Lattice rules for multivariate integration

Ronald Cools
Dept. of Computer Science, K.U.Leuven

Woudschoten 10 October 2008

Introduction

Given is an integral

$$I[f] := \int_\Omega w(\mathbf{x}) f(\mathbf{x}) \, \mathrm{d}\mathbf{x}$$

where $\Omega \subseteq \mathbb{R}^s$ and $w(\mathbf{x}) \geq 0$, $\forall \mathbf{x} \in \mathbb{R}^s$.
Search an approximation for $I[f]$

$$I[f] \simeq Q[f] := \sum_{j=1}^{n} w_j f(\mathbf{y}^{(j)})$$

with $w_j \in \mathbb{R}$ and $\mathbf{y}^{(j)} \in \mathbb{R}^s$.
Webster:
quadrature: the process of finding a square equal in area to a given area.
cubature: the determination of cubic contents.
If $s = 1$ then Q is called a quadrature formula.
If $s \geq 2$ then Q is called a cubature formula.
Note: in qMC one usually uses s to denote the dimension.

$$Q[f] := \sum_{j=1}^{n} w_j f(\mathbf{y}^{(j)})$$

Cubature/quadrature formulas are basic integration rules
$\rightarrow$ choose points $\mathbf{y}^{(j)}$ and weights w_j independent of integrand f.

One used to say:
It is difficult (time consuming) to construct basic integration rules, but the result is usually hard coded in programs or tables.
This argument is nowadays not always valid!

$$Q[f] := \sum_{j=1}^{n} w_j f(\mathbf{y}^{(j)})$$

Cubature/quadrature formulas are basic integration rules
$\rightarrow$ choose points $\mathbf{y}^{(j)}$ and weights w_j independent of integrand f.

One used to say:
It is difficult (time consuming) to construct basic integration rules, but the result is usually hard coded in programs or tables.
This argument is nowadays not always valid!

Restriction to unit cube: given is

$$I[f] = \int_0^1 \cdots \int_0^1 f(x_1, \ldots, x_s) \mathrm{d}x_1 \cdots \mathrm{d}x_s = \int_{[0,1)^s} f(\mathbf{x}) \mathrm{d}\mathbf{x}$$

Monte Carlo methods

(S. Ulam, N. Metropolis & J. von Neumann, 1945)

$$Q[f] := \frac{1}{N}\sum_{j=1}^{N} f(y^{(j)}) \text{ with } \mathrm{Vol}(\Omega) = 1$$

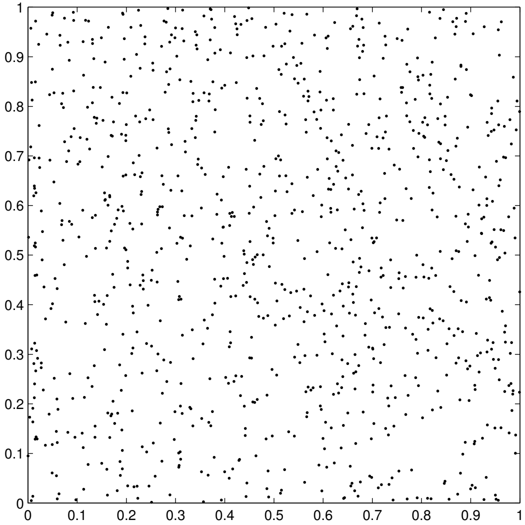
Random points (according to pdf $w(x)$) $y^{(j)} \in \Omega$.
For any given $\varepsilon > 0$

$$\lim_{N\to\infty} \mathrm{prob}\left(I[f] - \varepsilon \le Q[f] := \frac{1}{N}\sum_{j=1}^{N} f(y^{(j)}) \le I[f] + \varepsilon \right) = 1.$$

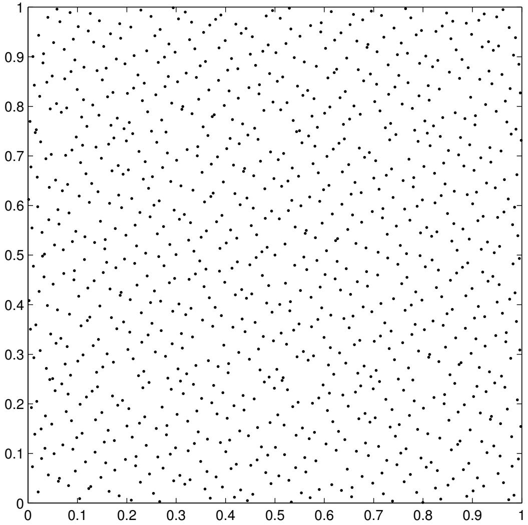
Central limit theorem $\to$

$$\lim_{N\to\infty} \mathrm{prob}\left(\left| \frac{1}{N}\sum_{j=1}^{N} f(y^{(j)}) - I[f] \right| \le \frac{\lambda\sigma}{\sqrt{N}} \right) = \frac{1}{\sqrt{2\pi}} \int_{-\lambda}^{\lambda} e^{-t^2/2} \mathrm{d}t.$$

Random points appear in clusters with blind spots in between



The same number of points, more pleasingly distributed



Quasi-Monte Carlo methods (R.D. Richtmyer, 1952)

Points $y^{(j)}$ are chosen "**better than random**"
$\longrightarrow$ based on **low-discrepancy sequences** (open methods)
(Richtmyer, Sobol, Halton, Niederreiter, Faure,... generalisations)

The following theorem is due to Koksma (1942/3) for $s = 1$ and has been generalised to multivariate functions by Hlawka (1961).

Theorem

If f has bounded variation $V(f)$ on $[0,1]^s$ in the sense of Hardy and Krause, then, for any $y^{(1)}, \ldots, y^{(N)} \in [0,1]^s$, we have

$$\left| \frac{1}{N} \sum_{j=1}^{N} f(y^{(j)}) - \int_{[0,1]^s} f(x)\mathrm{d}x \right| \leq V[f] D_N^\star(y^{(1)}, \ldots, y^{(N)}).$$

$$D_N^\star \asymp \frac{(\log N)^s}{N}$$

Quasi-Monte Carlo methods (R.D. Richtmyer, 1952)

Points $y^{(j)}$ are chosen "**better than random**"
$\longrightarrow$ based on low-discrepancy sequences (open methods)
(Richtmyer, Sobol, Halton, Niederreiter, Faure,... generalisations)

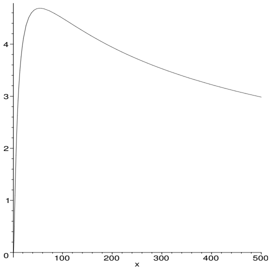
Theorem

If f has bounded variation $V(f)$ on $[0,1]^s$ in the sense of Hardy and Krause, then, for any $y^{(1)}, \dots, y^{(N)} \in [0,1]^s$, we have

$$\left| \frac{1}{N} \sum_{j=1}^{N} f(y^{(j)}) - \int_{[0,1]^s} f(x) \mathrm{d}x \right| \leq V[f] D_N^\star(y^{(1)}, \dots, y^{(N)}).$$

$\longrightarrow$ **lattice rules** (closed methods)
so-called "good lattice points" (Korobov, 1959)
more general (Frolov, 1977), (Sloan & Kachoyan, 1987)

Quasi-Monte Carlo loses its effectiveness when s is large.



This can be anticipated from the discrepancy bound $\frac{(\log N)^s}{N}$

For large s this bound is dominated by $(\log N)^s$ unless $N > e^s$.

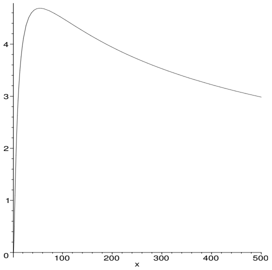
E.g.: $d = 30 \Rightarrow N > 10^{13}$
$\Rightarrow$ Not applicable for high dimensions....

However,...

Quasi-Monte Carlo loses its effectiveness when s is large.

This can be anticipated from the discrepancy bound $\frac{(\log N)^s}{N}$

For large s this bound is dominated by $(\log N)^s$ unless $N > e^s$.



E.g.: $d = 30 \Rightarrow N > 10^{13}$
$\Rightarrow$ Not applicable for high dimensions....

However,...
it works perfect for financial applications with $s = 360$ and more!

Discovered by accident by Paskov ($\pm$ 1994)
and the starting point for a lot of research in the area.

Quasi-random number generators

There are various notions of *discrepancy* to measure the irregularity of a point set (finite or infinite).

In other words, discrepancy is a measure to quantify the deviation from a uniform distribution.

An infinite sequence $\{x_i\}$ is *uniformly distributed* if

$$\lim_{N \to \infty} D_N = 0.$$

The sequence is *quasi-random* if

$$D_N \leq c(\log N)^k N^{-1}$$

where c and k are constants independent of N.
It is possible to construct sequences with $k = s$.

