ELSEVIER

Journal of Computational and Applied Mathematics 149 (2002) 1–12

JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS

www.elsevier.com/locate/cam

Advances in multidimensional integration

Ronald Cools

Department of Computer Science, Katholieke Universiteit Leuven, Celestijnenlaan 200A, B-3001 Heverlee, Belgium

Received 27 October 2001; received in revised form 6 May 2002

Abstract

This paper gives a personal bird's eye view of some aspects of multivariate numerical integration. It will sketch what happened during the past 50 years and point the reader's attention to what did not happen.

Keywords: Cubature; Quadrature; Numerical integration

1. Introduction

In this paper, I will give a very personal view on some aspects of the approximations of multivariate integrals

$$I[f] := \int_{\Omega} w(\mathbf{x}) f(\mathbf{x})\,\mathrm{d}\mathbf{x}, \quad \Omega \subset \mathbb{R}^d, \quad w(\mathbf{x}) \geqslant 0 \text{ for all } \mathbf{x} \in \Omega. \tag{1}$$

Such integrals appear in many mathematical models and can seldom be calculated analytically. This is especially true for multivariate integrals. The numerical approximation on integrals is one of the corner stones of numerical analysis. We encounter it in software for finite and boundary element problems. Statistical software also contains techniques to approximate integrals. The approximation of integrals is also an important step in methods to solve integral equations, e.g., in computer graphics. An application area that recently became very popular is financial mathematics: determining the value of sophisticated financial derivatives, such as exotic options, and determining the value at risk.

We are used to approximating functions by weighted sums of "easier" functions, such as monomials, splines or wavelets. Similarly, we are used to approximating functionals by weighted sums of "easier" functionals. An integral is typically approximated by a weighted sum of integrand

E-mail address: ronald.cools@cs.kuleuven.ac.be (R. Cools).

0377-0427/02/$ - see front matter
PII: S0377-0427(02)00517-4

evaluations. One must keep in mind that one may also use other functionals, such as derivatives of the integrand or even integrals of lower-dimensional projections. Here, I will only consider approximations of the form

$$Q[f] := \sum_{j=1}^{N} w_j f(\mathbf{y}^{(j)}) \quad \text{with } \mathbf{y}^{(j)} \in \Omega. \tag{2}$$

If the dimension of the integration region $d = 1$ the approximation Q is called a *quadrature formula*. If $d \geqslant 2$ the approximation Q is called a *cubature formula*. The term *integration rule* is also used.

Basic integration rules (cubature formulas) for the approximation of multiple integrals are essential building blocks in many applications. In this text, I will almost entirely restrict myself to this topic. I will ignore almost completely the important aspect of error estimators, convergence acceleration by extrapolation and adaptive subdivisions strategies.

How should the points and weights of a basic integration rule be chosen? The choice of the points $\mathbf{y}^{(j)}$ and weights w_j is independent of the function f. There is no unique best criterion for this choice. There are several criteria to specify and classify integration rules based on their behaviour for specific classes of functions. People seem to agree that the approximation should be exact for constant functions, that is,

$$I[1] = \int_{\Omega} w(\mathbf{x})\,\mathrm{d}\mathbf{x} = \sum_{j=1}^{N} w_j = Q[1].$$

Roughly speaking, one can distinguish two classes of cubature formulas: *polynomial-based methods* and *number theoretic methods*. The first class contains rules designed to be exact for some set of polynomials, while the second contains rules based on uniformly distributed points.

The polynomial-based methods received most attention during the past century and that is the topic I have to say most about. I devote some space to quasi-Monte Carlo methods because the future of these methods looks very bright.

2. Will faster computers make everything trivial eventually?

In a "Present state of the art" section in their survey book [8], Davis and Rabinowitz wrote that

> Given the present state of theory and hardware (1983), one can distinguish three dimensional ranges in addition to the two-dimensional case which deserves separate treatment:
>
> *Range* I: integrals of dimension 3 to about 6 or 7,
> *Range* II: dimensions 7 or 8 to about 15,
> *Range* III: dimensions greater than 15.

