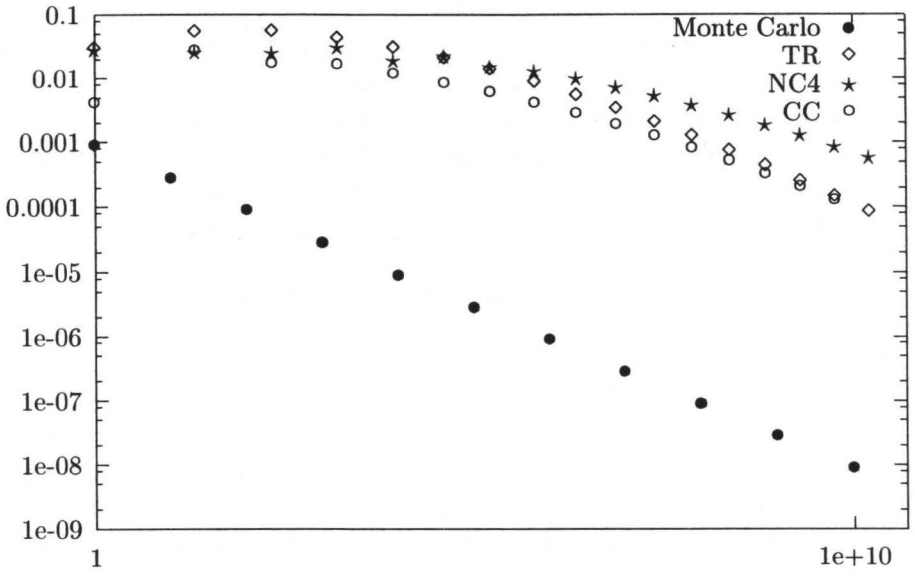

Figure 2: L_2-discrepancies in dimension 10 for $r = 0$

Another interesting feature is the fact, that the L_2-discrepancies of all three Smolyak quadratures are almost equal for high dimensions, as can be seen in fig. 5 and the last two lines of table 2. To explain that we have to recall a property of the function $B_r^{(d)}(x,t)$. If $r > 0$ and the dimension increases, then we are approximating the integral of functions which fit closer and closer into the corner $(0,0,\ldots,0)$ of the integration domain $G = [0,1]^d$. If the number of nodes is not high enough, Smolyak quadratures underestimate the integral of those functions extremely, so both terms $(IB_r^{(d)}, QB_r^{(d)})$ and $(QB_r^{(d)}, QB_r^{(d)})$ in (6) become so small that they do not have any influence on the value $D_r(Q)$. So for $r > 0$, high dimensions and moderate numbers of nodes the L_2-discrepancy is approximately

$$D_r(Q) \approx \sqrt{(IB_r^{(d)}, IB_r^{(d)})} = ((r+1)!)^{-d}(2r+3)^{-d/2}. \tag{25}$$

This circumstance also influences the non-convergent behaviour of TR in fig. 3 for $r = 4$. For numbers of quadrature points up to 10^4 (25) holds. Later a slight overestimation occurs, until convergence sets in from about 10^{10} points on, while NC4 and CC show convergence already for numbers of nodes above 10^5.

Although the L_2-discrepancies of the Smolyak quadratures seem to be equal for $r = 0$ and high dimensions, too (see fig. 5), this is not the case. A closer look shows that they are almost of the same order, but not equal (see table 3 below). In contrast to the situation described above, for $r = 0$ the term $(QB_r^{(d)}, QB_r^{(d)})$ dominates in (6), while the other terms are loosing their influence on $D_r(Q)$ with increasing dimension. These observations might seem pathological, but they simply reflect the fact that in very high dimensions for the respective classes of functions no convergence can be obtained with 10^6 points — for most functions integral and quadrature are orders away from each other.