Basic ingredient for constructions:

Let r be a fixed integer > 1.
Then any positive integer k can be written in the form

$$k = c_0 + c_1 r + \cdots + c_q r^q, \quad 0 \le c_i < r.$$

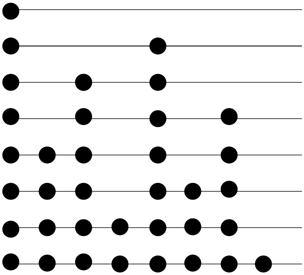
The radical-inverse function $\phi_r(k)$ is then given by

$$\phi_r(k) = c_0 r^{-1} + c_1 r^{-2} + \cdots + c_q r^{-q-1}.$$

Let $p(j)$ be the jth prime so that $p(1) = 2, p(2) = 3, \ldots$.

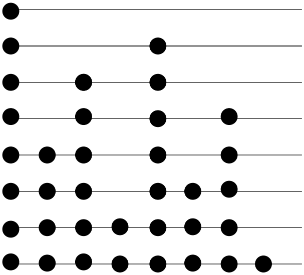
In one dimension, the sequence $\{\phi_2(i) : i = 1, 2, \ldots\}$ is known as the van der Corput sequence.

i		$\phi_2(i)$	
0	0	0.0	0.0
1	1	0.1	0.5
2	10	0.01	0.25
3	11	0.11	0.75
4	100	0.001	0.125
5	101	0.101	0.625
6	110	0.011	0.375
7	111	0.111	0.875



In one dimension, the sequence $\{\phi_2(i) : i = 1, 2, \ldots\}$ is known as the van der Corput sequence.

i		$\phi_2(i)$	
0	0	0.0	0.0
1	1	0.1	0.5
2	10	0.01	0.25
3	11	0.11	0.75
4	100	0.001	0.125
5	101	0.101	0.625
6	110	0.011	0.375
7	111	0.111	0.875



Then the Halton sequence (1960) in s dimensions is given by $\{x_i\}$ where

$$x_i = (\phi_{p(1)}(i), \phi_{p(2)}(i), \ldots, \phi_{p(s)}(i)), \quad i = 1, 2, \ldots$$

Other important sequences are due to Sobol' (1967), Faure (1982), Niederreiter (1992), Niederreiter-Xing (1995).
Niederreiter developed a general theory of (t, s) sequences (nets), which contains all these sequences.

Other important sequences are due to Sobol' (1967), Faure (1982), Niederreiter (1992), Niederreiter-Xing (1995).
Niederreiter developed a general theory of (t, s) sequences (nets), which contains all these sequences.

Implementations of sequences (for moderate s):
Bratley & Fox (1988) $\longrightarrow$ Sobol' ($s \leq 40$)
Press et al. (1992) $\longrightarrow$ Sobol'
Bratley, Fox & Niederreiter (1994) $\longrightarrow$ Niederreiter

Other important sequences are due to Sobol' (1967), Faure (1982), Niederreiter (1992), Niederreiter-Xing (1995).
Niederreiter developed a general theory of (t, s) sequences (nets), which contains all these sequences.

Implementations of sequences (for moderate s):
Bratley & Fox (1988) $\rightarrow$ Sobol' ($s \leq 40$)
Press et al. (1992) $\rightarrow$ Sobol'
Bratley, Fox & Niederreiter (1994) $\rightarrow$ Niederreiter

Implementations of sequences (for $s > 300$):
Tezuka $\rightarrow$ generalised Faure (patents by IBM)
Traub et al. $\rightarrow$ generalised Faure (patents by Columbia Univ., NY)
Kuo & Joe (2003) $\rightarrow$ Sobol' (for $s \leq 1111$)
(Recently added to Matlab.)

If you don't plan to take notes

- A good introduction to lattice rules is Sloan & Joe (1994).
- There are no quadrature/cubature specific journals.

  $\Downarrow$

  Relevant results are scattered around in many journal publications.
- A recent survey is (free sample pages!)
  C. & Nuyens (2008) "A Belgian View on Lattice Rules".

Springer
ISBN 978-3-540-74495-5
Ed. Keller, Heinrich & Niederreiter

A quote from
"A Belgian View on Lattice Rules":

7. Tourist Information

It is beyond any doubt that the biggest monument in the world devoted to a lattice is the Atomium in Brussels, Belgium. This monument was designed for the Brussels World's Fair that took place in 1958 (Expo '58). The Atomium consist of 9 balls symbolizing a unit cell of the body centered cubic lattice crystal structure of iron magnified 165×10^9 *times.*

Taxonomy: two major classes

1. polynomial based methods
   incl. methods exact for algebraic or trigonometric polynomials
2. number theoretic methods
   incl. Monte Carlo and quasi-Monte Carlo methods

As in zoology, some species are difficult to classify.

Taxonomy: two major classes

1. polynomial based methods
   incl. methods exact for algebraic or trigonometric polynomials
2. number theoretic methods
   incl. Monte Carlo and quasi-Monte Carlo methods

As in zoology, some species are difficult to classify.
For example

Definition

An s-dimensional lattice rule is a cubature formula which can be expressed in the form

$$Q[f] = \frac{1}{d_1 d_2 \dots d_t} \sum_{j_1=1}^{d_1} \sum_{j_2=1}^{d_2} \dots \sum_{j_t=1}^{d_t} f\left(\left\{\frac{j_1 \mathbf{z}_1}{d_1} + \frac{j_2 \mathbf{z}_2}{d_2} + \dots + \frac{j_t \mathbf{z}_t}{d_t}\right\}\right),$$

where $t < s$, $d_i \in \mathbb{N}_0$ and $\mathbf{z}_i \in \mathbb{Z}^s$ for all i.

Alternative formulation:

Definition

A multiple integration lattice Λ is a subset of $\mathbb{R}^s$ which is discrete and closed under addition and subtraction and which contains $\mathbb{Z}^s$ as a subset.

Definition

A lattice rule is a cubature formula where the n points are the points of a multiple integration lattice Λ that lie in $[0,1)^s$ and the weights are all equal to $1/n$.

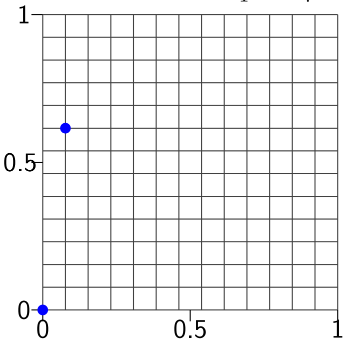
$$n = n(Q) = \#\{\Lambda \cap [0,1)^s\} \ .$$

Example

The Fibonnaci lattice with $n = F_j$ and $\mathbf{z} = (1, F_{j-1})$ has points $\mathbf{x}^{(j)} = \left(\frac{j}{F_j}, \frac{jF_{j-1}}{F_j}\right)$

$$\Rightarrow \text{lattice rule } Q[f] = \frac{1}{n}\sum_{j=0}^{n-1} f\left(\left\{\frac{(j, jF_{j-1})}{n}\right\}\right)$$

Example: the lattice rule with $n = d_1 = F_7 = 13$ and $\mathbf{z}_1 = (1, 8)$

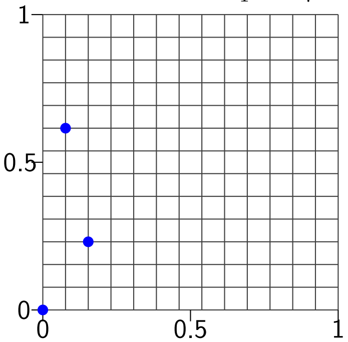


Example

The Fibonnaci lattice with $n = F_j$ and $\mathbf{z} = (1, F_{j-1})$ has points $\mathbf{x}^{(j)} = \left(\frac{j}{F_j}, \frac{jF_{j-1}}{F_j}\right)$

$$\Rightarrow \text{lattice rule } Q[f] = \frac{1}{n} \sum_{j=0}^{n-1} f\left(\left\{\frac{(j, jF_{j-1})}{n}\right\}\right)$$

Example: the lattice rule with $n = d_1 = F_7 = 13$ and $\mathbf{z}_1 = (1, 8)$

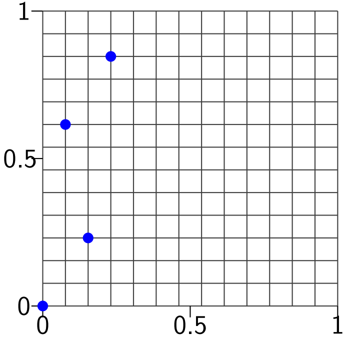


Example

The Fibonnaci lattice with $n = F_j$ and $\mathbf{z} = (1, F_{j-1})$
has points $\mathbf{x}^{(j)} = \left(\frac{j}{F_j}, \frac{jF_{j-1}}{F_j}\right)$

$$\Rightarrow \text{lattice rule } Q[f] = \frac{1}{n}\sum_{j=0}^{n-1} f\left(\left\{\frac{(j, jF_{j-1})}{n}\right\}\right)$$

Example: the lattice rule with $n = d_1 = F_7 = 13$ and $\mathbf{z}_1 = (1, 8)$

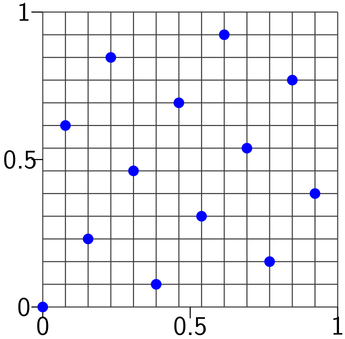


Example

The Fibonnaci lattice with $n = F_j$ and $\mathbf{z} = (1, F_{j-1})$ has points $\mathbf{x}^{(j)} = \left(\frac{j}{F_j}, \frac{jF_{j-1}}{F_j}\right)$

$$\Rightarrow \text{lattice rule } Q[f] = \frac{1}{n}\sum_{j=0}^{n-1} f\left(\left\{\frac{(j, jF_{j-1})}{n}\right\}\right)$$

Example: the lattice rule with $n = d_1 = F_7 = 13$ and $\mathbf{z}_1 = (1, 8)$



Warning!