In two dimensions, they considered the situation satisfactory. In Range I, they considered adaptive methods based on rules exact for polynomials as the most important tools. Range II was considered the borderline range. Here, much depends on the smoothness of the integrand. In this range, adaptive Monte Carlo and quasi-Monte Carlo methods definitely become competitive, but only low accuracy is

achievable. Range III was considered "really high" dimensional. Monte Carlo and quasi-Monte Carlo are indicated. One should not hope for more than 2 digits accuracy. Is this much different today?

To put everything in perspective, let us turn back the clock about 20 years, when the routine TRIEX [9] was developed to approximate integrals with singularities over a triangle. The authors put an upper limit of 23 000 function evaluations on their experiments. The experiments for TRIEX were done on an IBM 370/158 which was a 1 MIPS machine. Today a Pentium III is about 1000 MIPS. One cpu-hour of 1980 is devaluated to about 4 cpu seconds in the year 2000. (The real speedup will be less since the measure "mips" cannot really be used to compare machines developed 20 years apart.) So, current machines can afford to spend more function evaluations on a problem to have it completed in the same amount of wall clock time.

Will faster computers be sufficient to solve our future problems even if we do not make any other progress? I believe not. As computers become more powerful every day, researchers start thinking about problems nobody dreamed about some years ago: integrals with hundreds of variables. Solving these problems remains very difficult or even impossible. Application areas will demand more and more. The growing speed of computers alone will not help. Indeed, we are working with a lot of techniques that were developed with low dimensions in mind and some of these cannot directly be applied in high dimensions. That is the main reason why the ranges indicated by Davis and Rabinowitz apply almost unchanged.

For the approximation of low-dimensional integrals a reasonable collection of techniques and software is nowadays available. The approximation of multivariate integrals is however a fundamentally different problem. In this research area, one suffers from the so-called curse of dimensionality: the computational complexity grows exponentially with the dimension.

3. Polynomial-based methods

Polynomials are very popular for approximations. The oldest criterion for the approximation of integrals is the *algebraic degree* of a quadrature or cubature formula. A quadrature or cubature formula has algebraic degree m if it is exact for all polynomials or (total) degree at most m.

One can think about several variants of this criterion, e.g., one can consider the trigonometric degree or one can consider other spaces of polynomials.

By $\mathbb{P}^d$ we denote the vector space of all polynomials in d variables. This space is spanned by the monomials

$$\mathbf{x}^{\mathbf{k}} := x_1^{k_1} \cdots x_d^{k_d}, \quad \mathbf{k} := (k_1, \ldots, k_d) \in \mathbb{N}^d.$$

The (total) degree of a monomial is

$$\deg(\mathbf{x}^{\mathbf{k}}) := |\mathbf{k}| := \sum_{j=1}^{d} k_j.$$

By $\mathbb{P}_m^d$ we denote the subspace of all polynomials in d variables of degree at most m. The definition of degree of a cubature formula is equivalent with demanding that

$$Q[f_j] = I[f_j], \quad j = 1, 2, \ldots, \dim \mathbb{P}_m^d, \tag{3}$$

where the f_j form a basis for $\mathbb{P}_m^d$.

If f_j and N are fixed, then (3) form a system of nonlinear equations

$$\sum_{i=1}^{N} w_i f_j(\mathbf{y}^{(i)}) = I[f_j], \quad j = 1, 2, \ldots, \dim \mathbb{P}_m^d. \tag{4}$$

Each point introduces $d+1$ unknowns: the d coordinates of the point $\mathbf{y}^{(i)}$ and the weight w_i. Actually, each equation in (4) is a polynomial equation.

One can distinguish between two approaches to construct cubature formulas:

(1) search for polynomials that vanish at the points of the formula;
(2) solve the system of nonlinear equations (4) directly.

In Sections 3.1 and 3.2, I will briefly introduce these approaches. For a more through introduction I refer to [3] and its references.

