The evaluation of definite integrals by interval subdivision

By H. O'Hara and Francis J. Smith*

An algorithm is described for the efficient and reliable evaluation of badly behaved definite integrals to a prescribed accuracy by concentrating the abscissas near the regions of greatest irregularity in the integrand. This is achieved by subdividing the interval of integration and by using a combination of the 7-point Clenshaw–Curtis quadrature and the 9-point Romberg quadrature in each subinterval. We argue that our algorithm will nearly minimise the number of function evaluations needed to evaluate a badly behaved integral.

(Received September 1968)

1. Introduction

In a previous paper (O'Hara and Smith, 1968) we discussed the problem of the efficient evaluation of an integral

$$I = \int_a^b f(x)dx \qquad (1.1)$$

to a prescribed accuracy when $f(x)$ is well behaved and when we can choose the abscissas at any points in the finite closed interval $[a, b]$. We argued that the integral is best evaluated by a modification of the Clenshaw–Curtis method (Clenshaw and Curtis, 1960) provided that the coefficients in the Chebyshev expansion of the integrand fall off fast enough (which we used to define 'well behaved'). When the integrand is sufficiently badly behaved it is known (Ralston, 1965; Wright, 1966) that the integral is best evaluated by splitting the interval of integration and by using low order formulas to evaluate the integral over each subinterval. This was illustrated with an example in our previous paper. Another example is given in **Fig. 1** where the integrand has a discontinuous derivative.

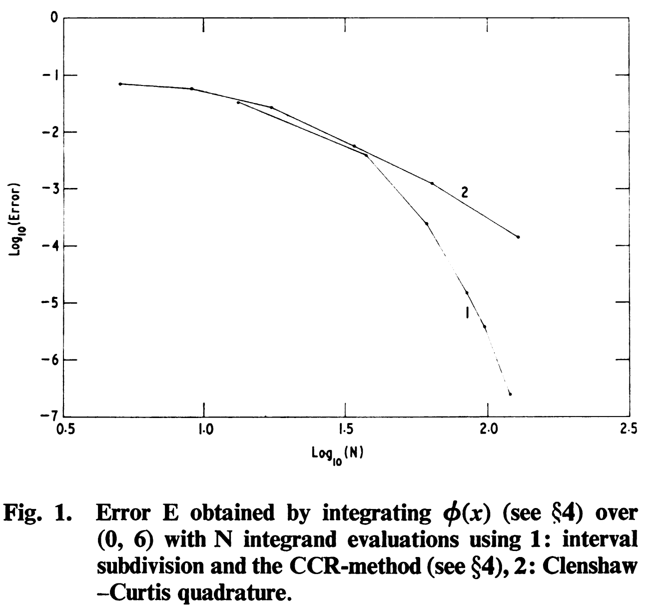


Fig. 1. Error E obtained by integrating $\phi(x)$ (see §4) over (0, 6) with N integrand evaluations using 1: interval subdivision and the CCR-method (see §4), 2: Clenshaw –Curtis quadrature.

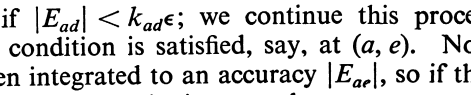
In this paper we describe a method for subdividing the interval which concentrates the abscissas near the regions of greatest irregularity, and we examine which quadrature should be used in each subinterval to evaluate the integral reliably with the minimum number of function evaluations. We assume that numerical values of $f'(x)$ are not available, that all singularities have been removed as far as possible by changes of variable, etc., and that it is known that $f(x)$ is sufficiently well behaved that the integral can be evaluated with at most a few thousand abscissas. For example, when it is known that $f(x)$ is liable to have sudden peaks, whose positions are unknown, with half-width, say, $(b - a)/10^5$, then the whole interval should first be subdivided into a set of 10^3 or 10^4 smaller intervals; otherwise the method we describe would be unreliable.

2. The algorithm

We consider first an algorithm for evaluating an integral to a prescribed absolute accuracy, ϵ. Sometimes a relative or percentage accuracy is required; this can be treated with a similar algorithm which we discuss briefly in the Appendix.