The previous figure might give the impression that a lattice rule is a rotated grid.
That is not correct!

Some differences:

1. Lattice rules can be constructed $\forall n \in \mathbb{N}$ (prime n are most popular).
   Grids use $n = N^s$ points $\longrightarrow$ not done in high dimensions.
2. The 1-dimensional projection of a lattice rule is (usually) a trapezoidal rule with n points.
   For a grid that is a trapezoidal rule with only $n^{1/s}$ points.
   (So the order of accuracy is determined by $N = n^{1/s}$.)

The same lattice rule Q may have many different representations of the form

$$\frac{1}{d_1 d_2 \dots d_t} \sum_{j_1=1}^{d_1} \sum_{j_2=1}^{d_2} \dots \sum_{j_t=1}^{d_t} f\left(\left\{\frac{j_1 \mathbf{z}_1}{d_1} + \frac{j_2 \mathbf{z}_2}{d_2} + \dots + \frac{j_t \mathbf{z}_t}{d_t}\right\}\right),$$

using different values of t and other parameters.

Definition

A lattice rule is of rank r if it can be expressed in this form with $t = r$, but cannot be so expressed with $t < r$.

The same lattice rule Q may have many different representations of the form

$$\frac{1}{d_1 d_2 \dots d_t} \sum_{j_1=1}^{d_1} \sum_{j_2=1}^{d_2} \cdots \sum_{j_t=1}^{d_t} f\left(\left\{\frac{j_1 \mathbf{z}_1}{d_1} + \frac{j_2 \mathbf{z}_2}{d_2} + \ldots + \frac{j_t \mathbf{z}_t}{d_t}\right\}\right),$$

using different values of t and other parameters.

Definition

A lattice rule is of rank r if it can be expressed in this form with $t = r$, but cannot be so expressed with $t < r$.

Examples

rank-1 simple:

$$Q[f] = \frac{1}{3544} \sum_{j=1}^{3544} f\left(\{j\mathbf{z}/N\}\right) \quad \text{with} \quad \mathbf{z} = (1, 17, 129, 985)$$

The same lattice rule Q may have many different representations of the form

$$\frac{1}{d_1 d_2 \ldots d_t} \sum_{j_1=1}^{d_1} \sum_{j_2=1}^{d_2} \cdots \sum_{j_t=1}^{d_t} f\left(\left\{\frac{j_1 \mathbf{z}_1}{d_1} + \frac{j_2 \mathbf{z}_2}{d_2} + \ldots + \frac{j_t \mathbf{z}_t}{d_t}\right\}\right),$$

using different values of t and other parameters.

Definition

A lattice rule is of rank r if it can be expressed in this form with $t = r$, but cannot be so expressed with $t < r$.

Examples

rank-2

$$Q[f] = \frac{1}{3312} \sum_{j=1}^{1656} \sum_{k=1}^{2} f\left(\left\{\frac{j\mathbf{z}_2}{1656} + \frac{k\mathbf{e}_1}{2}\right\}\right)$$

with

$$\mathbf{z}_2 = (1431, 919, 495, 1);\ \mathbf{e}_1 = (1, 0, 0, 0).$$

Polynomials

Let $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_s) \in \mathbb{Z}^s$ and $|\alpha| := \sum_{j=1}^{s} |\alpha_j|$.

algebraic polynomial

$$p(\mathbf{x}) = \sum a_\alpha \mathbf{x}^\alpha = \sum a_\alpha \prod_{j=1}^{s} x_j^{\alpha_j}, \quad \text{with } \alpha_j \geq 0$$

trigonometric polynomial

$$t(\mathbf{x}) = \sum a_\alpha e^{2\pi i \alpha \cdot \mathbf{x}} = \sum a_\alpha \prod_{j=1}^{s} e^{2\pi i x_j \alpha_j}$$

The degree of a polynomial $= \max_{a_\alpha \neq 0} |\alpha|$.

$\mathbb{P}_d^s$ = all algebraic polynomials in s variables of degree at most d.

$\mathbb{T}_d^s$ = all trigonometric polynomials in s variables of degree at most d.

Quality criteria?

Definition

A cubature formula Q for an integral I has algebraic (trigonometric) degree d if it is exact for all polynomials of algebraic (trigonometric) degree at most d.

Quality criteria?

Definition

A cubature formula Q for an integral I has algebraic (trigonometric) degree d if it is exact for all polynomials of algebraic (trigonometric) degree at most d.

How many points are needed in a cubature formula to obtain a specified degree of precision?

The dimensions of the vector spaces of polynomials are:

$$\dim \mathbb{P}_d^s = \left(\begin{array}{c} s+d \\ d \end{array} \right)$$

$$\dim \mathbb{T}_d^s = \sum_{j=0}^{s} \left(\begin{array}{c} s \\ j \end{array} \right) \left(\begin{array}{c} d \\ j \end{array} \right) 2^j.$$

We will use the symbol $\mathbb{V}_d^s$ to refer to one of the vector spaces $\mathbb{P}_d^s$ or $\mathbb{T}_d^s$.

Theorem

If a cubature formula is exact for all polynomials of $\mathbb{V}_{2k}^s$, then the number of points $n \geq \dim \mathbb{V}_k^s$.

Algebraic degree: For $s = 2$ (Radon, 1948); general s (Stroud, 1960)
Trigonometric degree: (Mysovskikh, 1987)

J. Radon

A. Stroud

И.П. Мысовских

Theorem

If a cubature formula is exact for all polynomials of degree $d > 0$ and has only real points and weights, then it has at least $\dim \mathbb{V}_k^s$ positive weights, $k = \lfloor \frac{d}{2} \rfloor$.

Algebraic degree: (Mysovskikh, 1981)
Trigonometric degree: (C. 1997)
$\Rightarrow$ minimal formulas have only positive weights.

Corollary

If a cubature formula of trigonometric degree $2k$ has $n = \dim \mathbb{T}_k^s$ points, then all weights are equal.

This is a reason to restrict searches to

$$Q[f] = \frac{1}{n} \sum_{j=1}^{n} f(\mathbf{x}_j).$$

Improved bound for odd degrees

For algebraic degree, the improved lower bound for odd degrees takes into account the symmetry of the integration region.

E.g., centrally symmetric regions such as a cube
$\longrightarrow$ (Möller, 1973)

H.M. Möller

Result for trigonometric degree is very similar.

Improved bound for odd degrees

$G_k :=$ span of trigonometric monomials of degree $\leq k$ with the same parity as k.

Theorem ((Noskov, 1985), (Mysovskikh, 1987))

The number of points n of a cubature formula for the integral over $[0,1)^s$ which is exact for all trigonometric polynomials of degree at most $d = 2k + 1$ satisfies

$$n \geq 2 \dim G_k.$$

Definition

A cubature formula is called shift symmetric if it is invariant w.r.t. the group of transformations

$$\left\{\mathbf{x} \mapsto \mathbf{x}, \mathbf{x} \mapsto \{\mathbf{x} + (\frac{1}{2}, \dots, \frac{1}{2})\}\right\}$$

(*This is the 'central symmetry' for the trig. case.*)

Theorem (Beckers & C., 1993)

If a shift symmetric cubature formula of degree $2k+1$ has $n = 2\dim G_k$ points, then all weights are equal.

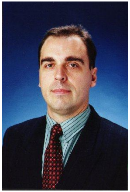
Theorem (Osipov, 2001), Conjectured (C., 1997)

Any cubature formula that attains the lower bound is shift symmetric.

Known minimal formulas for trigonometric degree

- for all s
  - degree 1
  - degree 2 (Noskov, 1988)
  - degree 3 (Noskov, 1988)
- for $s = 2$
  - all even degrees (Noskov, 1988)
  - all odd degrees
    (Reztsov, 1990) (Beckers & C., 1993) (C. & Sloan, 1996)
- for $s = 3$
  - degree 5 (Frolov, 1977)

M.B. Носков



A. Резцов

I.H. Sloan

All known minimal formulas of trigonometric degree are lattice rules, except...