3.1. Vanishing polynomials

The approximation of one-dimensional integrals using a quadrature formula based on the zeros of an orthogonal polynomial is a very classical mathematical subject. Only very few mathematicians tried to generalise this to higher dimensions prior to the middle of the previous century. A very important milestone was Radon's paper [17] which triggered a lot of research [6]. Only in the 1960s the effort increased and significant progress was made by, amongst others,[1] Stroud and Mysovskikh. (See their survey books [21] and [14].) They introduced elements from algebraic geometry. Möller [13] was the first to recognise that this connection could be presented more transparently by using ideal theory. Significant theoretical progress was made in the 1970s and 1980s with the introduction of ideal theory. The practical results were however disappointing.

An apparently simple question such as "What is the minimal number of points required by a cubature formula of degree m for the integral I" does not have a simple answer.

A very general upper and lower bound is given in the following theorem, which combines the easy-to-prove lower bound for positive functionals and Tchakaloff's upper bound, see e.g., [21].

Theorem 1. *A cubature formula Q of the form* (2) *of degree m for an integral* (1) *exists with its number of points N satisfying*

$$\dim \mathbb{P}_{\lfloor m/2 \rfloor}^d \leqslant N \leqslant \dim \mathbb{P}_m^d.$$

[1] I restrict myself to the two most mentioned names in this area.

Actually, no cubature formula can exist with less points than $\dim \mathbb{P}^d_{\lfloor m/2 \rfloor}$. The common proof of this lower bound is not constructive. If one has a cubature formula with more than $\dim \mathbb{P}^d_m$ points, then one can always delete points and recompute the weights by solving a system of linear equations. Furthermore, it is known that a cubature formula attaining this upper bound exists with only real points inside Ω and all weights positive.

Observe that the lower bound of Theorem 1 is equal for $m = 2k$ and $2k + 1$. So, it will not come as a surprise that in general this lower bound cannot be attained for odd degrees. There is a connection with orthogonal polynomials which in general shows this.

In the multivariate case there is more than one orthogonal polynomial of a given degree. It is common to normalise the orthogonal polynomials of degree k to the form

$$p_k = \mathbf{x}^{\mathbf{k}} + q_k, \quad \mathbf{k} \in \mathbb{N}^d, \ |\mathbf{k}| = k, \ \deg(q_k) \leqslant k - 1. \tag{5}$$

Orthogonal polynomials in the form (5) are called *basic orthogonal polynomials*. Each basic orthogonal polynomial of degree k has only one term of degree k, with constant 1.

As an illustration we formulate a generalisation by Mysovskikh of a property of Gauss quadrature formulas.

Theorem 2. *A necessary condition for the existence of a cubature formula of degree* $2k + 1$ *with* $N = \dim \mathbb{P}^d_k$ *points is that the basic orthogonal polynomials of degree* $k + 1$ *have* N *common zeros.*

It is nowadays known that the condition of this theorem does not hold for standard regions such as a square or triangle with a constant weight function.

Consider the vector space of all polynomials that vanish in a given set of points. This is more than simply a vector space. Indeed, if one multiplies a polynomial that vanishes at all points of the cubature formula by an arbitrary polynomial, the product also vanishes at all points. The set of all polynomials that vanish in a given set of points is a polynomial ideal.

The following theorem, first proven by Möller [13], shows the relation between ideals and cubature formulas.

Theorem 3. *Let* I *be an integral of the form* (1) *over a* d*-dimensional region. Let* $\{\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(N)}\} \subset \mathbb{C}^d$ *and* $\mathfrak{A} := \{f \in \mathbb{P}^d : f(\mathbf{y}^{(i)}) = 0, \ i = 1, \ldots, N\}$. *Then the following statements are equivalent.*

- $f \in \mathfrak{A} \cap \mathbb{P}^d_m$ *implies* $I[f] = 0$.
- *There exists a cubature formula* Q *of the form* (2) *such that* $I[f] = Q[f]$, *for all* $f \in \mathbb{P}^d_m$, *with at most* $\dim \mathbb{P}^d_m - \dim(\mathfrak{A} \cap \mathbb{P}^d_m)$ (*complex*) *weights different form zero.*

Using ideal theory, a theorem such as Theorem 2 becomes very easy to prove, see [3].

