Handling accuracy in Bayesian automatic adaptive quadrature

Gh Adam¹,² and S Adam¹,²

¹ Laboratory of Information Technologies, Joint Institute for Nuclear Research, 6 Joliot Curie St., 141980 Dubna, Moscow Region, Russia
² Horia Hulubei National Institute for Physics and Nuclear Engineering (IFIN-HH), 30 Reactorului St., Magurele - Bucharest, 077125, Romania
E-mail: adamg@jinr.ru

Abstract. New Bayesian inferences which significantly extend the coverage of the Bayesian automatic adaptive quadrature (BAAQ) approach to the solution of Riemann integrals are reported. The scrutiny of the possible floating-point machine number approximations of abscissa values inside an integration domain unveiled the occurrence of five classes of integration domains entering the quadrature problems: zero-length, open void, microscopic, mesoscopic, macroscopic. Correct approach to the class identifications and class adapted advancement to the solution are described. In the most complex, macroscopic case, the reversible addition of a new decision branch extends the BAAQ approach coverage to both difficult and easy integrals. The reliability of the code used for solving easy integrals was enhanced with four new analysis tools as compared to the standard automatic adaptive quadrature solution.

1. Introduction

We describe new progress toward the derivation and implementation of the Bayesian automatic adaptive quadrature (BAAQ) [1, 2] as a robust, reliable, fast, and highly accurate computational tool of interest to the modeling of physical phenomena within numerical experiments asking for the evaluation of large numbers of Riemann integrals by numerical methods. Instances of such phenomena are, e.g., the phase transitions or processes involving fragmentation and fusion, where the behaviour of the system is characterized by the sudden change of an inner order parameter, which results in drastic and unpredictable modification of the mathematical properties of the integrands.

Such circumstances cannot be accommodated within the standard automatic adaptive quadrature (AAQ) approach to the numerical solution (see, e.g., dedicated chapters in the monographs [3, 4, 5]), due to the impossibility to decide in advance on the right choice of the convenient library procedure from an existing menu (see, e.g., [6]).

The presentation is structured in five sections: pros and cons of the standard AAQ (section 2), Bayesian inferences on the classes of integration domain lengths (section 3), integration domain adapted BAAQ algorithms (section 4), and conclusions (section 5).
2. Pros and cons of the automatic adaptive quadrature

2.1. The mathematical problem

We seek a numerical solution of the (proper or improper) Riemann integral

\[ I \equiv I[a, b]f = \int_a^b w(x)f(x)dx, \quad (1) \]

over an integration domain \([a, b] \subseteq \mathbb{R}\), for a real valued integrand function \(f : [a, b] \rightarrow \mathbb{R}\) which is almost continuous everywhere on \([a, b]\), such that (1) exists and is finite. The weight function \(w(x)\) either absorbs an analytically integrable difficult factor entering the integrand (e.g., an oscillatory or endpoint singular function), or else \(w(x) \equiv 1, \forall x \in [a, b]\).

2.2. Numerical quadrature

The numerical solution of (1) consists of a pair \(\{Q, E > 0\}\), where \(Q\) is an approximate value of \(I\), while \(E\) is an estimate of the error associated to \(Q\). The obtained output is asked to satisfy the accuracy requirement

\[ |I - Q| < E < \tau, \quad (2) \]

where the prescribed input accuracy \(\tau\) is defined in terms of two parameters: the absolute accuracy \(\varepsilon_a\) and the relative accuracy \(\varepsilon_r\), such that [3]

\[ \tau = \max\{\varepsilon_r|I|, \varepsilon_a\} \simeq \max\{\varepsilon_r|Q|, \varepsilon_a\}. \quad (3) \]

If the condition \(E < \tau\) is not fulfilled, the value of \(Q\) is refined and a new error estimate \(E\) is computed. If this process is converging, then (2) is eventually fulfilled and the computation is terminated.