A further purpose of fig. 4 and fig. 5 is to show the power of the new algorithm presented in this paper. For Smolyak quadrature rules, which use composite quadratures in the underlying one-

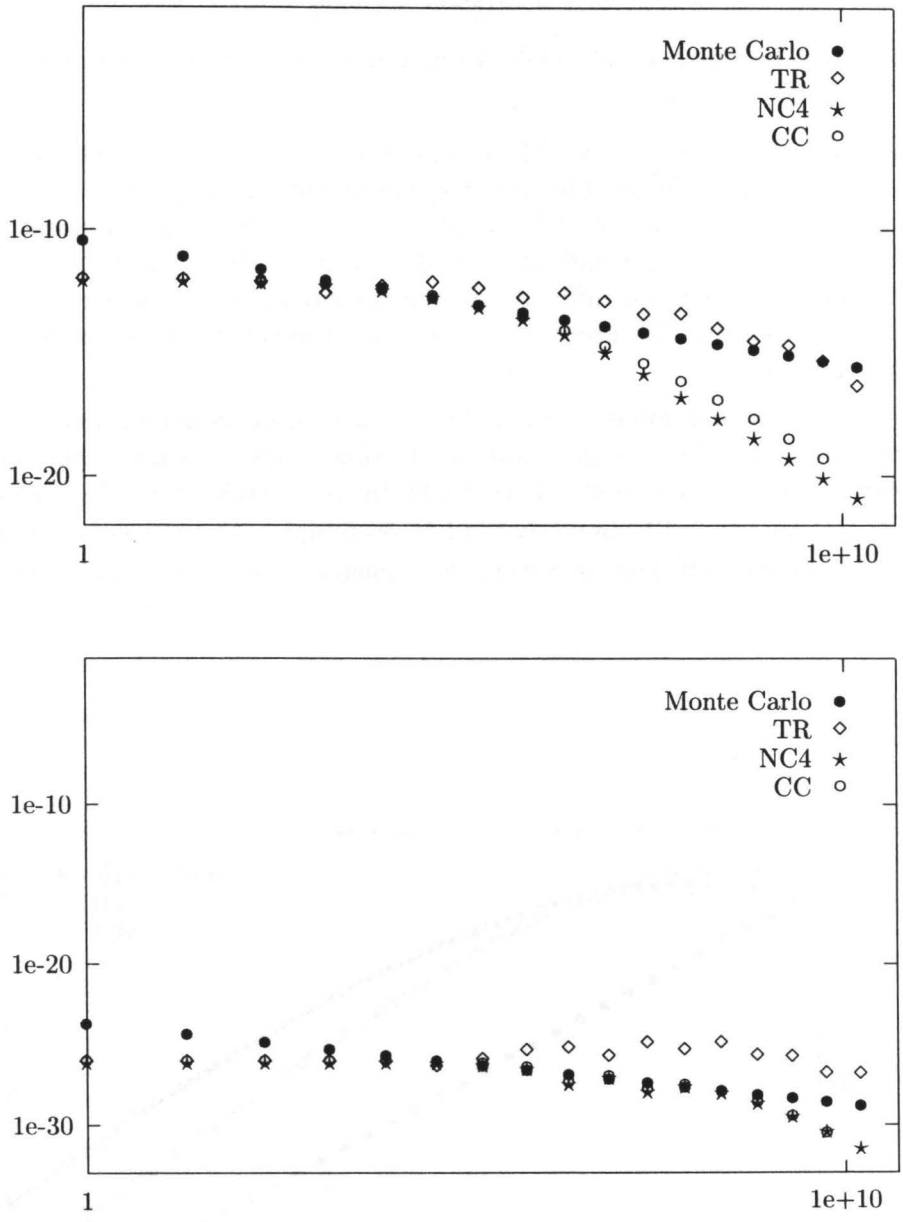

Figure 3: L_2-discrepancies in dimension 10 for higher smoothness: $r = 2$ (top) and $r = 4$ (bottom)

d	M	TR	NC4	CC
100	1.354e+06	2.087e-12	4.615e-13	5.099e-13
300	3.618e+07	4.758e-41	1.291e-41	1.326e-41
500	1.672e+08	1.762e-70	4.957e-71	5.034e-71
700	4.583e+08	3.837e-100	1.096e-100	1.108e-100

