The approximation of low-dimensional integrals: available tools and trends

Ronald Cools

Report TW 259, May 1997

Katholieke Universiteit Leuven
Department of Computer Science
Celestijnenlaan 200A – B-3001 Heverlee (Belgium)

The approximation of low-dimensional integrals: available tools and trends

Ronald Cools

Report TW 259, May 1997

Department of Computer Science, K.U.Leuven

Abstract

This text describes several methods to approximate multivariate integrals. Cubature formulae that are exact for a space of polynomials and Monte Carlo methods are the best known. More recently developed methods such as quasi-Monte Carlo methods (including lattice rules), Smolyak rules and stochastic integration rules are also described.

This short note describes the contents of a session keynote talk at the 15th IMACS World Congress on Scientific Computation, Modelling and Applied Mathematics, August 24–29, 1997 in Berlin.

The approximation of low-dimensional integrals: available tools and trends

Ronald Cools
Dept. of Computer Science, Katholieke Universiteit Leuven
Celestijnenlaan 200A, B-3001 Heverlee, Belgium
e-mail: Ronald.Cools@cs.kuleuven.ac.be

Keywords: quadrature, cubature, integration rule, quasi-Monte Carlo, lattice rule

ABSTRACT

This text describes several methods to approximate multivariate integrals. Cubature formulae that are exact for a space of polynomials and Monte Carlo methods are the best known. More recently developed methods such as quasi-Monte Carlo methods (including lattice rules), Smolyak rules and stochastic integration rules are also described.

1 Introduction

The problem of measuring areas and volumes has always been present in everyday life. The ancient Babylonians and Egyptians already had rules for finding the areas of triangles, trapezoids and circles and the volumes of parallelepipeds, pyramids and cylinders. They thought of these figures in concrete terms, mainly as storage containers for grain, and discovered these rules empirically.

The start of the modern study of volume computation is usually linked with Kepler, in the 17th century. His motivation also had to do with storage containers: wine barrels.

The formulation of the problem of measuring in terms of integrals and functions is more recent. In fact, the computation of areas and volumes motivated the field of integration. The calculation of integrals is nowadays one of the basic operations in analysis and is needed in many technical computations. In practical situations, many integrations have to be performed numerically because the analytic evaluation is too complicated or even impossible.

We will now formulate the problem in today's terminology. An integral I is a linear continuous functional

$$I[f] := \int_\Omega w(\mathbf{x}) f(\mathbf{x}) d\mathbf{x}, \text{ where the region } \Omega \subset \mathbb{R}^n.$$

We use $\mathbf{x}$ as a shorthand for the variables $x_1, x_2, \ldots, x_n$. We will always assume that $w(\mathbf{x}) \geq 0$, for all $\mathbf{x} \in \Omega$, that is, I is a positive functional.

It is often desired to approximate I by a weighted sum of function values such that

$$I[f] \simeq Q[f] := \sum_{j=1}^{N} w_j f(\mathbf{x}_j) \text{ where } w_j \in \mathbb{R} \text{ and } \mathbf{x}_j \in \mathbb{R}^n. \tag{1}$$

It is desirable that all $\mathbf{x}_j \in \Omega$ (to be certain $f(\mathbf{x}_j)$ is defined) and $w_j > 0$ (to avoid cancellation). If Ω is an interval, i.e., $n = 1$, then the rule Q (1) is called a *quadrature formula.* If $n \geq 2$, it is called a *cubature formula.* The points $\mathbf{x}_j$ and weights w_j are chosen independent of the function f. They are chosen such that the cubature formula is useful for a particular class of functions.

There are several criteria to specify and classify cubature formulae based on their behaviour for specific classes of functions. Depending on the criteria used, one can distinguish several classes of methods. In the following section, several classes will be briefly surveyed. They all have in common that they are exact for the constant function: $I[1] = Q[1] = \mathrm{vol}(\Omega)$.

2 Some classes of cubature formulae

2.1 Cubature formulae of algebraic degree

A first approach to constructing cubature formulae is to consider the multivariate integral as a repeated integral and use repeated (1-dimensional) quadrature formulae to approximate it. This approach is inviting because it is easy and a lot of quadrature formulae are known, for all types of weight functions, of different algebraic degree of precision, i.e., the formulae are exact for all polynomials up to a certain degree. These so-called *product rules* can be constructed for the n-dimensional cube, simplex and sphere [19]. A Gauss quadrature formula of degree $d = 2k + 1$ for the unit interval with constant weight function requires $k + 1$ points. The corresponding product formula for the n-cube of the same degree requires $(k + 1)^n$ points. This type of rules is expensive because they also give the exact result for a lot of multivariate monomials of degree higher than d.