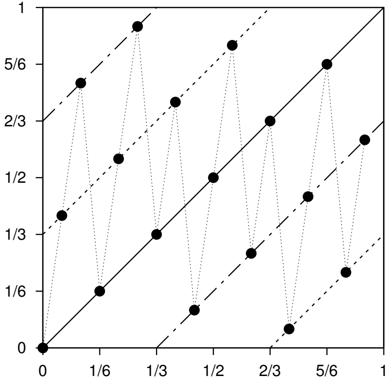
All known minimal formulas of trigonometric degree are lattice rules, except...

Theorem (C. & Sloan, 1996)

The following points

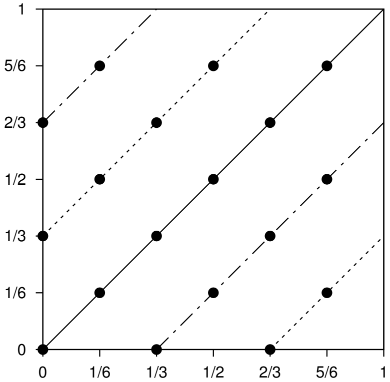
$$\left(C_p + \frac{j}{2(k+1)}, C_p + \frac{j+2p}{2(k+1)}\right) \quad \begin{array}{rcl} j & = & 0,\ldots,2k+1 \\ p & = & 0,\ldots,k \end{array}$$

with $C_0 = 0$ and $C_1, \ldots, C_k$ arbitrary are the points of a minimal cubature formula of trigonometric degree $2k+1$.



$k = 2$, $n = 18$, $C_1 = \frac{1}{18}$, $C_2 = \frac{1}{9}$

$$Q[f] = \frac{1}{n} \sum_{j=0}^{n-1} f\left(\frac{j}{n}, \frac{j(2m+1)}{n}\right) \quad \text{with} \quad n = 2(m+1)^2$$



$k = 2$, $n = 18$, $C_1 = C_2 = 0$: body-centered cubic lattice

$$Q[f] = \frac{1}{2(m+1)^2} \sum_{k=0}^{2m+1} \sum_{j=0}^{m} f\left(\frac{2j+k}{2(m+1)}, \frac{k}{2(m+1)}\right) \quad \text{with} \quad n = 2(m+1)^2$$

Technology used to obtain these results: Reproducing kernels

The integral I defines an inner product $(\phi, \psi) = I[\overline{\phi} \cdot \psi]$.
Let $\mathbb{F}$ be a subspace of $\mathbb{T}^s$.
Choose $\phi_1(\mathbf{x}), \phi_2(\mathbf{x}), \ldots \in \mathbb{F}$ so that

- $\phi_i(\mathbf{x})$ is I-orthogonal to $\phi_j(\mathbf{x})$, $\forall j < i$, and
- $(\phi_i(\mathbf{x}), \phi_i(\mathbf{x})) = 1$.

For a given $k \in \mathbb{N}$ and $t := \dim(\mathbb{F} \cap \mathbb{T}^s_k)$ we define

$$K(\mathbf{x}, \mathbf{y}) := \sum_{j=1}^{t} \overline{\phi_j(\mathbf{x})} \cdot \phi_j(\mathbf{y})$$

$K(\mathbf{x}, \mathbf{y})$ is a polynomial in $2s$ variables of degree $\leq 2k$.

Definition

K is a reproducing kernel in the space $\mathbb{F} \cap \mathbb{T}_k^s$

$$\text{if } \; f \in \mathbb{F} \cap \mathbb{T}_k^s \; \text{ then } \; f(\mathbf{a}) = (f(\mathbf{x}), K(\mathbf{x}, \mathbf{a})) = \sum_{j=1}^{t} \phi_j(\mathbf{a}) \cdot I[f(\mathbf{x})\overline{\phi_j}(\mathbf{x})]$$

The trigonometric monomials form an orthonormal sequence.

$$K(\mathbf{x}, \mathbf{y}) = \sum_{\mathbf{k} \in \Lambda_d} e^{2\pi i \mathbf{k} \cdot (\mathbf{x} - \mathbf{y})}$$

$$\Lambda_d = \{\mathbf{k} \in \mathbb{Z}^s : 0 \leq \sum_{l=1}^{s} |k_l| \leq \lfloor \frac{d}{2} \rfloor\}$$

A simplifying aspect of the trigonometric case is that the reproducing kernel is a function of one variable:

$$K(\mathbf{x}, \mathbf{y}) = \mathcal{K}(\mathbf{x} - \mathbf{y})$$

with

$$\mathcal{K}(\mathbf{x}') = \sum_{\mathbf{k} \in \Lambda_d} e^{2\pi i \mathbf{k} \cdot \mathbf{x}'}$$

For $s = 2$ it has the following simple form:
let $g(z) = \cos(\pi(2\lfloor \frac{d}{2} \rfloor + 1)z)\ \cos \pi z$, then

$$\mathcal{K}(\mathbf{x}') = \frac{g(x_1) - g(x_2)}{\sin(\pi(x_1 + x_2)) \sin(\pi(x_1 - x_2))}.$$

On route to other quality criteria

Assume f can be expanded into
an absolutely convergent multiple Fourier series

$$f(\mathbf{x}) = \sum_{\mathbf{h} \in \mathbb{Z}^s} \hat{f}(\mathbf{h}) e^{2\pi i \mathbf{h} \cdot \mathbf{x}}$$

with

$$\hat{f}(\mathbf{h}) = \int_{[0,1)^s} f(\mathbf{x}) e^{-2\pi i \mathbf{h} \cdot \mathbf{x}} \, \mathrm{d}\mathbf{x}$$

Then

$$\begin{aligned} Q[f] - I[f] &= \frac{1}{n}\sum_{j=1}^{n}\left(\sum_{\mathbf{h}\in\mathbb{Z}^s\setminus\{0\}} \hat{f}(\mathbf{h}) e^{2\pi i \mathbf{h}\cdot\mathbf{x}_j}\right) \\ &= \sum_{\mathbf{h}\in\mathbb{Z}^s\setminus\{0\}} \left(\hat{f}(\mathbf{h})\frac{1}{n}\sum_{j=1}^{n} e^{2\pi i \mathbf{h}\cdot\mathbf{x}_j}\right). \end{aligned}$$

Observe that

$$\frac{1}{n}\sum_{j=1}^{n} e^{2\pi i \mathbf{h}\cdot\mathbf{x}_j} = \begin{cases} 1, & \mathbf{h}\cdot\mathbf{x}_j \in \mathbb{Z} \\ 0, & \mathbf{h}\cdot\mathbf{x}_j \notin \mathbb{Z} \end{cases}$$

A very important tool to investigate the error of a lattice rule is . . .

Definition

The dual of the multiple integration lattice Λ

$$\Lambda^\perp := \{\mathbf{h} \in \mathbb{Z}^s \ : \mathbf{h} \cdot \mathbf{x} \in \mathbb{Z}, \forall \mathbf{x} \in \Lambda\}.$$

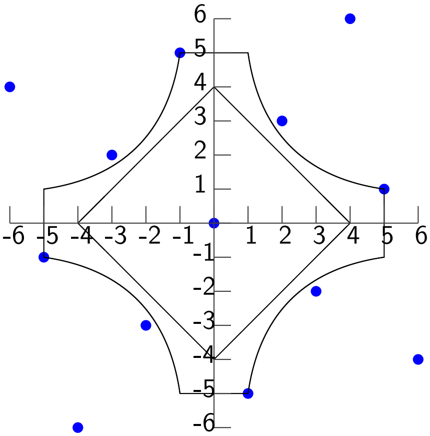
Theorem (Sloan & Kachoyan, 1987)

Let Λ be a multiple integration lattice.
Then the corresponding lattice rule Q has an error

$$Q[f] - I[f] = \sum_{\mathbf{h} \in \Lambda^\perp \setminus \{0\}} \hat{f}(\mathbf{h}).$$

Example

Dual lattice of Fibonnaci lattice



Construction criteria

For many years, only used in Russia...

Definition

The trigonometric degree is

$$d(Q) := \min_{\substack{\mathbf{h} \neq \mathbf{0} \\ \mathbf{h} \in \Lambda^\perp}} \left(\sum_{j=1}^{s} |h_j| \right) - 1 \, .$$

The enhanced degree $\delta := d + 1$.

Some names:
Mysovskikh (1985–1990), Reztsov (1990), Noskov (1985–1988), Temirgaliev (1991), Semenova (1996–1997)

Construction criteria

Mainly used in the 'West'...