2.3. Permanent features of the automatic adaptive quadrature

The computation scheme developed within the AAQ approach implements an integrand adapted discretization of \([a, b]\), which defines a partition of \([a, b]\),

\[ \Pi_N[a, b] \equiv \{a = x^0 < x^1 < \cdots < x^i < \cdots < x^N = b | N \geq 1\}. \quad (4) \]

Over each subrange \([x^{i-1}, x^i] \subseteq [a, b]\), a (possibly subrange dependent) local quadrature rule is used to get an approximate value \(q\) of \(I[x^{i-1}, x^i]f\), together with an associated error estimate \(e > 0\).

A partition dependent global pair solving (1), \(\{Q = Q_N, E = E_N > 0\}\), is got by summing up the individual outputs \(\{q, e > 0\}\) over the subranges of the partition (4). In what follows, in order to avoid cumbersome notations, we will consider a generic subrange \([\alpha, \beta] \subseteq [a, b]\), standing for any subrange \([x^{i-1}, x^i]\) of (4).

The number of subranges of (4) starts with \(N = 1\) and it is increased by gradual refinement of \(\Pi_N[a, b]\) until either the accuracy condition (2)–(3) is fulfilled, or a failure diagnostic is issued.

2.4. The standard automatic adaptive approach to numerical quadrature

The standard AAQ implements the refinement of \(\Pi_N[a, b]\) as a subrange binary tree the evolution of which is controlled by an associated priority queue. The binary tree initialization equates the root with the input integration domain \([a, b]\) over which a first global \(\{Q_1, E_1 > 0\}\) output is computed. If the termination criterion (2)–(3) is not fulfilled, then a recursive procedure is followed: the priority queue is activated, the resulting root is bisected, local estimates \(\{q, e > 0\}\) are computed over each resulting sibling, the global quantities \(\{Q_N, E_N > 0\}\) are updated, and the end of computation is checked again.
The standard subdivision scheme can be supplemented with a convergence acceleration algorithm if the occurrence of an integrand singularity was heuristically inferred.

The scrutiny of the existing AAQ implementations into program libraries (IMSL, NAG, SLATEC) or in separate modules (see the evidence reported in [3, 4, 5, 7, 8, 9, 10]) points to the existence of a wide variety of specific integrals (1) which are successfully solved by means of the existing standard AAQ codes.

Taking into account this circumstance, the BAAQ approach to the solution of (1) was upgraded by a decision branch enabling the computation of an enhanced AAQ (EAAQ) solution in cases where the simpler AAQ approach is sufficient to get quick fulfilment of the termination criterion (2). The most important improvements done within EAAQ with respect to the standard AAQ approach described in QUADPACK [3], which substantially increase reliability while keeping on the code fast, are listed below.

2.5. Ways of enhancing reliability of the AAQ codes

(i) A more reliable local error estimator within Clenshaw-Curtis quadrature.

Within the local quadrature rules, the assessment of the magnitude of the local quadrature error $e > 0$ uses probabilistic arguments and this feature naturally results in the dependence of the reliability of the local error output on the details of the integrand variation inside the current integration range.

Extensive numerical results [8] providing comparisons of three kinds of local error estimators for Clenshaw-Curtis (CC) numerical quadrature (CC-like [11] — see also [12, 13, 14], Sloan-Smith-like [15], and QUADPACK-like [3]) pointed to the existence of significantly different reliability levels of the three kinds of local error estimators. The most reliable one was found to be the CC-like error estimator involving unweighted contributions coming from the four highest order terms of the Chebyshev series expansion of the integrand over the current range. For this reason, our implementation of CC local quadrature rules uses namely this kind of local error estimator.

(ii) Handling the priority queue.

The already defined subranges in the partition (4) are divided into two classes: subranges bearing significant meaning $\{q, e > 0\}$ at output and subranges bearing conventional meaning $\{q = 0, e = \Omega\}$ at output, where $\Omega$ denotes a very large positive number close to the machine overflow. Being characterized by the highest possible local error estimate, a subrange carrying conventional meaning is put by the priority queue ordering algorithm at the beginning of the priority queue, such that it is bisected before any subrange bearing significant meaning at output.