Instead of treating the multivariate integral as a repeated integral, one can deal directly with it and search for cubature formulae that are of degree d. The number of points in such formulae for the n-cube, n-simplex, etc. is $N = \binom{n+k}{n} + \mathcal{O}(k^{n-1})$ with $k = \lfloor d/2 \rfloor$. Such formulae are however very hard to construct. Almost all published cubature formulae are for low dimensions (≤ 5). Only a few are known for higher dimensions. If the degree or the dimension increases, the $\mathcal{O}(k^{n-1})$ becomes more and more significant. The surface of the sphere is an important region of integration that requires special attention: it does not have interior points. This property influences the above estimate for the number of points. For a survey of available cubature formulae, see [19, 6]. For more on this type of integration rules, see [4].

If a cubature formula does not give a result that is accurate enough, instead of increasing the degree one can often divide the region in smaller regions and apply a cubature formula to each of the subregions. The sum of all cubature formulae on all subregions is a so-called *compound rule*. One can continue this process until one obtains the desired accuracy. An interesting compound rule for regions that can be subdivided in regions of the same shape, is the so-called *copy rule*. If, for example, the integration region is a square, one can divide this into m^2 identical squares, each of side $1/m$th the original side, and apply a properly scaled version of the given cubature formula to each. This approach looks expensive, especially if the dimension goes up, but is appealing because an error expansion is readily available. For regular $f(x, y) \in C^p$, $p \in \mathbb{N}$, the extension of the one-dimensional Euler–Maclaurin expansion may be expressed as

$$Q^{(m)}[f] - I[f] = \sum_{i=1}^{p-1} \frac{B_i(Q, f)}{m^i} + \mathcal{O}(m^{-p}), \tag{2}$$

where $Q^{(m)}$ is the m^2-copy of Q and the coefficients B_i depend on the cubature formula Q, the integral I and the integrand f. Once it is known that an error expansion exists, Richardson extrapolation can be used to speed up convergence. The m-copy rules for cubes and simplices have received considerable attention, because for some classes of non-regular functions, error expansions are also available and thus extrapolation can be used to speed up convergence. For more information and additional references, see [12, 13, 21].

Cubature formulae of algebraic degree form the basis of so-called *automatic integration routines*. The aspired ideal is "to relieve the person who has to compute an integral of any need to think" [7]. Such routines are meant to be used as black boxes: a user gives an integration region, integrand and a desired accuracy and after some number crunching an approximation of the integral is returned hopefully satisfying the error request. Nowadays, such routines are global adaptive. The given region is subdivided in such a way that the subregions are smaller in those parts of the given region where the integrand differs more from the polynomials for which the basic cubature formulae in the routine give the exact result. At each step the subregion with the largest error is selected for further processing. The goal is to make the global error smaller than the error request. Most automatic integrators available are developed for the triangle, the tetrahedron and the n-cube. If one has an integral over another region, one has to transform it to one of these regions before one can call the routine to help. At least for 2 dimensions, this is changing. Cubpack++ goes as far as we can toward the ideal [5]. It allows a user to approximate integrals over a large variety of 2-dimensional regions. Other automatic integrators for 2 dimensions are mentioned in [5]. For the tetrahedron see [1] and for the n-cube see [2].

2.2 Smolyak rules

Smolyak's method recently received a lot of attention, about 30 years after his paper appeared; see, e.g., [15, 22]. Roughly speaking, a subset of the points of a product rule is used.

Assume that for the interval $[-1, 1]$ a sequence of quadrature formulae $U_1, U_2, \ldots$ is available. For dimensions $n > 1$ and $\mathbf{i} = (i_1, \ldots, i_n) \in \mathbb{N}^n$ the tensor product formula $U_{i_1} \otimes \cdots \otimes U_{i_n}$ is defined as a usual product rule for the n-cube $[-1, 1]^n$. The Smolyak formulae are given by

$$A(q, n) := \sum_{q-n+1 \leq |\mathbf{i}| \leq q} (-1)^{q-|\mathbf{i}|} \binom{n-1}{q-|\mathbf{i}|} (U_{i_1} \otimes \cdots \otimes U_{i_n}), \quad \text{with} \quad q \geq n.$$

They are linear combinations of the product formulae $U_{i_1} \otimes \cdots \otimes U_{i_n}$ such that only tensor products with a relatively small number of points are used and the linear combination preserves the interpolation property of the quadrature rules for $n > 1$.