Theorem 3 allows that the cubature formulas have complex or multiple points. Schmid proposed to avoid this by considering *real ideals.*

If the common zeros of an ideal $\mathfrak{A}$ are denoted by $V(\mathfrak{A})$ and the ideal of all polynomials which vanish at a finite set $X \subset \mathbb{R}^d$ by $\mathfrak{A}_X$, then obviously $X \subset V(\mathfrak{A}_X)$. An ideal $\mathfrak{A}$ is called real if $X = V(\mathfrak{A}_X)$. Using real ideals Schmid obtained a complete characterisation of cubature formulas

with real points. His characterisation was successfully applied to construct cubature formulas for several two-dimensional regions, see, e.g., [18,19].

Meanwhile higher lower bounds are known. In [3] all these are collected together with all known formulas that attain these bounds. As an illustration, I mention the improved lower bound for regions such as a square and a triangle with a constant weight function.

Theorem 4. *The number of points N in a cubature formula* (2) *of degree* $2k-1$ *for an integral* (1) *where* Ω *is a square or a triangle and* $w(x)=1$, *satisfies*

$$N \geqslant \frac{k(k+1)}{2} + \left\lfloor \frac{k}{2} \right\rfloor .$$

We know nowadays that the minimal number of points depends on the region of integration *and* on the weight function. We still do not know the lowest possible number of points for simple regions such as a square or triangle with a constant weight function. For these simple regions there is a gap between the highest known lower bound for the number of points, given in Theorems 1 and 4, and the lowest number of points in a known formula already for degree resp. 13 and 9. In Table 1 we summarise what is known about lower bounds and lowest number of points in a known cubature formula for these two standard two-dimensional regions. Observe that for a square for even degrees $\geqslant 10$ no formula is known (except of course those of higher odd degree).

Although some mathematicians continue this line of research, I believe it is exhausted for standard regions and I expect that only very few (if any) new cubature formulas will be constructed during the following years based on this approach. Actually, for standard regions most known cubature formulas exact for polynomials were not constructed with the aid of this theory.

Looking for cubature formulas using orthogonal polynomials will become of practical interest again only if we better understand the relation between the minimal number of points in a cubature formula and the given integral. The construction methods based on this theory strongly depend on the quality of the known lower bounds and that restricts their applicability.

3.2. Direct approach

One can construct cubature formulas exact for a space of polynomials by solving the large system of polynomial equations associated with it. As computers continue to become more powerful, one can give them larger systems and occasionally they will chew out a useful (real points inside the integration region) cubature formula. Although significant progress was made in recent years on methods to solve systems of polynomial equations, these methods still cannot deal with systems associated to cubature formulas. No good news on this is expected in the near future.

We should however keep in mind that most known formulas were first obtained by solving such a nonlinear system. This was not done by specialised techniques for polynomial equations but by iterative zero finders. To make this approach work, it is essential to restrict the search to cubature formulas with a certain structure. The more symmetry imposed on the cubature

Table 1
Overview of known results

Degree	$N_{\min}$	$\tilde{N}_{\min}$	□	◺
1	1	1	1	1
2	3		3	3
3	3	4	4	4
4	6		6	6
5	6	7	7	7
6	10		10	10
7	10	12	12	12
8	15		15	15
9	15	17	17	19
10	21			22
11	24	24	24	27
12	28			33
13	28	31	33	36
14	36			42
15	36	40	44	48
16	45			52
17	45	49	57	61
18	55			70
19	55	60	68	73
20	66			79

$N_{\min}$ is the lower bound of Theorem 1; $N_{\min}$ is the lower bound of Theorem 4; □ denotes a square with constant weight function; ◺ denotes a triangle with constant weight function.

formula, the smaller the system becomes. There are systematic ways to deal with symmetries in cubature formulas but this approach still requires a good part of luck. For an introduction, see [3].