(iii) Conditional activation of the global termination criterion.

The output brought by a subrange marked as conventional is useless for the update of the global pair $\{Q_N, E_N\}$. Therefore, in this case $Q_N$ and $E_N$ are left to their previous values and the check of the global termination criterion (2) is not attempted. An easy implementation of this feature was obtained by keeping record of the number of subranges carrying conventional meaning and by attempting the check of (2) only provided this number reaches the floor value zero.

(iv) Stabilization of the local quadrature sums as Riemann sums.

The assignment of significant or conventional meanings to the local quadrature rule outputs $\{q, e > 0\}$ is easily implemented by means of two criteria checking the stabilization of the local quadrature sums as Riemann sums.

The first criterion is implemented inside the procedures devoted to the computation of the local quadrature rule outputs. It checks, for each sibling generated by the bisection of the current root node of the subrange binary tree, the relationship inside the pair $\{q, e > 0\}$. 
The condition
\[ |q| > \varphi_{\text{min}} \cdot e, \quad \varphi_{\text{min}} = 2^{-8}, \]  
(5)
rejects the computed \( \{q, e\} \) values and enforces the assignment of conventional output for
the current sibling because the estimated accuracy of the computed local quadrature sum
stays below the two accurate significant figure threshold requirement.

The second criterion is enforced inside the global control procedure. It checks the
relationship between the output \( \{q_p, e_p\} \), carried by the parent subrange which was brought
to the root position in the subrange binary tree by the priority queue ordering, and the
summed outputs \( \{q_d, e_d\} \) coming from its two descendant subranges. If the variation
of the quadrature sum outputs exceeds the summed local errors (where a value \( e_p = 0 \) is assigned
if the parent subrange carries conventional meaning),
\[ |q_p - q_d| > e_p + e_d, \]  
(6)
then conventional meanings are assigned to both descendants.

The fulfilment of one or both criteria entails corresponding modification of the number of
subranges recorded as carrying conventional meaning.

3. Bayesian inferences on the classes of integration domain lengths

3.1. Critical issues of the Bayesian automatic adaptive quadrature

While the reliability of the EAAQ code was significantly enhanced in comparison with those
of the standard AAQ codes, there remained critical issues which could not be solved within it.
Thus, in the overwhelming part of the practical situations, the refinement of the partition (4) by
subrange bisection cannot enforce the location of the inner discontinuity points of the integrand
at the inner abscissas of \( \Pi_N[a,b] \) and to secure in this way fast convergence of the quadrature
sums or of the convergence acceleration algorithms. As a consequence, in such a cases, the
derivation of outputs of the local quadrature sums such as to meet the termination condition (2)
is well-known to be highly ineffective. Even worse, in most cases, the computation ends with
failure.

The BAAQ approach tries to overcome this difficulty by resolving the location of the offending
abscissas to machine accuracy and including the obtained results into the inner abscissas entering
the partition \( \Pi_N[a,b] \). We have learned how to solve efficiently such problems [16, 17].

During the development of the corresponding code, we had to learn about the conformity
of the hardware and software environment at hand with the IEEE 754 standard (1985 version
or 2008 revision) which governs the binary floating-point arithmetic (number formats, basic
operations, conversions, exceptional conditions).

Subsequent investigations have unveiled the dependence of the degree of precision of the
floating-point algorithms on the position and extension of the integration domain on the real
number axis. This topics deserves special consideration, since it provides the key to the consistent
implementation of BAAQ algorithms over all possible integration domains \([a,b]\).

3.2. Forward floating-point degree of precision of a quadrature sum [18]

The derivation of the local quadrature rule outputs \( \{q, e > 0\} \) within the existing AAQ
algorithms is done by use of interpolatory quadrature sums. An interpolatory quadrature sum
of the algebraic degree of precision \( d \) solves exactly the integrals over the fundamental power set
\( \{1, x, x^2, \ldots, x^d\} \).