One can formulate restrictions on the quadrature rules U_i such that the corresponding Smolyak rule is exact for all polynomials of degree at most d. Most studied are Smolyak rules that are based on Clenshaw-Curtis quadrature points. For large dimensions n and fixed degree $d = 2k + 1$, the number of points in such a cubature formulae $N \approx 2^k n^k / k!$. This is comparable with the cost of known cubature formulae for the unit cube. The Smolyak rules can however be constructed for all degrees of precision.

This method looks promising if the dimension of the integral is reasonable. Practical experience is however still limited and only experimental implementations exist.

2.3 Monte Carlo methods

The need to approximate multi-dimensional integrals at a reasonable cost, lead to the development of Monte Carlo methods. An integral can be seen as the expected value of a certain stochastic process. The expected value is estimated by sampling. The basic form of the Monte Carlo method is

$$I[f] := \int_\Omega f(\mathbf{x})d\mathbf{x} \simeq Q[f] := \frac{\mathrm{vol}(\Omega)}{N}\sum_{j=1}^{N} f(\mathbf{x}_j) \tag{3}$$

where the points $\mathbf{x}_j$ are chosen independently uniformly distributed in Ω. The $Q[f]$ in (3) is a random variable with expected value $I[f]$. Behind this computation is the Law of Large Numbers. From the Central Limit Theorem, one can derive a confidence interval for the approximation $Q[f]$. For a fixed level of confidence λ, the error bound is $\lambda\sigma/\sqrt{N}$, where σ^2 is the variance of $f(\mathbf{x})$ on Ω. Note that the rate of convergence does not depend on the dimension. See, e.g., [18] for an introduction.

In practice one does not use truly random numbers. One uses so-called *random number generators* that produce a deterministic sequence of so-called *pseudorandom numbers* that pass a number of statistical tests for near-randomness. Such generators are so easily available that one uses them without thinking, although they are a very important aspect of all Monte Carlo computations. It is known nowadays by a small number of experts in this matter that some widely available generators have structural defects that might influence some simulations. For more information we refer to [11, 14, 20].

The variance of the integrand plays an important role in this procedure. In practice, the usefulness of Monte Carlo methods largely depends on variance reduction techniques. Many techniques exist, e.g., stratified sampling, importance sampling, antithetic variates. The idea behind all these is to take the shape of the integrand into account to determine the sample points.

Importance sampling is based on the following idea. First, the given integral (3) is rewritten as

$$I[f] = \int_\Omega \frac{f(\mathbf{x})}{p(\mathbf{x})}p(\mathbf{x})d\mathbf{x} \;\text{ with }\; p(\mathbf{x}) > 0, \forall \mathbf{x} \in \Omega \;\text{ and }\; \int_\Omega p(\mathbf{x})d\mathbf{x} = 1.$$

Then $p(\mathbf{x})$ is used as a probability density function and the random points are selected according to $p(\mathbf{x})$. The Monte Carlo approximation is then

$$Q[f] = \frac{\mathrm{vol}(\Omega)}{N}\sum_{j=1}^{N}\frac{f(\mathbf{x}_j)}{p(\mathbf{x}_j)}.$$

The better $p(\mathbf{x})$ approximates the shape of $f(\mathbf{x})$, i.e., the more $f(\mathbf{x})/p(\mathbf{x})$ looks like a constant, the smaller the variance and thus the error bound.

Stratified sampling splits the integration region and estimates the contribution of each subregion using Monte Carlo integration. One can take smaller subregions where the function differs more from a constant. This approach ensures that in each subregion a given number of samples are taken and that reduces the variance. Recursive stratified sampling is a technique that subdivides the region adaptively. This effort to make an automatic integration routine based on the Monte Carlo methods is described in [16].