3.3. Consolidation phase

The time has come to consolidate what is known about cubature formulas exact for polynomials and to make this knowledge more easily available to potential users. The Handbook of Mathematical Functions [1] contains Radon's cubature formula of degree 5 with 7 points and this probably explains the widespread use of that formula. The National Institute of Standards and Technology (NIST) is working on a new (digital) version of this widespread Handbook. (See [12] and http://dlmf.nist.gov/ for more about this project.) Whatever it will contain, will become widespread! It is not expected to include many tables of cubature formulas however and so there is room and need for another initiative in this specialised area. We collected the references to known cubature formulas [7,4] and started to make these available electronically. Meanwhile the webpage at, www.cs.kuleuven.ac.be/~nines/research/ech/ecf.html also contains the points and weights of many cubature formulas. The formulas that are included are recomputed and

presented with 16 and 32 significant digits. The files are computer generated, avoiding human typing errors.

3.4. Is there a need for higher degree cubature formulas?

In many two- and three-dimensional applications, practitioners use very low-degree cubature formulas (even degree 1) and they deal with their problem by using a finer triangularisation. A significant number of users of the above webpage say they want to apply higher-order methods, which we all hope will require less subregions.

An example illustrating the positive effect of using higher-degree cubature formula is the following. Consider

$$\int_0^1 \int_0^1 \int_0^1 r^{-0.5} \log r \exp(xy + z)\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z \quad \text{with } r := \sqrt{x^2 + y^2 + z^2}. \tag{6}$$

We approximate this integral using CUBPACK [5]. In Figs. 1 and 2 the results are presented when a formula of degree 7 (diamond), degree 9 (square) and degree 13 (circle) is used. These are connected by a different type of line for clarification. In Fig. 1 the estimated error is presented. In

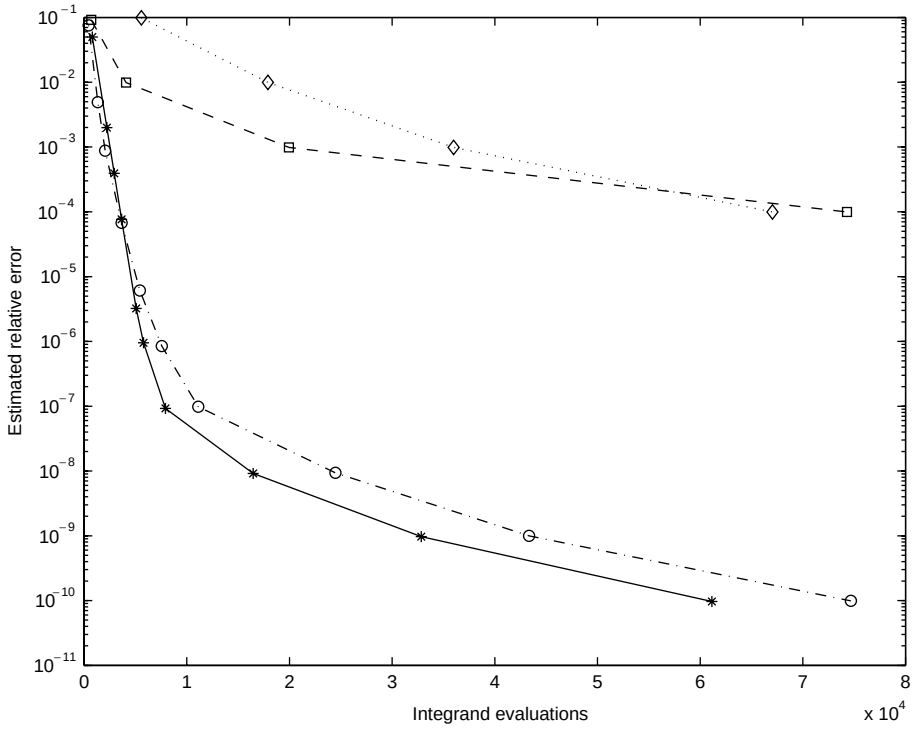

Fig. 1. Comparison of estimated errors for the integral (6).