Table 3: L_2-discrepancies of Smolyak quadratures for $r = 0$ in high dimensions.

dimensional sequence (Q_n), we are now able to compute discrepancies of point sets up to $M \approx 10^{40}$ points. We think that this might be of interest for experiments accompanying theoretical investigations. Furthermore, since the main effort is enclosed in the first dimension, whereas the cost of each step in the recursion (12) is very small, the calculation of the discrepancy of a high-dimensional Smolyak quadrature represents no difficulty, independently of the sequence (Q_n). However, for some parameter constellations the precision of the calculations turned out to be a limiting factor due to cancellation in (6).

For moderate dimension and numbers of nodes the algorithm could be employed to optimize Smolyak quadratures. Finally, the generality of section 4 allows many other reproducing kernels to be used as quality measures. This might be reasonable since discrepancies based on $B_r^{(d)}(x,t) = \frac{1}{(r!)^d}(t-x)_+^r$ $(r \geq 0)$ obviously tend to overemphasize the role of the point $(0,...,0)$. With new kernels, however, efficient algorithms for dimension one become an issue again.

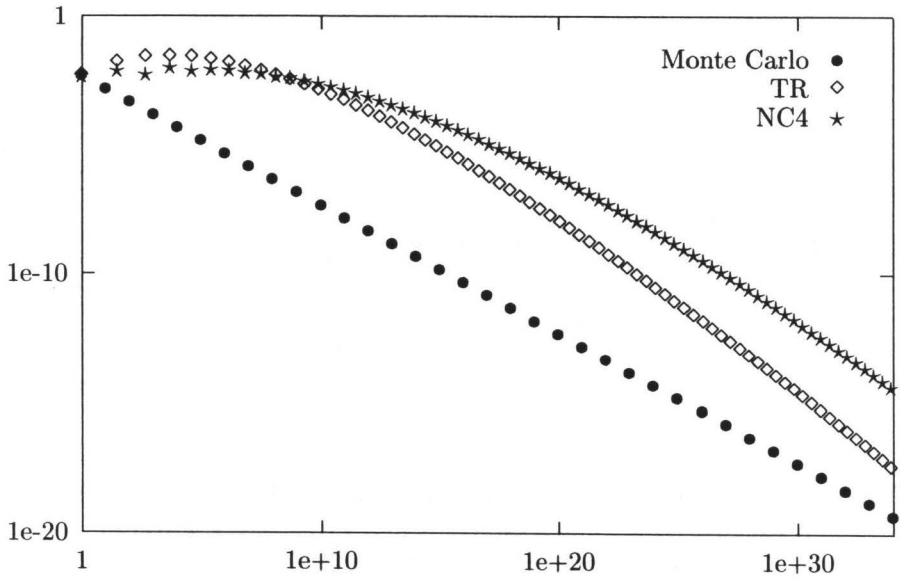

Figure 4: L_2-discrepancies in dimension $d = 15$ for large point sets $(r = 0)$

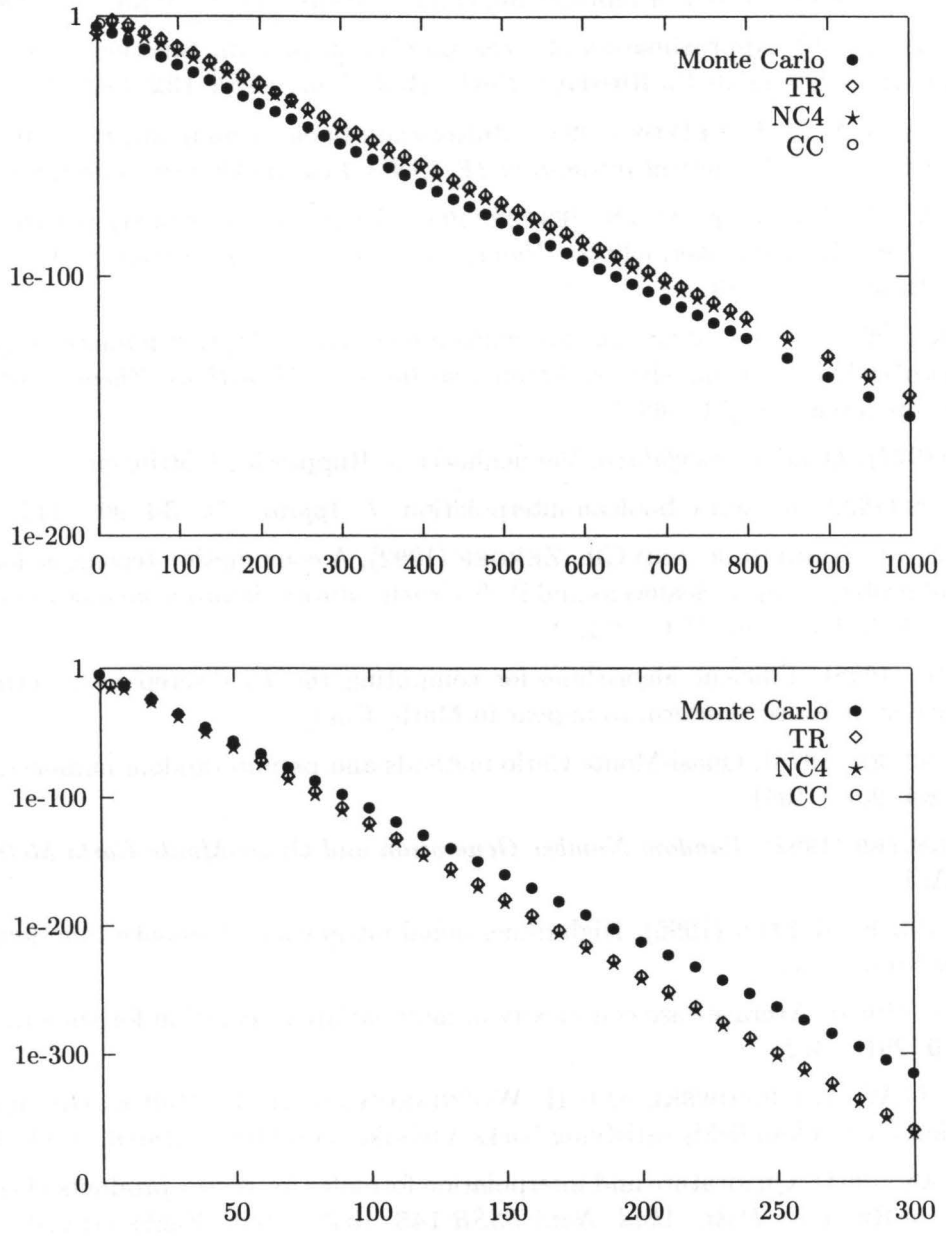

Figure 5: L_2-discrepancies in high dimensions for quadratures with about 10^6 points for smoothness $r = 0$ (top) and $r = 2$ (bottom)

References

M. ABRAMOWITZ AND I. A. STEGUN (1972), *Handbook of mathematical functions*, National Bureau of Standards, Washington D. C..

N. ARONSZAJN (1950), Theory of reproducing kernels, *Trans. Amer. Math. Soc.* **68**, 337 – 404.

K.I. BABENKO (1960), Approximation of a certain class of periodic functions of several variables by trigonometric polynomials (in Russian), *Dokl. Akad. Nauk SSSR* **132**, 982 – 985.

G. BASZENSKI AND F.-J. DELVOS (1993), Multivariate boolean midpoint rules, in H. Brass and G. Hämmerlin, editors, *Numerical integration IV*, ISNM **112**, Birkhäuser, Basel, pp. 1 – 11.

G. BASZENSKI, F.-J. DELVOS, AND S. JESTER (1992), Blending approximation with sine functions, in D. Braess and L.L. Schumaker, editors, *Numerical methods of approximation, Vol. 9*, ISNM **105**, Birkhäuser, Basel, pp. 1 – 19.