The basic feature of the algorithm is that the interval is broken up into subintervals; each subinterval is divided until the estimated error bound for the subinterval is less than the acceptable error. Then to make the algorithm as efficient as possible the difference between the error bound and the acceptable error is used to increase the acceptable errors in the remaining subintervals, keeping the sum of the absolute errors less than ϵ.

The main structure of the algorithm we propose is independent of the quadrature, I_{pq}, used to evaluate the integral over the interval (p, q). We let $|E_{pq}|$ denote a computable absolute error bound for this quadrature (assuming that there is one). We begin by calculating $|E_{ab}|$. If $|E_{ab}| < \epsilon$, the quadrature I_{ab}, is accepted; otherwise we bisect (a, b) at c. If $|E_{ac}| < k_{ac}\epsilon$, where k_{ac} is a constant less than one (we will assign k_{pq} a value later), then we accept I_{ac} as the integral over (a, c) and the interval (c, b) is considered. Otherwise we bisect (a, c) at d and



check if $|E_{ad}| < k_{ad}\epsilon$; we continue this process until such a condition is satisfied, say, at (a, e). Now (a, e) has been integrated to an accuracy $|E_{ae}|$, so if the whole interval (a, b) is to be integrated to an accuracy ϵ then

* *Departments of Computer Science and Applied Mathematics, The Queen's University of Belfast, N. Ireland*

the remaining interval (e, b) must be integrated to an accuracy $\epsilon_e = \epsilon - |E_{ae}|$. We therefore consider next the interval (e, d), and check if $|E_{ed}| < k_{ed}\epsilon_e$. Provided that the constants k_{pq} are chosen small enough for this process to converge we eventually obtain a value for the integral

$$I_{ab} = \sum_{p=a}^{q=b} I_{pq}$$

and an error

$$|E'_{ab}| = |\Sigma E_{pq}| \leqslant \Sigma |E_{pq}| \leqslant \epsilon. \qquad (2.1)$$

We considered several possible ways of choosing the constants k_{pq}, including some which were functions of the number of subintervals between q and b, (details will be given in a thesis of O'Hara, 1969), but in practice we found little difference between them. Those which were marginally more efficient occasionally did not converge; we therefore adopted the simple choice

$$k_{pq} = 0{\cdot}1 \text{ if } q \neq b \qquad (2.2)$$

and when $q = b$, k_{pq} must be set equal to unity to ensure that the inequality in (2.1) is satisfied. This gave convergence in all but a few rare cases, and in these cases $0{\cdot}1$ can be replaced by $0{\cdot}01$ or a smaller number to ensure convergence.

In the foregoing discussion we have assumed that $|E_{pq}|$ is a computable error bound for the quadrature I_{pq}. In practice it is only rarely that it is possible to compute a realistic bound. Usually we have to depend on a computable error estimate which is occasionally fallible (for example, by comparing two or more independent quadratures). If the interval is subdivided several times, however, the quadrature over the whole interval is very much more reliable than the quadrature over each subinterval. This follows from the first inequality in (2.1) and because $|E_{pq}|$ will be bigger than the actual error in I_{pq} in all but a very few cases if the error estimate $|E_{pq}|$ is reliable. This is verified in the results we discuss later.

3. Low-order quadrature

A wide range of low order quadratures can be used to evaluate the integrals over each subinterval, but most of them are unsuitable because all or nearly all previous function evaluations are lost each time an interval has to be divided. Hence all of the Gaussian quadratures and the wide range of optimal formulae due to Stern (1967) are unsuitable and, as expected, we found them to be inefficient in practice. Those due to Sard (1949) we found to be unreliable. On the other hand some simple formulae such as the trapezoidal rule or Simpson's rule are not accurate enough to be efficient even though they lose no function evaluations at each interval subdivision. The 5-point Newton–Cotes and the 9-point Romberg quadratures are better because they are in general more accurate and they also lose no function evaluations at each subdivision. The 17-point, 33-point, etc. Romberg quadratures lose no function evaluations, but algorithms based on these quadratures are in general no more efficient than those using the 9-point Romberg, so we will not discuss them further.