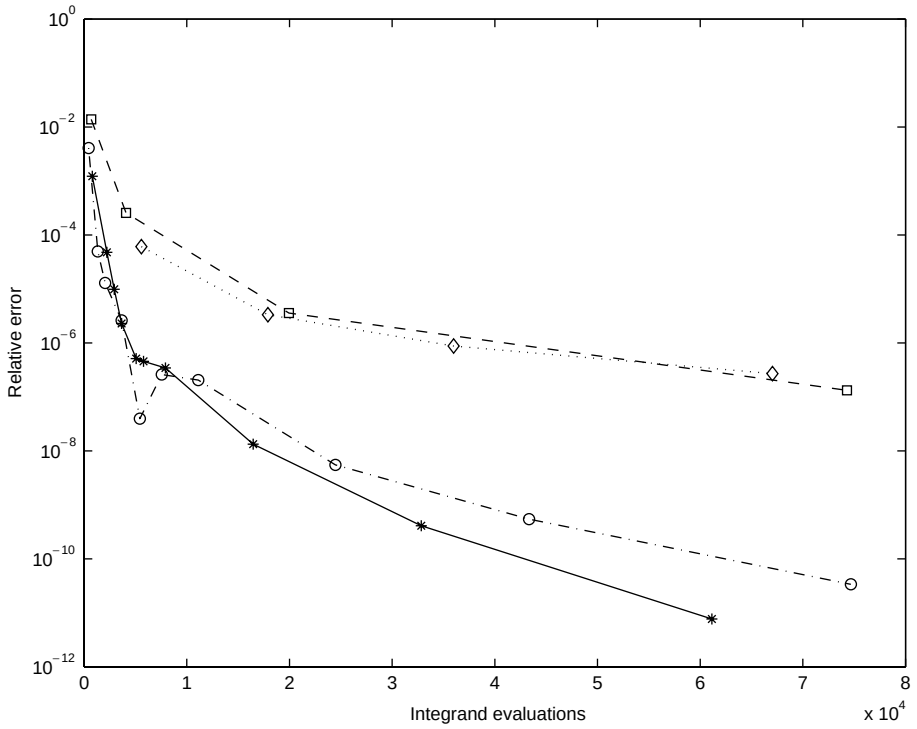

Fig. 2. Comparison of real errors for the integral (6).

Fig. 2 the real error is presented. In all cases the same global adaptive strategy with subdivision in 2 at each step was used. The stars represent results obtained using the degree 13 rule in combination with an improved subdivision strategy. This clearly shows that high accuracy can only be obtained using the high-degree rule.

4. Quasi-Monte Carlo methods

For higher-dimensional problems, practitioners use Monte Carlo methods since many years. In many situations it is the method of last resort. The weight function $w(\mathbf{x})$ in I (1) is now interpreted as a probability density function (pdf) and the value of the integral is interpreted as the mean or expected value of the integrand f. If the points $\mathbf{y}^{(j)}$ are random variables selected according to the pdf $w(\mathbf{x})$ then, given an $\varepsilon > 0$, we have

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

The probability that the sample average is close to the mean value I can theoretically be made arbitrarily close to 1 if we take sufficiently large samples.

An advantage of a Monte Carlo method is that one obtains an error estimate almost for free. The central limit theorem gives a probabilistic error bound:

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

The variance of the integral

$$\sigma^2 = \int_{\Omega} (f(\mathbf{x}) - I[f])^2 w(\mathbf{x})\,\mathrm{d}\mathbf{x} = \int_{\Omega} f^2(\mathbf{x}) w(\mathbf{x})\,\mathrm{d}\mathbf{x} - I[f]^2$$

can with almost no extra cost be estimated from the function values. This very general and easy applicable approach has two drawbacks. First, its convergence is very slow (proportional to $N^{1/2}$) and the method is only useful to meet low accuracy requests. Second, the error estimate is only probabilistic.

It is important to note that the error bound consists of two parts: the variance (which is a property of the integrand) and the number of samples (which is a property of the set of points used). There is half-a-century experience in variance reduction techniques which is the main reason these methods are so successful!