2.4 Quasi-Monte Carlo methods

The error estimate of the Monte Carlo method is only probabilistic. In quasi-Monte Carlo methods the equal-weights formula (3) is still used, but the sequence of points are chosen to be "better than random". In addition rigourous error bounds are obtained that behave better than the $N^{1/2}$ law.

Equidistributed or *uniformly distributed* sequences are the crucial concept in quasi-Monte Carlo methods. The fraction of points of an equidistributed sequence of numbers that lie in any interval is asymptotically proportional to the length of the interval. Important sequences are due to Halton, Sobol and Faure and the generalisations of these by Niederreiter and Tezuka [14, 20]. The notion of *discrepancy* is used to measure the irregularity of a point set in the unit n-cube. The Koksma-Hlawka inequality says the error in a quasi-Monte Carlo method (3) is bounded by the star-discrepancy of the points times $V(f)$, the variation of f in the sense of Hardy and Krause. For specific sequences (a bound for) the discrepancy is known; e.g., for the Halton sequence this gives the error bound

$$|I[f] - Q[f]| \leq C\frac{(\log N)^n}{N}V(f)$$

where C is a constant independent of f.

Quasi-Monte Carlo methods are finding their way into applications. The recent hype around high dimensional integrals in risk analysis of financial derivatives is built on experience with quasi-Monte Carlo methods; see, e.g., [3]. Other experiences suggest that for high dimensions they do not live up to the expectations.

Quasi-Monte Carlo methods are thoroughly reviewed in [14].

2.5 Lattice rules

Lattice rules are a type of quasi-Monte Carlo method. They are designed for the integral over the unit cube $[0,1)^n$ in situations in which f is one-periodic with respect to each component of $\mathbf{x}$ and in addition is reasonable smooth. A lattice rule for the approximation of $I[f]$ is the average value of the integrand f over all points of an integration lattice L inside the cube $[0,1)^n$. An *integration lattice* is a discrete subset of $\mathbb{R}^n$ which is closed under addition and subtraction, and which contains $\mathbb{Z}^n$ as a subset. An example of a lattice rule for 1-dimensional integration is the rectangle rule. The oldest interesting generalisation is the so-called *method of good lattice points* which uses approximations of the form

$$Q[f] = \frac{1}{N}\sum_{j=0}^{N-1} f\left(\frac{j}{N}\mathbf{z}\right) \tag{4}$$

where N is an a priori chosen number of cubature points and $\mathbf{z}$ a carefully selected integer vector. The error of a lattice rule depends on the coefficients in the Fourier expansion of f that correspond to the points in the dual of the lattice L. The quality of a lattice rule thus depends on the dual lattice. The *Zaremba index* is the best known criteria to study this.

The lattice rule (4) is a so-called *rank-1* lattice rule: one sum only is needed to write the expression. If s sums are required, the rule is called a *rank-s* lattice rule. The product rectangle rule is a *rank-n* rule.

An implementation is given in [8, 9]. Practical error estimation is based on sequences of embedded lattice rules. If one wants to apply this to a non-periodic function, then one must force periodisation by nonlinear transformations. This does not only cause an overhead but it tends to transform nice integrands into nasty ones. Practical experience with lattices is still limited. It has been shown that they can already be very useful in very low, say two, dimensions.

The recommended starting point for all potential users is [17].

2.6 Stochastic integration rules

Cubature formulae of algebraic degree are hard to construct for high degrees and dimensions and an error estimate is not directly available. They can however deliver a high accuracy for smooth functions. Monte Carlo methods come with a simply-assessed error estimate but are normally only exact for constant functions. The idea of combining the good properties of each method is about 30 years old but recently resurfaced.

For the region $\Omega = [-1,1]^n$, instead of using the Monte Carlo method (3), one can use

$$Q[f] := \frac{2^n}{N} \sum_{j=1}^{N} Q_{\alpha_j}[f] \;\; \text{with, e.g.,} \;\; Q_{\alpha_j}[f] := \frac{f(\mathbf{x}_j) + f(-\mathbf{x}_j)}{2}. \tag{5}$$

The Q_{α_j} are here parameter dependent cubature formulae of degree 1. In contrast with the original Monte Carlo method, the so-called *stochastic integration rule* (5) is thus exact for all first degree polynomials. This approach equals the variance reduction technique known as *antithetic variates.*

Higher degree formulae Q_{α_j} can be used in (5). In order to obtain an unbiased estimate for $I[f]$ one has to be careful. In general the random samples have to be taken using special distribution functions and that causes practical difficulties. If the region of integration is the surface of a sphere, introducing randomisation is easy. It suffices to apply a rotation to the cubature formula to obtain another one of the same degree.