TH. BONK (1994), A new algorithm for multi-dimensional adaptive numerical quadrature, in W. Hackbusch and G. Wittum, editors, *Adaptive methods — Algorithms, Theory and Applications*, Vieweg, Braunschweig, pp. 54 – 68.

H. BRASS (1977), *Quadraturverfahren*, Vandenhoeck & Rupprecht, Göttingen.

F.-J. DELVOS (1982), d-variate boolean interpolation, *J. Approx. Th.* **34**, 99 – 114.

M. GRIEBEL, M. SCHNEIDER, AND CH. ZENGER (1992), A combination technique for the solution of sparse grid problems, In R. Beauwens and P. de Groen, editors, *Iterative methods in linear algebra*, Elsevier, North–Holland, pp. 263 – 281.

S. HEINRICH (1995), Efficient algorithms for computing the L_2 discrepancy, Technical Report 267/95, Universität Kaiserslautern, to appear in Math. Comp.

H. NIEDERREITER (1978), Quasi-Monte Carlo methods and pseudo-random numbers, *Bull. Amer. Math. Soc.* **84**, 957 – 1041.

H. NIEDERREITER (1992), *Random Number Generation and Quasi-Monte Carlo Methods*, CBMS–NSF 63, SIAM.

E. NOVAK AND K. RITTER (1996), High dimensional integration of smooth functions over cubes, to appear in Num. Math..

S.H. PASKOV (1993), Average case complexity of multivariate integration for smooth functions, *J. Complexity* **9**, 291 – 312.

K. RITTER, G.W. WASILKOWSKI, AND H. WOŹNIAKOWSKI (1995), Multivariate integration and approximation for random fields satisfying Sacks-Ylvisaker conditions, *Annal. Prob.* **5**, 518 – 540.

S.A. SMOLYAK (1963), Quadrature and interpolation formulas for tensor products of certain classes of functions (in Russian), *Dokl. Akad. Nauk SSSR* **148**, 1042 – 1045. English transl.: *Soviet. Mat. Dokl.* **4**, 240 - 243.

J. SACKS AND D. YLVISAKER (1970), Statistical design and integral approximation, In R. Pyke, editor, *Proc. 12th. Bienn. Semin. Can. Math. Congr.*, Can. Math. Soc., Montreal, pp. 115 – 136.

V.N. TEMLYAKOV (1990), On a way of obtaining lower estimates for the errors of quadrature formulas, *Mat. Sbornik* **181**, 1403 – 1413. English transl.: *Math. USSR Sbornik* **71**(1992), 247 –257.

V.N. TEMLYAKOV (1993), On approximate recovery of functions with bounded mixed derivative, *J. Complexity* **9**, 41 – 59.

S. TEZUKA (1995), *Uniform Random Numbers: Theory and Practice*, Kluwer Academic Publishers.

T.T. WARNOCK (1972), Computational investigations of low discrepancy point sets, In S.K. Zaremba, editor, *Applications of Number Theory to Numerical Analysis*, Academic Press, New York.

D. WERNER (1995), *Funktionalanalysis*, Springer, Berlin.

H. WOŹNIAKOWSKI (1991), Average case complexity of multivariate integration, *Bull. Amer. Math. Soc.* **24**, 185 – 194.

G.W. WASILKOWSKI AND H. WOŹNIAKOWSKI (1995), Explicit cost bounds of algorithms for multivariate tensor product problems, *J. Complexity* **11**, 1 – 56.

S.K. ZAREMBA (1968), Some applications of multidimensional integration by parts, *Ann. Polon. Math.* **21**, 85 – 96.

CH. ZENGER (1991), Sparse grids, In W. Hackbusch, editor, *Parallel algorithms for partial differential equations*, Vieweg, Braunschweig, pp. 241 – 251.

J. ZIELINSKI (1994), Multidimensionale Integration mittels Tensorprodukt-Technik, Report Universität Kaiserslautern.