There are two other quadratures which are very suitable for any method of integration by interval subdivision. These are the 5-point Lobatto quadrature and the 7-point Clenshaw–Curtis quadrature. The abscissas of the 5-point Lobatto quadrature include the two end points and the mid-point, therefore only two function evaluations are lost each time an interval is subdivided. Similarly the 7-point Clenshaw–Curtis formula includes, in the interval $(-1, +1)$, the 5 abscissas ± 1, $\pm \frac{1}{2}$ and 0, and hence only the function evaluations at the two other abscissas are lost when the interval is subdivided. This quadrature can be written:

$$\int_{-1}^{+1} F(t)dt = \tfrac{1}{35}[F(1) + F(-1)] + \tfrac{16}{35}[F(\tfrac{1}{2}) + F(-\tfrac{1}{2})] + \tfrac{164}{315}F(0) + \tfrac{16}{63}[F(\sqrt{3}/2) + F(-\sqrt{3}/2)]. \qquad (3.1)$$

Like other Clenshaw–Curtis quadratures (O'Hara and Smith, 1968) it has a high accuracy, comparable to or better than that of the 9-point Romberg quadrature. This is illustrated in **Table 1** where we compare some of the quadratures we have discussed for two integrands. Similar results were found for other integrands. The maximum errors shown in the table are obtained by introducing an arbitrary parameter α and changing the variable from x to y

$$\text{where } x = \frac{b+a}{2} + \frac{b-a}{2}\left[\frac{\alpha - 1 + (\alpha+1)y}{\alpha + 1 + (\alpha - 1)y}\right] \qquad (3.2)$$

to give

$$I = \int_a^b f(x)dx = \int_{-1}^{+1} g(\alpha, y)dy. \qquad (3.3)$$

The respective quadrature is then applied to the second integral for 100 values of α between $0{\cdot}5$ and $2{\cdot}5$. This is equivalent to evaluating 100 different but similar integrals for each integrand $f(x)$. This process helps to eliminate the probability of an error being accidentally small. Similar results were obtained by comparing the root-mean-square errors. We also compare the quadratures in Table 1 by giving the coefficient σ_R of the Davis–Rabinowitz (1954) error estimate:

$$|E| < \sigma_R ||f||$$

where $||f||$ is the norm of $f(z)$ over the region R in the complex plane within which $f(z)$ is assumed analytic; in the table R is taken as an ellipse with semi-major axis $a = 1{\cdot}2$. Similar results were found for other values of a.

From the table it is clear that the Clenshaw–Curtis and Romberg formulas are the most accurate. They also have many abscissas in common, so it is not surprising that when they are combined in one algorithm they yield a very efficient method for evaluating integrals.

4. Application to the algorithm

We tested the previous quadratures in our algorithm by evaluating large numbers of integrals and comparing the results. The test integrals were as follows:

$$\int_0^1 \frac{dx}{1+25x^2}; \quad \int_0^1 \frac{20}{1+6400\left(x - \frac{\sqrt{17}}{5}\right)^2}dx; \quad \int_0^1 \frac{dx}{1+100x^2};$$

$$\int_0^1 \frac{dx}{1-0{\cdot}5x^4}; \quad \int_0^1 \frac{dx}{1-0{\cdot}98x^4}; \quad \int_0^1 \frac{dx}{1-0{\cdot}998x^4};$$

$$\int_0^{(5/4)^3+1} [x - \tfrac{3}{4}(x-1)^{1/3}]dx; \quad \int_{-1}^{1} (x+\tfrac{1}{2})^{1/2}dx; \quad \int_0^6 \phi(x)dx$$

where

$$\phi(x) = e^x, \quad x \leqslant \tfrac{1}{2};$$
$$= e^{1-x}, x > \tfrac{1}{2}.$$

These were evaluated first by changing the variable so that

$$\int_a^b f(x)dx = \int_a^b \frac{1 + \alpha b}{(1 + \alpha(b - y))^2} f\left(\frac{y}{1 + \alpha(b - y)}\right) dy \tag{4.1}$$

and by evaluating the right-hand integral for values of α in the range $0 \leqslant \alpha \leqslant 255$. This distorted the integrands considerably for the extreme values of α and made the corresponding integrals very difficult to evaluate. We tested the algorithm in each case for five different accuracies ϵ between $=\frac{1}{2}10^{-3}$ and $\frac{1}{2}10^{-7}$.