The recent work in this area aims at approximating integrals over n-space with a Gaussian (or related) weight function. The target applications originate in Bayesian statistics. For recent results and references, see [10].

References

[1] J. Berntsen, R. Cools, and T. O. Espelid, *Algorithm 720: An algorithm for adaptive cubature over a collection of 3-dimensional simplices*, ACM Trans. Math. Software 19, p. 320–332 (1993).

[2] J. Berntsen, T. O. Espelid, and A. Genz, *Algorithm 698: DCUHRE – an adaptive multidimensional integration routine for a vector of integrals*, ACM Trans. Math. Software 17, p. 452–456 (1991).

[3] J. Case, *Wall Street's dalliance with number theory*, SIAM News 28(10), p. 8–9 (1995).

[4] R. Cools, *Constructing cubature formulae: the science behind the art*, Acta Numerica 6, p. 1–54 (1997).

[5] R. Cools, D. Laurie, and L. Pluym, *Algorithm 764: Cubpack++: A C++ package for automatic two-dimensional cubature*, ACM Trans. Math. Software 23, p. 1–15 (1997).

[6] R. Cools and P. Rabinowitz, *Monomial cubature rules since 'Stroud': A compilation*, J. Comput. Appl. Math. 48, p. 309–326 (1993).

[7] P. J. Davis and P. Rabinowitz, *Methods of numerical integration*, Academic Press, London (1984).

[8] S. Joe and I. H. Sloan, *Implementation of a lattice method for numerical multiple integration*, ACM Trans. Math. Software 19, p. 523–545 (1993).

[9] S. Joe and I. H. Sloan, *Corrigendum: "Implementation of a lattice method for numerical multiple integration"*, ACM Trans. Math. Software 20, p. 245 (1994).

[10] A. Genz and J. Monahan, *Stochastic integration rules for infinite regions*, SIAM J. on Sci. Comp., To appear (1998). (See http://www.sci.wsu.edu/math/faculty/genz/papers/ranrul.ps.)

[11] P. L'Ecuyer, *Bad lattice structures for vectors of non-successive values produced by some linear recurrences*, INFORMS J. on Computing 9, To appear (1997).
(See http://www.iro.umontreal.ca/~lecuyer/myftp/papers/badlac.ps.)

[12] J. N. Lyness and R. Cools, *A survey of numerical cubature over triangles*, Mathematics of Computation 1943-1993: A Half-Century of Computational Mathematics (Providence, Rhode Island) (W. Gautschi, ed.), Proceedings of Symposia in Applied Mathematics, vol. 48, American Mathematical Society, p. 127–150 (1994).

[13] J. N. Lyness and E. de Doncker, *Quadrature error expansions. II. The full corner singularity*, Numer. Math. 64, p. 355–370 (1993).

[14] H. Niederreiter, *Random number generation and quasi-Monte Carlo methods*, CBMS-NSF regional conference series in applied mathematics, vol. 63, SIAM, Philadelphia (1992).

[15] E. Novak and K. Ritter, *High dimensional integration of smooth functions over cubes*, Numer. Math. 75, p. 79–97 (1996).

[16] W. H. Press and G. R. Farrar, *Recursive stratified sampling for multidimensional Monte Carlo integration*, Comput. Phys. 4, p. 190–195 (1990).

[17] I. H. Sloan and S. Joe, *Lattice methods for multiple integration*, Oxford University Press (1994).

[18] I. M. Sobol, *A primer for the Monte Carlo method*, CRC Press (1994).

[19] A. H. Stroud, *Approximate calculation of multiple integrals*, Prentice-Hall, Englewood Cliffs, N.J. (1971).

[20] S. Tezuka, *Uniform random numbers: theory and practice*, Kluwer Academic Publishers, Boston (1995).

[21] P. Verlinden and A. Haegemans, *An error expansion for cubature with an integrand with homogeneous boundary singularities*, Numer. Math. 65, p. 383–406 (1993).

[22] G. W. Wasilkowski and H. Woźniakowski, *Explicit cost bounds of algorithms for multivariate tensor product problems*, J. of Complexity 11, p. 1–56 (1995).