Under computations done over \( \mathbb{R} \), the field of real numbers, \( d \) is a characteristic invariant of
the given interpolatory quadrature sum: the integral over the probe integrand \( \pi_d(x) = \sum_{l=0}^{d} x^l \)
is solved exactly such that the individual monomial contributions to the output are distinct
from each other and non-negligible irrespective on the extent and localization of the integration domain \([a,b]\) over the real axis.

However, under floating-point computations on a computer using \(t\)-bit length of the significand approximation of the numbers within the continuous field \(\mathbb{R}\), the quantity \(d\) ceases to be an invariant of the interpolatory quadrature sum. Nevertheless, an integration domain dependent probe integrand \(\pi_{dp}(x) = \sum_{i=0}^{d_{fp}} x^i\) can be defined for which non-negligible, distinguishable from each other, individual monomial contributions to the computed output are present.

The formal definition of the floating-point degree of precision, \(d_{fp} = d_{fp}[a,b]\), was done in [18]. This definition asks for the preservation of the non-negligibility of the individual monomial contributions to \(I[a,b]\pi_{dp}\) at both ends of the power series set \(\{1, x, x^2, \ldots, x^{d_{fp}}\}\), \(d_{fp} \leq d\).

For the implementation of the BAAQ algorithm over all possible scales of the integration domain lengths, the fact that a lowest degree subset of monomials entering \(\pi_{dp}(x)\) might result in negligible contributions to the quadrature sum is irrelevant. The essential feature is the preservation of the non-negligibility of the monomial contributions at the high degree end of the power series set. This results in the following definition of the forward floating-point degree of precision of an interpolatory quadrature sum:

1. Let \(I[\alpha, \beta]f\) denote the input integral of interest, defined over a finite integration range \([\alpha, \beta] \subset \mathbb{R}\).

2. Let \(q[\alpha, \beta]f\) denote an interpolatory quadrature sum of the algebraic degree of precision \(d\), which solves \(I[\alpha, \beta]f\) through floating point computations over a set of machine numbers characterized by a \(t\)-bit significand.

3. Let \(fl(a)\) denote the floating point approximation of \(a \in \mathbb{R}\), and let

\[
X = \max\{fl(|\alpha|), fl(|\beta|)\}, \quad X > 0, \quad \rho = fl(|\beta - \alpha|/X), \quad 0 < \rho \leq 2.
\]  

4. If \(\xi > 0\) stands for either \(X\) or \(\rho\), we define

\[
d_\xi = \begin{cases} 
\frac{d}{\lfloor \ln \varepsilon_0 / \ln \xi \rfloor} & \text{iff } \xi \geq x_m \\
\lfloor \ln \varepsilon_0 / \ln \xi \rfloor & \text{iff } \xi < x_m,
\end{cases}
\]  

where \(x_m = fl(\varepsilon_0^{1/d})\), \(\varepsilon_0 = 2^{-t}\), and \([a]\) is the ceiling of \(fl(a)\).

5. Then the floating point degree of precision, \(d_{fp}\), associated to \(q[\alpha, \beta]f\) is the positive integer

\[
d_{fp} = \min\{d_X, d_\rho\},
\]  

with \(d_X\) and \(d_\rho\) computed from Eq. (8) for the terms \(X\) and \(\rho\) of the pair (7).

3.3. Features of the machine number distributions inside integration domains

The floating-point degree of precision is a straightforward consequence of the floating-point representation of the infinite continuous set of the real numbers by a finite set of machine numbers, each of which being characterized by a fixed length of the significand and a finite range of variation of the characteristic [19, 20].

The distribution of the set of non-negative machine numbers is discrete and highly non-uniform: denser toward zero and scarcer toward the largest value machine number end. A similar property holds true for the set of non-positive machine numbers.

In the monographs devoted to the AAQ, [3, 4, 5], it is, however, assumed that there is a continuous, uniform distribution of the possible values of the argument of the integrand inside the integration domain, such that we can always choose convenient sets of distinct quadrature
knots at which the integrand is interpolated by a polynomial for which a quadrature sum is derived.