Definition

The Zaremba index or figure of merit is

$$\rho(Q) := \min_{\substack{\mathbf{h} \neq \mathbf{0} \\ \mathbf{h} \in \Lambda^\perp}} \left(\bar{h}_1 \bar{h}_2 \cdots \bar{h}_s\right) .$$

with

$$\bar{h}_j := \begin{cases} 1 & \text{if } h_j = 0 \\ |h_j| & \text{if } h_j \neq 0. \end{cases}$$

Some names:
Maisonneuve (1972), . . ., Sloan & Joe (1994), Langtry (1996)

Where does this come from?

For $c > 0$ and fixed $\alpha > 1$,
let $E_s^\alpha(c)$ be the class of functions f
whose Fourier coefficients satisfy

$$|\hat{f}(\mathbf{h})| \leq \frac{c}{(\overline{h}_1 \overline{h}_2 \cdots \overline{h}_s)^\alpha},$$

where $\overline{h} = \max(1, |h|)$.

Worst possible function in class $E_s^\alpha(1)$ is

$$f_\alpha := \sum_{\mathbf{h} \in \mathbb{Z}^s} \frac{1}{(\overline{h}_1 \overline{h}_2 \cdots \overline{h}_s)^\alpha} e^{2\pi i \mathbf{h} \cdot \mathbf{x}}$$

$P_\alpha(Q) :=$ the error of the lattice rule for f_α.

P_α is easy to compute for α an even integer
because f_α can be written as products of Bernoulli polynomials.

Theoretical convergence is

$$O\left((\log(n))^{\alpha s} n^{-\alpha}\right).$$

P_α introduced by (Korobov, 1959)

Obviously related to the figure of merit:

$$\frac{2}{\rho^\alpha} \le P_\alpha.$$

Figure of merit used by (Maisonneuve, 1972)

P_α is easy to compute for α an even integer because f_α can be written as products of Bernoulli polynomials.

Theoretical convergence is

$$O\left((\log(n))^{\alpha s} n^{-\alpha}\right).$$

P_α introduced by (Korobov, 1959)

Obviously related to the figure of merit:

$$\frac{2}{\rho^\alpha} \le P_\alpha.$$

Figure of merit used by (Maisonneuve, 1972)

H. Niederreiter

Other criteria:

- $R(\mathbf{z}, n)$ (Niederreiter, 1987)

$$P_\alpha(\mathbf{z}, n) < R(\mathbf{z}, n)^\alpha + \mathcal{O}(n^{-\alpha})$$

- Discrepancy

$$D_n = O\left(\frac{(\log n)^{s-1}}{\rho}\right)$$

Modern interpretation of P_α is the worst-case error in a RKHS with Korobov kernel with smoothness α.
In general, for a shift-invariant kernel K and rank-1 lattice points

$$e^2(\Lambda, K) = -\int_{[0,1)^s} K(\mathbf{x}, \mathbf{0}) \, \mathrm{d}\mathbf{x} + \frac{1}{n} \sum_{k=0}^{n-1} K\left(\left\{\frac{k\mathbf{z}}{n}\right\}, \mathbf{0}\right)$$

see e.g. (Hickernell, 1998)

Typical form for a weighted space:

$$e_s^2(\mathbf{z}) = -1 + \frac{1}{n} \sum_{k=0}^{n-1} \prod_{j=1}^{s} \left[1 + \gamma_j \, \omega\left(\left\{\frac{k z_j}{n}\right\}\right)\right]$$

This is a tensor product space: a product of 1-dimensional kernels

The weights γ_j, $\gamma_1 \geq \gamma_2 \geq \cdots \geq \gamma_s$, model anisotropicness of the integrand functions

Between the big braces we have the 1-dimensional kernel

Searches for lattice rules

Remember that

1. The cost to verify that a lattice rule has degree d is proportional to d^s, so only “moderate” dimensions are feasible.
2. The search space is huge.

$\Rightarrow$ Restrict the search space.

For example:

Definition

A rank-1 simple lattice is generated by <u>one</u> vector $\mathbf{z}$ and has the form

$$Q[f] := \frac{1}{n} \sum_{j=0}^{n-1} f\left(\left\{\frac{j\mathbf{z}}{n}\right\}\right)$$

$$P_n := \left\{\left\{\frac{j\mathbf{z}}{n}\right\} : j = 0, \ldots, n-1\right\}, \quad \mathbf{z} \in U_n^s \,.$$

For example:

Definition

A rank-1 simple lattice is generated by one vector $\mathbf{z}$ and has the form

$$Q[f] := \frac{1}{n} \sum_{j=0}^{n-1} f\left(\left\{\frac{j\mathbf{z}}{n}\right\}\right)$$

- Restricting to rank-1 simple rules
  $\longrightarrow$ only 1 vector, $s-1$ components, to be determined.
- Further restriction of the search space:
  consider only generator vectors of the form

$$\mathbf{z}(\ell) = (1, \ell, \ell^2 \mod n, ..., \ell^{s-1} \mod n), 1 \leq \ell < n$$

(Korobov, 1959)

Technology used: matrices

Any s-dimensional lattice Λ can be specified in terms of s linearly independent vectors $\{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_s\}$.
$\rightarrow$ These vectors are known as generators of Λ.

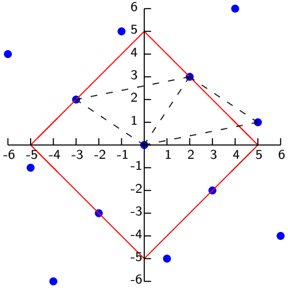
Associated with the generators is an $s \times s$ generator matrix A whose rows are $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_s$.
All $\mathbf{h} \in \Lambda$ are of the form $\mathbf{h} = \sum_{i=1}^{s} \lambda_i \mathbf{a}_i = \boldsymbol{\lambda} A$ for some $\boldsymbol{\lambda} \in \mathbb{Z}^s$.

The dual lattice $\Lambda^\perp$ may be defined as having generator matrix $B = (A^{-1})^T$.

It can be shown that the number of points $n = |\mathrm{det} A|^{-1} = |\mathrm{det} B|$.

Recent searches for low dimensions: K-Optimal rules

Not restricted to rank-1 lattices.
Based on a property of the dual lattice:



Dynamic approach:

- start from an admissible lattice
  (only 0 is inside $\Omega(s, \delta) = \mathrm{conv} S(O_s, \delta)$).
- perturb it

  to reduce the s-volume of its unit cell
  not allowing any lattice point to enter

Making unit cell smaller
$=$ making lattice $L^{\perp}$ denser
$=$ reducing the number of points n of the lattice rule
$=$ moving points toward the origin.

Dynamic approach:

- start from an admissible lattice
  (only 0 is inside $\Omega(s,\delta) = \mathrm{conv} S(O_s,\delta)$).
- perturb it

  to reduce the s-volume of its unit cell
  not allowing any lattice point to enter

Making unit cell smaller
$=$ making lattice $L^\perp$ denser
$=$ reducing the number of points n of the lattice rule
$=$ moving points toward the origin.

Argument by (C. & Lyness, 2001):
It is reasonable to believe that the lattice Λ of an optimal lattice rule will have $\Lambda^\perp$ with many elements on the boundary of $\mathrm{conv} S(O_s, d+1)$ (a scaled version of the unit octahedron).

High computational cost!

- The cost depends mainly on the number of generator matrices considered:
$$\binom{\delta+s-1}{s-1}^s = \mathcal{O}(\delta^{s^2-s})$$
- Only a portion of these is treated further to find n.
- A minute proportion of these are retained to find their degree.
A simple algorithm requires time proportional to $\bar{\delta}^{s-1}$,
where $\bar{\delta}$ is the degree.
Most have degree $< \delta$.

This leads to a complexity bounded above by δ^{s^2-1}.

In spite of shortcuts, exploiting symmetry,... the computational cost increases very rapidly.

In spite of shortcuts, exploiting symmetry,... the computational cost increases very rapidly.

For example

- for $s = 3$, all $\delta \leq 30$ are treated within 33 minutes;
- for $s = 4$, $\delta = 10$ required 6.5 hours (anno 2002).
- for $s = 4$, $\delta = 20$ was estimated at 2000 days
  We further restricted the search space $\rightarrow$ 30 days.

C. & Lyness obtained many new rules

- in 3 dimensions ($\delta \leq 30$) and
- in 4 dimensions ($\delta \leq 24$).

Lyness & Sørevik (Math. Comp., 2006) describe a large scale search in 5 dimensions.
What did we learn from this?
(There was plenty of time to think while we were waiting for computational results.)

Observation:

> In 3D and 4D many K-optimal rules can be generated by skew-circulant matrices.