A variant of the Monte Carlo method called *quasi-Monte Carlo*, received increasing attention from mathematicians after the work by (to mention just tow names) Sobol and Korobov. In a quasi-Monte Carlo method the random points of a Monte Carlo method are replaced by deterministic points that are distributed "better than randomly". Moreover, quasi-Monte Carlo methods provide the user with deterministic error bounds.

These theoretical bounds are in general very pessimistic and that is one of the reasons for which these methods did not receive much attention in the past. The by now classical result is the Koksma–Hlawka error bound, see, e.g., [15].

Theorem 5. *If f has bounded variation $V(f)$ on $[0,1]^d$ in the sense of Hardy and Krause, then, for any $\mathbf{y}^{(1)},\ldots,\mathbf{y}^{(N)} \in [0,1]^d$, we have*

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

Also this bound is the product of a measure of the quality of the points used, the *discrepancy* D_N^*, and a factor that depends on the function, the variation $V(f)$. Computational experience confirms that discrepancy is a good indicator of quasi-Monte Carlo performance, while variation is not a typical indicator. Furthermore, one should keep in mind that this bound is not easily computable. The Koksma–Hlawka inequality is a worst-case bound.

The discrepancy typically contains a term $(\log N)^d/N$. For very large N, the Koksma–Hlawka bound nicely goes down. However, it only starts to go down if $N > \exp(d)$, and so one did not expect these methods to be applicable for, say $d = 30$, because then the number of points should be larger than 10^{13}, which will continue to be unpractical for many years to come.

Thanks to some successes in financial applications in the last years of the previous century, quasi-Monte Carlo methods nowadays receive the attention they deserve. The standard example used to show that quasi-Monte Carlo methods can work very well in high-dimensional practical problems is a 360-dimensional integral coming from finance.

One distinguishes to classes: (1) lattice rules and (2) methods based on low-discrepancy sequences. Lattice rules are designed especially for periodic functions. So far quasi-Monte Carlo methods are derived for integrals over a unit cube with constant weight function.

The many activities in this area in recent years become obvious if one looks at the many recent surveys, e.g., [15,20,22,10,2]. To have an impression of what goes on nowadays in the research community, I invite the reader to have a look at some recent conference proceedings, e.g., [16,11].

The direct application of quasi-Monte Carlo methods is nowadays limited to integrals on hypercubes. These methods consider each dimension of a problem to be of equal importance, i.e., they are designed for isotropic problems. High-dimensional practical problems tend to be anisotropic however.

Because of the unexpected successes in some applications, research on quasi-Monte Carlo methods now concentrates on the fast generation of very high-dimensional low-discrepancy sequences and on transferring techniques originally designed to improve Monte Carlo methods. Searching for explanations why these methods work unexpectedly well for some classes of practical problems, is also a research topic.

Practical error estimation is nowadays based on ideas from Monte Carlo methods: one includes randomisations. A lot of effort will be devoted in the near future to obtain other (faster and deterministic) error estimators. Even more effort will be devoted to developing variants of these methods for anisotropic problems and for other regions than hypercubes. What is needed are adaptive algorithms that detect the anisotropy in a given problem and exploit this. This feature is available in current automatic integration routines for low-dimensional integration but remains a challenge for, e.g., 30-dimensional problems. It is a challenge researchers should accept.

5. Final remarks

We observe a stagnation of progress in the area of constructing cubature formulas exact for polynomials. During the past five years almost no theoretical progress was made and only a few new cubature formulas were constructed. There is still work to do here! Surprisingly, almost no attention has been paid to the construction of cubature formulas for four- and higher-dimensional regions. Consequently, the available adaptive software uses cubature formulas which are probably much more expensive than really necessary. New cubature formulas will probably extend Range I.

For lower dimensions a reasonable collection of techniques is available and combined with the growing speed of computers, adaptive software will evolve to more complex algorithms. Several strategies will be embedded in the same piece of software. The software will try them all and will proceed with the most promising ones until the requested accuracy is attained.