The need of abandoning the algebraic degree of precision in favour of the floating-point degree of precision is at variance with this assumption. A consistent and economic choice of the interpolatory quadrature sums for BAAQ solution is to make use of the available floating-point degree of precision over the current integration interval.

Over large enough integration domains, the distances inbetween adjacent quadrature knots exceed the distances inbetween neighbouring machine numbers by many orders of magnitude such that the AAQ assumption looks justified.

However, if the length of the integration interval falls below some position dependent threshold (and results in a floating-point degree of precision less than two units), then the AAQ assumption obviously fails.

In the next subsection, we show that elementary intervals can be defined inside which the discrete set of inner machine numbers is nearly uniformly distributed.

3.4. Ranges of nearly uniform machine number distributions

For the sake of concreteness, we assume henceforth that the computations are done in binary64 format as defined by the IEEE 754-2008 standard.

Therefore, the physical length of the significand is 52 bits and, with the addition of the hidden bit, the relative error in the approximation of any real number sufficiently far from zero does not exceed \( \bar{\varepsilon}_0 = 2^{-53} \).

The minimum positive normal floating-point number corresponds to the underflow threshold \( u_0 = 2^{-1022} \).

Given \( u_0 \) and \( \bar{\varepsilon}_0 \), the ratio \( v_0 = u_0 / \bar{\varepsilon}_0 \) provides a floating-point interval around zero, \((-v_0, v_0)\), within which the normalized floating-point values are evenly spaced.

The relative distance between two neighbouring machine numbers is \( \varepsilon_0 = 2\bar{\varepsilon}_0 = 2^{-52} \) (see [1] for the experimental proof of a consistency criterion checking this value).

Given the floating-point number \( \eta_0 \notin (-v_0, v_0) \), lateral intervals \((\eta_0^-, \eta_0)\) and \((\eta_0, \eta_0^+)\) can be defined inside which the floating-point distance equates \( |\eta_0| \varepsilon_0 \).

In order to define how far from \( \eta_0 \) can stay \( \eta_0^- \) and \( \eta_0^+ \), the negligibility of second order effects is to be secured. The relative distance \( \varepsilon_d \) corresponding to this aim is found from the condition \( \varepsilon_d^2 < 2^{-68} \), where \( 64 + 1 = 65 \) denotes the number of the binary bits in the CPU floating-point unit and further three units were added to skip effects coming from the guard digits.

We get \( \varepsilon_d = 2^{-34} \), a result which shows that the 18 least significant binary bits can have only linear effects upon the floating point operations.

4. Integration domain adapted BAAQ algorithms

The results derived in the previous section allow the implementation of a BAAQ approach which selects local quadrature rules adapted to the characteristic feature of each integration interval \([\alpha, \beta] \subseteq [a, b] \).

A number of five distinct cases were identified. They are resolved in a hierarchical order following the description pattern given below.

1. Vanishing integration domain length
   The condition \( b \neq a \) is checked at the beginning of the computation.
   If \( b = a \) is found to hold, then a vanishing value of the integral with zero associated error are returned.

2. Void open integration interval \((a, b)\)
   If \(|b - a| < \max\{|a|, |b|, v_0\} \cdot \varepsilon_0 \), then the cardinality of the machine numbers inside \((a, b)\) is zero.
The code returns a warning message together with a result \( \{Q, E = |Q|\} \), where \( Q \) is computed using the trapeze quadrature sum for the ends of \([a, b]\).

(3) **Integration domain of microscopic length**

If \(|b - a| < \max\{|a|, |b|, v_0\} \cdot \varepsilon_d\), then there is a (nearly) uniform discrete distribution of machine numbers inside \([a, b]\).

Estimates of both \( Q \) and \( E \) are derived using composite trapeze quadrature sums.

(4) **Integration domain of mesoscopic length**

A value \( d_{fp} < 5 \) was found for the forward floating-point degree of precision.