Circulant and skew-circulant lattices

A circulant lattice is one that can be generated by a circulant matrix, that is a generator matrix of the form

$$\mathcal{C}irc(\mathbf{a}) = \mathcal{C}irc(a_0, a_1, \ldots, a_{s-1}) := \begin{pmatrix} a_0 & a_1 & a_2 & \ldots & a_{s-1} \\ a_{s-1} & a_0 & a_1 & \ldots & a_{s-2} \\ \vdots & \vdots & \vdots & & \vdots \\ a_2 & a_3 & a_4 & \ldots & a_1 \\ a_1 & a_2 & a_3 & \ldots & a_0 \end{pmatrix}$$

Circulant and skew-circulant lattices

A circulant lattice is one that can be generated by a circulant matrix, that is a generator matrix of the form

$$\mathcal{C}irc(\mathbf{a}) = \mathcal{C}irc(a_0, a_1, \ldots, a_{s-1}) := \begin{pmatrix} a_0 & a_1 & a_2 & \ldots & a_{s-1} \\ a_{s-1} & a_0 & a_1 & \ldots & a_{s-2} \\ \vdots & \vdots & \vdots & & \vdots \\ a_2 & a_3 & a_4 & \ldots & a_1 \\ a_1 & a_2 & a_3 & \ldots & a_0 \end{pmatrix}$$

A skew-circulant lattice is one that can be generated by a skew-circulant matrix, that is a generator matrix of the form

$$\begin{aligned} \mathcal{SC}irc(\mathbf{a}) &= \\ \mathcal{SC}irc(a_0, a_1, \ldots, a_{s-1}) &:= \begin{pmatrix} a_0 & a_1 & a_2 & \ldots & a_{s-1} \\ -a_{s-1} & a_0 & a_1 & \ldots & a_{s-2} \\ \vdots & \vdots & \vdots & & \vdots \\ -a_2 & -a_3 & -a_4 & \ldots & a_1 \\ -a_1 & -a_2 & -a_3 & \ldots & a_0 \end{pmatrix} \end{aligned}$$

Drastic reduction of complexity bound! $\longrightarrow \delta^{2s-2}$.
$\Rightarrow$ 4D becomes almost trivial for δ's that required months of cpu-time.
$\Rightarrow$ 5D and 6D is very feasable.

Timing examples:

- Good news: $s = 5, \delta = 10$
  GRISK $\rightarrow$ 6.275 cpu-hours $\approx$ 260 cpu-days
  achieved $n = 1306$

Drastic reduction of complexity bound! $\longrightarrow \delta^{2s-2}$.
$\Rightarrow$ 4D becomes almost trivial for δ's that required months of cpu-time.
$\Rightarrow$ 5D and 6D is very feasable.

Timing examples:

- Good news: $s = 5, \delta = 10$
  GRISK $\longrightarrow$ 6.275 cpu-hours $\approx$ 260 cpu-days
  achieved $n = 1306$
  Circulant $\longrightarrow$ < 3 sec achieved $n = 1322$
- Bad news: $s = 5, \delta = 12$:
  GRISK $\longrightarrow$ 650.755 cpu-hours $\approx$ 74 cpu-years
  achieved $n = 3200$ ($\rho = 0.648$) (21% completed)

Drastic reduction of complexity bound! $\rightarrow \delta^{2s-2}$.
$\Rightarrow$ 4D becomes almost trivial for δ's that required months of cpu-time.
$\Rightarrow$ 5D and 6D is very feasable.

Timing examples:

- Good news: $s = 5, \delta = 10$
  GRISK $\rightarrow$ 6.275 cpu-hours $\approx$ 260 cpu-days
  achieved $n = 1306$
  Circulant $\rightarrow$ <3 sec achieved $n = 1322$
- Bad news: $s = 5, \delta = 12$:
  GRISK $\rightarrow$ 650.755 cpu-hours $\approx$ 74 cpu-years
  achieved $n = 3200$ ($\rho = 0.648$) (21% completed)
  Circulant $\rightarrow$ $\approx$3 sec achieved $n = 3844$ ($\rho = 0.539$)

High computational cost, $\mathcal{O}(\delta^{s^2-1})$. $(\delta = d+1)$

- (C. & Lyness, *Math. Comp.*, 2001): 3D ($\delta \leq 30$, 4D ($\delta \leq 24$)
- (Lyness & Sørevik, *Math. Comp.*, 2006): 5D ($\delta \leq 12$)

High computational cost, $\mathcal{O}(\delta^{s^2-1})$. $(\delta = d+1)$

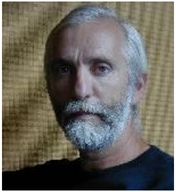
- (C. & Lyness, *Math. Comp.*, 2001): 3D ($\delta \leq 30$, 4D ($\delta \leq 24$)
- (Lyness & Sørevik, *Math. Comp.*, 2006): 5D ($\delta \leq 12$)

Restricting the search to (skew-)circulant generator matrices, reduces the cost to $\mathcal{O}(\delta^{2s-2})$.

- (Lyness & Sørevik, *Math. Comp.*, 2004): 4D
- (C. & Govaert, *J. Complexity*, 2003): 5D, 6D

This also lead to closed expressions for arbitrary degrees.

J. Lyness



T. Sørevik

H. Govaert

Packing factor

Definition

The packing factor

$$\hat{\rho}(n) := \frac{\delta^s}{s!n}.$$

This is a measure of the efficiency of a rule.
It is convenient for making pictures because $0 \leq \hat{\rho}(n) \leq 1$.

Actually, $\hat{\rho}(n)$ is bounded above by
the density of the densest lattice packing
of the crosspolytope (octahedron) $\theta(O_s)$.
($\rightarrow$ link with "Geometry of numbers")

Known values:

- $\theta(O_1) = \theta(O_2) = 1$
- $\theta(O_3) = \frac{18}{19}$ (Minkowski, 1911) used by (Frolov, 1977)

This provides a (higher) lower bound for lattice rules for trigonometric degree:

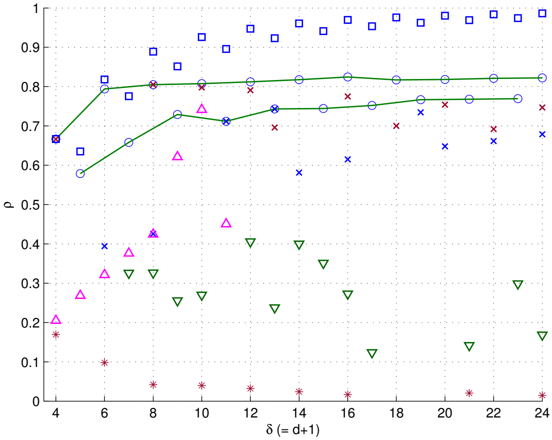
$$n \geq \frac{(d+1)^s}{s!\theta(O_s)}.$$

Lattice rules provide constructive lower bounds for $\theta(O_s)$. From a lattice rule with n points follows

$$\theta(O_s) \geq \frac{(d+1)^s}{s!n}.$$

The best known bounds for $\theta(O_4)$, $\theta(O_5)$ and $\theta(O_6)$ come from lattice rules
(C., *East Journal on Approximations*, 2006).

Results: 4D



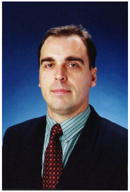
○ refers to n_{KO}, □ refers to n_{ME}.
× refers to (Noskov & Semenova, 1996)+corrections
∗ refers to (C., Novak & Ritter, 1999)
× refers to (Temirgaliev, 1991), △ refers to Good lattices
▽ refers to Korobov rules (Maisonneuve, 1972)

Recent searches for higher dimensions: Component-by-component construction

- Focus on rank-1 lattice rules $\Rightarrow$ find 1 vector $\mathbf{z}$.
- Idea: search $\mathbf{z}$ component by component

2000: I. Sloan & A. Reztsov (Tech. Report)
published Math. Comp. 2002
unweighted Korobov space, n prime

I.H. Sloan



A. Reztsov

Some milestones of component-by-component

- 2000-2002: F. Kuo (PhD) with S. Joe
  weighted Korobov space, weighted Sobolev space
- MCQMC 2002: J. Dick & F. Kuo
  basically for weighted Korobov space, n a product of few primes, but partial search, faster and for millions of points
- MCQMC 2004, 2006: D. Nuyens & C.
  fast construction in $O(sn\log(n))$, basic case for n prime, but also possible for any composite n (and full search)

F. Kuo

S. Joe

J. Dick

D. Nuyens

The CBC algorithm in a shift-invariant RKHS

for $s = 1$ **to** s_{max} **do**

for all z **in** U_n **do**

$$e_s^2(z) = -1 + \frac{1}{n} \sum_{k=0}^{n-1} \prod_{j=1}^{s} \left[1 + \gamma_j \, \omega\left(\left\{ \frac{k z_j}{n} \right\} \right) \right]$$

end for

$$z_s = \operatorname*{argmin}_{z \in U_n} e_s^2(z)$$

end for

Computational cost: $O(s_{\mathsf{max}} n^2)$

Rephrasing CBC: matrix-vector form

The inner loop can be formulated as a matrix-vector product with matrix

$$\boldsymbol{\Omega}_n := \left[\omega\left(\left\{\frac{kz}{n}\right\}\right)\right]_{\substack{z\in U_n\\ k\in\mathbb{Z}_n}} = \left[\omega\left(\frac{k\cdot z \bmod n}{n}\right)\right]_{\substack{z\in U_n\\ k\in\mathbb{Z}_n}}$$

This matrix has a lot of structure!