For higher dimensions, I expect a lot will happen in the near future, but unfortunately my crystal ball does not give a clear view on the outcome of this.

References

[1] M. Abramowitz, I.A. Stegun (Eds.), Handbook of Mathematical Functions with Formulas, Graphs and Mathematical Tables, National Bureau of Standards Applied Mathematics Series, Vol. 55, U.S. Government Printing Office, Washington, DC, 1964.
[2] R.E. Caflisch, Monte Carlo and Quasi-Monte Carlo Methods, Acta Numerica, Vol. 7, Cambridge University Press, Cambridge, 1998, pp. 1–49.
[3] R. Cools, Constructing Cubature Formulae: The Science Behind the Art, Acta Numerica, Vol. 6, Cambridge University Press, Cambridge, 1997, pp. 1–54.
[4] R. Cools, Monomial cubature rules since "Stroud": a compilation—part 2, J. Comput. Appl. Math. 112 (1–2) (1999) 21–27.
[5] R. Cools, A. Haegemans, CUBPACK: a package for automatic cubature; framework description, submitted for publication.
[6] R. Cools, I.P. Mysovskikh, H.J. Schmid, Cubature formulae and orthogonal polynomials, J. Comput. Appl. Math. 127 (2001) 121–152.
[7] R. Cools, P. Rabinowitz, Monomial cubature rules since 'Stroud': a compilation, J. Comput. Appl. Math. 48 (1993) 309–326.
[8] P.J. Davis, P. Rabinowitz, Methods of Numerical Integration, Academic Press, London, 1984.
[9] E. de Doncker, I. Robinson, TRIEX: integration over a TRIangle using nonlinear EXtrapolation, ACM Trans. Math. Software 10 (1984) 17–22.
[10] M. Drmota, R.F. Tichy, Sequences, Discrepancies and Applications, Lecture Notes in Mathematics, Vol. 1651, Springer, Berlin, 1997.
[11] K.-T. Fang, F. Hickernell, H. Niederreiter (Eds.), Monte Carlo and quasi-Monte Carlo methods 2000, Proceedings of the Fourth International Conference on Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing, Springer, Berlin, 2002.
[12] D.W. Lozier, B.R. Miller and B.V. Saunders, Design of a digital mathematical library for science, technology and education, Proceedings of the IEEE Forum on Research and Technology Advances in Digital Libraries; IEEE ADL 1999, Baltimore, MD, May 19, 1999.
[13] H.M. Möller, Polynomideale und Kubaturformeln, Ph.D. Thesis, Universität Dortmund, 1973.
[14] I.P. Mysovskikh, Interpolatory Cubature Formulas, Izdat. Nauka, Moscow, 1981 (Russian).
[15] H. Niederreiter, Random Number Generation and Quasi-Monte Carlo Methods, CBMS-NSF Regional Conference Series in Applied Mathematics, Vol. 63, SIAM, Philadelphia, 1992.
[16] H. Niederreiter, J. Spanier (Eds.), Monte Carlo and quasi-Monte Carlo methods 1998, Proceedings of the Third International Conference on Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing, Springer, Berlin, 2000.
[17] J. Radon, Zur mechanischen Kubatur, Monatsh. Math. 52 (1948) 286–300.
[18] H.J. Schmid, Interpolatorische Kubaturformeln, Dissertationes Math., Vol. CCXX, Polish Scientific Publishers, Warszawa, 1983.
[19] H.J. Schmid, Two-dimensional minimal cubature formulas and matrix equations, SIAM J. Matrix Anal. 16 (3) (1995) 898–921.
[20] I.H. Sloan, S. Joe, Lattice Methods for Multiple Integration, Oxford University Press, Oxford, 1994.
[21] A.H. Stroud, Approximate Calculation of Multiple Integrals, Prentice-Hall, Englewood Cliffs, NJ, 1971.
[22] S. Tezuka, Uniform Random Numbers: Theory and Practice, Kluwer Academic Publishers, Boston, 1995.