The cardinality of the machine numbers inside \((a, b)\) exceeds \(2^{18} \approx 2.6 \times 10^5\). Composite Simpson, trapeze and centroid quadrature sum outputs are used to check the integrand conditioning and to compute estimates for \( Q \) and \( E \).

(5) **Integration domain of macroscopic length**

This is the case investigated within the standard AAQ. Most of the BAAQ developments have been described in [1].

The important addition described in the section 2.5 above is the inclusion of the possibility to solve easy integrals within the BAAQ approach.

The principal new BAAQ progress with respect to [1] was the derivation of local error estimates using the principle of redundancy.

The QUADPACK approach [3, 5], based on the use of two related to each other quadrature sums (such that the quadrature knots of the lower algebraic degree quadrature sum are a subset of the quadrature knots of the higher algebraic degree quadrature sum) was abandoned. The local error estimate was found from a pair of quadrature sums sharing a single quadrature knot, the centre \( \gamma = (\beta + \alpha)/2 \) of the current integration range \([\alpha, \beta]\), while the fractional reduced quadrature knots are statistically independent of each other.

The main quadrature sum output, \( q_{CC} \), was derived by means of CC-33, a closed 33-knot quadrature sum spanned by the set of reduced quadrature knots equating the extremal points of the 32-nd degree Chebyshev polynomial of the first kind over \([-1, 1]\). The resulting interpolatory polynomial is known to be close to the minimax polynomial. As a consequence, \( q_{CC} \) provides highly accurate solutions for the integrals involving smooth integrand functions inside the current integration range \([\alpha, \beta]\).

The algebraic degree of precision of CC-33 is \( d_{CC} = 32 \).

The auxiliary quadrature sum output, \( q_{GK} \), was derived by means of the Gauss-Kronrod GK10-21 of QUADPACK [3], an open 21-knot quadrature sum spanned at reduced Gauss-Kronrod quadrature knots.

The algebraic degree of precision of GK10-21 is \( d_{GK} = 31 \).

An immediate consequence follows: the accuracy predictions of the local error estimate are significantly higher than those of the conventional QUADPACK error estimates, where they were hindered by the less accurate quadrature sum entering the local error estimator.

After checking out the absence of the exceptional cases mentioned in [1], a local error estimator, equated to the modulus of the difference \(|q_{CC} - q_{GK}|\), was used.

If the termination criterion (2) was not fulfilled after the attempt to compute the integral over the whole integration domain \([a, b]\), then the relative error estimate was compared to the conventional accuracy threshold \( \tau_a = 2^{-11} \approx 0.49 \cdot 10^{-3} \).

Under disagreement smaller than \( \tau_a \), the integral was assumed to be ‘easy’ and its solution by means of EAAQ was attempted. The Bayesian inference that EAAQ is the code of choice is not definitive. It may be denied under refinement of the partition (4) and the Bayesian approach for solving ‘difficult’ integrals be activated again.

Under disagreement exceeding \( \tau_a \), the BAAQ procedure using the priority queue approach described in [1] was used.
5. Conclusions

The Bayesian inferences reported in this paper significantly extend the coverage of the BAAQ approach to the solution of Riemann integrals (1).

The scrutiny of the possible machine number approximations of abscissa values inside the input integration domain (partially based on the use of the floating-point degree of precision) unveiled the occurrence of five classes of integration domains entering the quadrature problems: zero-length, open void, microscopic, mesoscopic, and macroscopic. The mentioned classes point to different extensions of the integration domains, increasing from the first to the last. The order of class identification cannot be arbitrary: it precisely follows the present order of enumeration. Finally, each class needs specific advancement to the solution, based on the use of different sets of local quadrature sums optimally adapted to the derivation of the output.

In the most complex, macroscopic case, besides the already developed Bayesian solution for ‘difficult’ integrals (characterized by the occurrence of inner integrand discontinuities) [1], a new decision branch was proposed which allowed the inclusion of ‘easy’ integrals as well inside a same code.