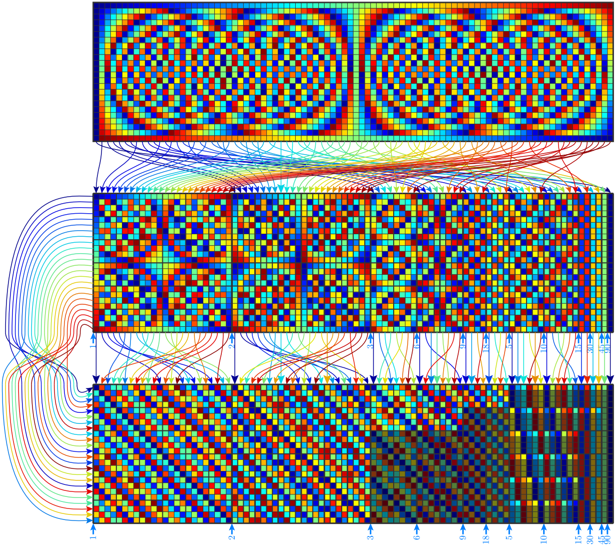
A matrix-vector multiplication can be done in $O(n\log n)$
(Nuyens & C. 2005, 2006)

$\boxed{\Rightarrow \text{Construction then takes } O(sn\log n) \text{ using } O(n) \text{ memory}}$

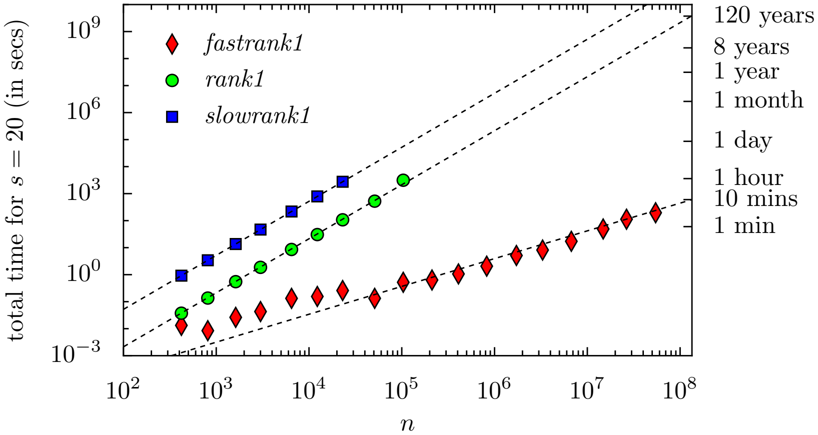
An example matrix Ω_n and its permutations

A nice view on $90 = 2 \times 3^2 \times 5$

The blocks of the last matrix are diagonizable by FFT's



Results in $O(sn\log(n))$



Timings generated on a P4 2.4GHz ht, 2GB RAM for 20 dimensions

Periodizing or not?

Lattice rules are constructed to be optimal for periodic functions.
E.g., for the Korobov class $E_s^\alpha(c)$ of periodic functions with smoothness $\alpha \geq 2$.
$\rightarrow$ Theoretical convergence is $O\left((\log(n))^{\alpha s} n^{-\alpha}\right)$.

Periodizing or not?

Lattice rules are constructed to be optimal for periodic functions.
E.g., for the Korobov class $E_s^\alpha(c)$ of periodic functions with smoothness $\alpha \geq 2$.
$\rightarrow$ Theoretical convergence is $O\left((\log(n))^{\alpha s} n^{-\alpha}\right)$.

A non-periodic function can be transformed into such a space by a periodizing transform ϕ

$$\int_{[0,1)^s} f(x_1, \ldots, x_s) \, \mathrm{d}\mathbf{x} = \int_{[0,1)^s} f(\phi(x_1), \ldots, \phi(x_s)) \, \phi'(x_1) \cdots \phi'(x_s) \, \mathrm{d}\mathbf{x}$$

Periodizing or not?

Lattice rules are constructed to be optimal for periodic functions.
E.g., for the Korobov class $E_s^\alpha(c)$ of periodic functions with smoothness $\alpha \geq 2$.
$\rightarrow$ Theoretical convergence is $O\left((\log(n))^{\alpha s} n^{-\alpha}\right)$.

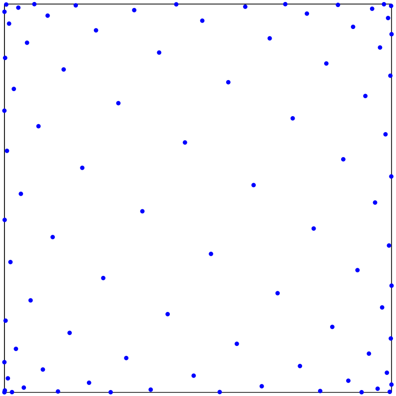
A non-periodic function can be transformed into such a space by a periodizing transform ϕ

$$\int_{[0,1)^s} f(x_1, \ldots, x_s)\,\mathrm{d}\mathbf{x} = \int_{[0,1)^s} f(\phi(x_1), \ldots, \phi(x_s))\,\phi'(x_1)\cdots\phi'(x_s)\,\mathrm{d}\mathbf{x}$$

Example: Sidi transform (Sidi, 1993) with $\phi(x) = x - \frac{1}{2\pi}\sin(2\pi x)$

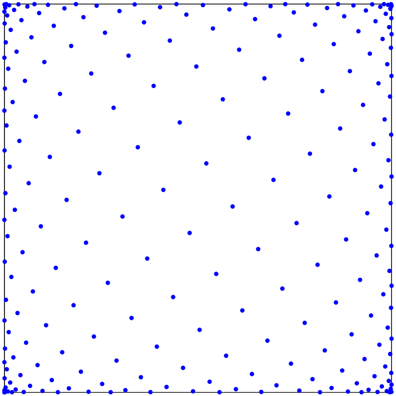
A.Sidi

This could be seen as using a transformed point set, $x \mapsto \phi(x)$, with weights $w_k = \prod_{j=1}^{s} \phi'(x_j^{(k)})$.



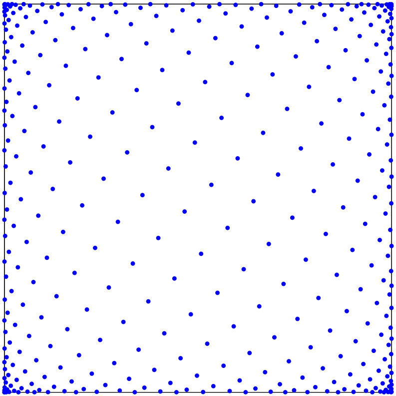
$n = F_{11} = 89$

This could be seen as using a transformed point set, $x \mapsto \phi(x)$, with weights $w_k = \prod_{j=1}^{s} \phi'(x_j^{(k)})$.



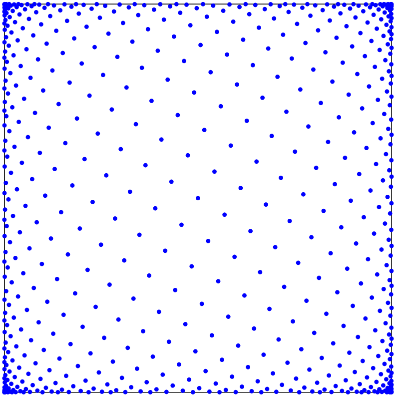
$n = F_{13} = 233$

This could be seen as using a transformed point set, $x \mapsto \phi(x)$, with weights $w_k = \prod_{j=1}^{s} \phi'(x_j^{(k)})$.



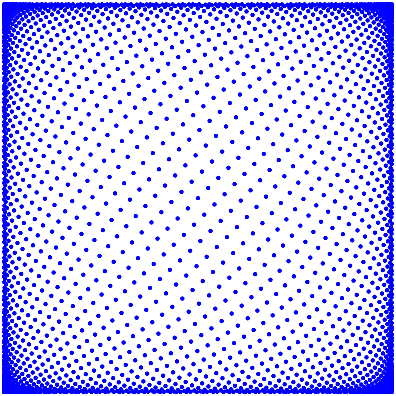
$n = F_{14} = 377$

This could be seen as using a transformed point set, $x \mapsto \phi(x)$, with weights $w_k = \prod_{j=1}^{s} \phi'(x_j^{(k)})$.



$n = F_{15} = 610$

This could be seen as using a transformed point set, $x \mapsto \phi(x)$, with weights $w_k = \prod_{j=1}^{s} \phi'(x_j^{(k)})$.



$$n = F_{18} = 2584$$

Problems with periodization:

- Selecting the right transform for your function is not trivial.
  Many exist, mainly used in 1D.
- The factor $(\log(n))^{\alpha s}$ is often very well visible (worse for higher s).
  $\Rightarrow$ more initial points needed to achieve the hoped for $O(n^{-\alpha})$.
- Leads to machine dependent cubature rules.
  - When n gets higher calculations have to be done in higher precision. (IEEE double is not enough; different points $\mapsto$ same floating point representation; floating point cube $[0,1]^s$ is not symmetric.)
  - When s gets higher the weights at the sides get very small.

  Well known long time ago. Recent reference:
  (Hill & Robinson, *Theoretical Comp. Sci.*, 2006)

Conclusion:
Periodization is a powerful tool in the hands of an expert.
In the hands of a novice, it is a dangerous tool!

Error estimation

In practice one wants more than 1 approximation.

Common approaches (for all types of cubature):

- randomization (randomly shifted rules) (Cranley & Patterson, 1976)

Error estimation

In practice one wants more than 1 approximation.

Common approaches (for all types of cubature):

- randomization (randomly shifted rules) (Cranley & Patterson, 1976)
- embedded sequences
  - copy rules, with intermediate lattice rules (Joe & Sloan, 1992) augmentation sequences (Li, Hill & Robinson, 2007)
  - embedded rank-1 rules
    (Hickernell, Hong, L'Ecuyer, Lemieux, *SISC* 2000)
    (C., Kuo, Nuyens, *SISC* 2006)

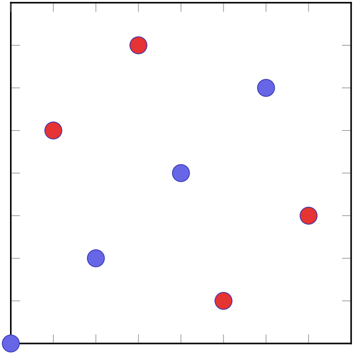

T. Patterson


R. Hong


P. L'Ecuyer

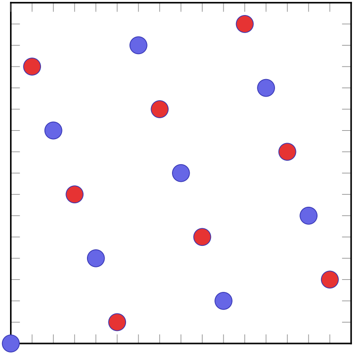

C. Lemieux

Example of embedded rank-1 rules



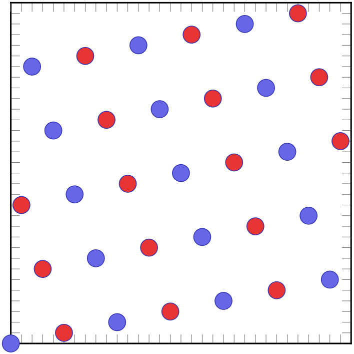
$n = 8$

Example of embedded rank-1 rules



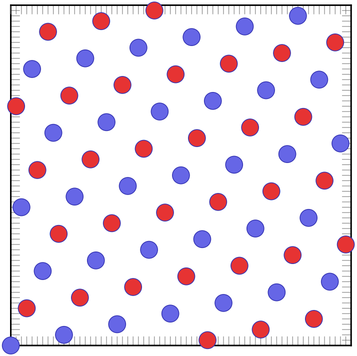
$n = 16$

Example of embedded rank-1 rules



$n = 32$

Example of embedded rank-1 rules



$n = 64$

Generating points in non-trivial order

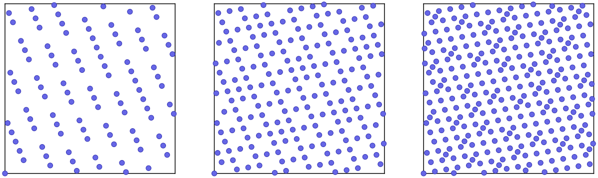
- Simple rank-1 lattice:

$$\mathbf{x}^{(k)} = \left\{ \frac{k\,\mathbf{z}}{n} \right\}, \text{ for } k = 0, 1, 2, \ldots, n-1.$$

- Embedded rank-1 lattice: in order to stop at any time, you need a good ordering of the points:

$$\mathbf{x}^{(k)} = \left\{ \frac{\varphi(k)}{n}\,\mathbf{z} \right\}, \text{ for } k = 0, 1, 2, \ldots, n-1.$$

- If n is very large, this can be seen as an extensible cubature rule.



$n = 100$ $n = 200$ $n = 300$

Are lattice rules used?

Are lattice rules used?

Low dimensions

General purpose, black-box integration routines for 2D:

- DITAMO: (Robinson & De Doncker, 1981)
  product rectangle rule + periodizing transfo (IMT)
- d2lri (Hill & Robinson, 1999)
- r2d2lri (Robinson & Hill, *ACM TOMS*, 2002)

All based on sequences of embedded rules and error estimators derived from these.

I. Robinson

E. De Doncker

Are lattice rules used?

Low dimensions

General purpose, black-box integration routines for 2D:

- DITAMO: (Robinson & De Doncker, 1981)
  product rectangle rule + periodizing transfo (IMT)
- d2lri (Hill & Robinson, 1999)
- r2d2lri (Robinson & Hill, *ACM TOMS*, 2002)

All based on sequences of embedded rules and error estimators derived from these.

Application of 2D lattice rules:
M. Revers, "Numerical integration of the Radon transform on classes E_s^α in multiple finite dimensions", *Computing*, 1995.

Are lattice rules used?

Intermediate dimensions

Suppose we want to calculate

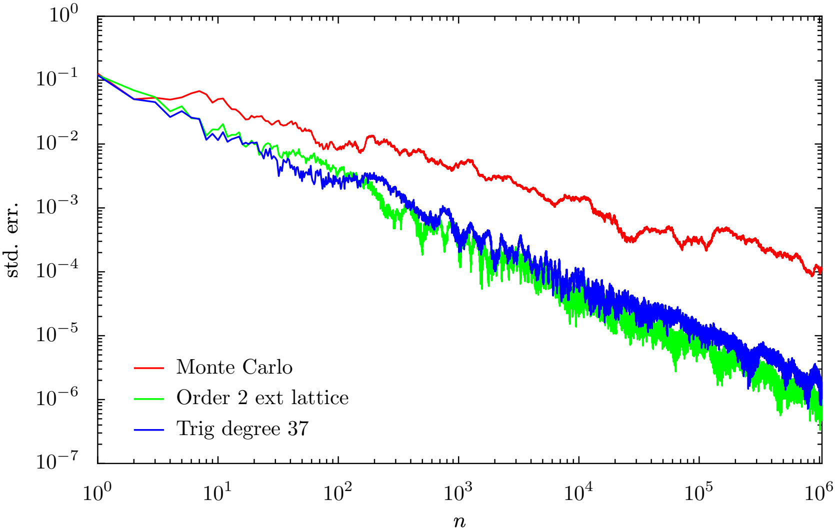
$$T_1 = \int_{\mathbb{R}^s} \frac{\exp\left(\sum_{q=1}^{s} \theta_q - \beta\right)}{1 + \exp\left(\sum_{q=1}^{s} \theta_q - \beta\right)} \mathcal{N}_s(\theta; \mathbf{0}, \mathbf{\Sigma}) \, \mathrm{d}\theta$$

which is a simplified version of an integral popping up in a likelihood problem to estimate $\mathbf{\Sigma}$ and β.

We consider the problem in 5-dimensions using

- plain Monte Carlo
- a lattice rule with trigonometric degree 37 and $n = 1\,044\,808$
- an extensible lattice rule for 2^{10} to 2^{20} points in a Sobolev space

A numerical example



Are lattice rules used?

High dimensions

Positive results,
mainly illustrated with examples from mathematical finance.
(These examples actually revived the interest in quasi-Monte Carlo methods.)
Preprocessing required:

1. transformation to unit cube
2. rearranging the dimensions (e.g., by using PCA)
3. determining weights of the dimensions

Final remarks

Construction:

- Searches for lattice rules using the “classical” criteria are doomed to fail for increasing dimensions.
- The CBC algorithm, relying on “worst-case-error” for “reproducing kernel Hilbert spaces” beats this curse of dimensionality. Rules can be constructed very fast even if n and s are large.

But work remains to be done, e.g.,

- for CBC, tuning of the function space using the weights,
- practical error estimates based on sequences.

Finally note that

- lattice rules are useful for low and high dimensions, and are not only for integrating periodic functions;
- all quality criteria have a reason to exist;
- the difference between lattice rules and “classical” low discrepancy sequences evaporates.
  Lattice rules with large n can be constructed easily and can be used as sequences.

Finally note that

- lattice rules are useful for low and high dimensions, and are <u>not</u> only for integrating periodic functions;
- all quality criteria have a reason to exist;
- the difference between lattice rules and "classical" low discrepancy sequences evaporates.
  Lattice rules with large n can be constructed easily and can be used as sequences.

Use a lattice rule anywhere & anytime!

The end.

Thank you!