The reliability of the EAAQ code used for solving ‘easy’ integrals contains four kinds of improvements which supersede the reliability of the standard AAQ codes [3, 5, 6] and allow the turn back to the full BAAQ code under the detection of a previously unnoticed difficulty.

Acknowledgments

The work was supported within the JINR topic 05-6-1119-2014/2016 and the Romania–LIT Hulubei-Mashcheryakov Programme, JINR Orders 94/17.02.2014, p.25, 95/17.02.2014, pp.76, 77, and 96/17.02.2014, pp. 86–89.

References

[1] Adam Gh and Adam S 2012 Bayesian automatic adaptive quadrature: An overview *Mathematical Modeling and Computational Science* LNCS vol 7125, ed Gh Adam, J Buša and M Hnatič (Heidelberg: Springer) pp 1–16
[2] Adam Gh and Adam S 2009 Principles of the Bayesian automatic adaptive quadrature *Numerical Methods and Programming: Advanced Computing* 10 391–7 http://num-meth.srcc.msu.ru
[3] Piessens R, deDoncker-Kapenga E, Überhuber C W and Kahaner D K 1983 *QUADPACK, a subroutine package for automatic integration* (Berlin: Springer Verlag)
[4] Davis P J and Rabinowitz P 1984 *Methods of Numerical Integration* Second edition (Orlando (Fla), USA: Academic Press)
[5] Krommer A R and Ueberhuber C W 1998 *Computational Integration* (Philadelphia: SIAM)
[6] *NAG Library Manual* Mark 23 section X02 http://www.nag.co.uk/numeric/FL/nagdoc_fl23/html/FRONMTATTER/manconts.html
[7] Lyness J N 1983 When not to use an automatic quadrature routine *SIAM Rev.* 25 63–87
[8] Adam Gh and Nobile A 1991 Product integration rules at Clenshaw-Curtis and related points: a robust implementation *IMA J. Numerical Analysis* 11 271–96
[9] Adam Gh 1993 Case studies in the numerical solution of oscillatory integrals *Rom. Journ. Phys.* 38 527–38
[10] Adam Gh, Adam S and Plakida N M 2003 Reliability conditions on quadrature algorithms *Comp. Phys. Comm.* 154 49–64
[11] Clenshaw C W and Curtis A R 1960 A method for numerical integration on an automatic computer *Numer. Math.* 2 197-205
[12] Filippi S 1964 Angenäherte Tschebyscheff-Approximation einer Stammfunktion-eine Modifikation des Verfahrens von Clenshaw und Curtis, *Numer. Math.* 6 320–8
[13] Piessens R and Branders M 1975 Algorithm 002. Computation of oscillating integrals *J. Comp. Appl. Math.* 1 153–64
[14] Hasegawa T, Torii T and Ninomiya I 1983 Generalized Chebyhev interpolation and its application to automatic quadrature *Math. Comp.* 41 537–53
[15] Sloan I H and Smith W E 1980 Product-integration with the Clenshaw–Curtis points: Implementation and error estimates *Numer. Math.* 34 387–401
[16] Adam Gh, Adam S, Țifrea A and Neacșu A 2006 Resolving thin boundary layers in numerical quadrature 
Romanian Reports in Physics 58 155–66
[17] Adam Gh and Adam S 2008 The boundary layer problem in Bayesian adaptive quadrature Physics of Particles and Nuclei Letters 5 269–73
[18] Adam S and Adam Gh 2012 Floating-point degree of precision in numerical quadrature Mathematical Modeling and Computational Science LNCS vol 7125, ed Gh Adam, J Bușa and M Hnatič (Heidelberg: Springer) pp 189–94
[19] Goldberg D 1991 What every computer scientist should know about floating-point arithmetic ACM Computing Surveys 23 5–48 http://www.validlab.com/goldberg/paper.pdf
[20] Kahan W 1997 Lecture notes on the status of IEEE standard 754 for binary floating-point arithmetic http://www.eecs.berkeley.edu/~wkahan/ieee754status/IEEE754.PDF