We concluded that the following combination of quadrature formulas is the most reliable and efficient. The 9-point Romberg is used in each subinterval and its accuracy tested by comparing it with two 5-point Newton–Cotes formulas (using the same 9 abscissas). The interval is subdivided until the difference between these two is less than the tolerated error (no function evaluations have been lost up to this stage). We next compare the 9-point Romberg quadrature with the sum of two 7-point Clenshaw–Curtis quadratures over each half of the interval; this requires the evaluation of the integrand at four additional points in each subinterval. If this check is also satisfied, we use in addition the sum of the absolute error estimates for the 7-point Clenshaw–Curtis quadratures (O'Hara and Smith, 1968) based on the formula for the interval $(-1, +1)$

$$E_6^{(a)} = \frac{32}{(6^2 - 9)(6^2 - 1)} \sum_{s=0}^{6}{}'' (-1)^s F\left(\cos \frac{\pi s}{6}\right). \tag{4.2}$$

If this is less than the tolerated error then we adopt the sum of the two 7-point Clenshaw-Curtis quadratures as the result. In all, this result is checked by three independent error estimates and it should be very reliable. We found that amongst the approximately 6,000 applications of our algorithm to the extreme examples quoted, we had only 21 failures (by a failure we mean that the actual error is greater than the tolerated error).

We call the above method the CCR-method (Clenshaw–Curtis–Romberg-method).

Even greater reliability can be obtained by requesting an error ϵ less than the error actually required; for example there would have been only 4 failures in the above tests if we had requested an error equal to half that required, and no failures if we had requested an error one tenth that required. Alternatively we can check the final result in each subinterval with one 7-point Clenshaw–Curtis quadrature over the whole subinterval and in addition use the error estimate (4.2), and so introduce two extra checks at the expense of only 2 function evaluations. In the above tests this would have eliminated all 21 failures with about 45% more work.

We illustrate the efficiency of our CCR-method in **Table 2** where we compare it with two other methods, one based on Simpson's rule from the Atlas subroutine library and the other based on interval subdivision as in §2 but using the 4-point Gauss formula. (We illustrate only two of a large number of other comparisons we made.) In our tests the Gauss method was as reliable as the CCR-method, but much less efficient; the Atlas routine was much less reliable, it failed 59 times and in more than 1 in 5 of the test integrals it did not converge to any answer with single length arithmetic. The CCR-method converged to a result in all 6,000 integrals.

5. Conclusion

We have outlined an algorithm which will evaluate an integral to any required accuracy. It is efficient and reliable: out of several thousands of badly behaved integrals it failed only a few times, and it is easy to increase its reliability further as required.

A limited number of copies of a program in FORTRAN IV, based on the above algorithm are available on request.

Table 1

Comparison of low order quadrature formula; in the table are given maximum errors (as defined in the text) for integration of $f(x)$ over (0, 1); n is the number of abscissas and σ_R is the Davis–Rabinowitz error coefficient for $a = 1{\cdot}2$

FORMULA	n	$f(x)$		σ_R
		$(1 + 100x^2)^{-1}$	SINH (x)	
2 × 7 pt Clenshaw–Curtis	13	0·15 (−2)	0·007 (−4)	0·406 (−3)
2 × 5 pt Lobatto	9	0·94 (−2)	0·11 (−4)	0·399 (−2)
Clenshaw–Curtis	7	1·19 (−2)	0·37 (−4)	0·722 (−2)
Romberg	9	0·96 (−2)	1·54 (−4)	0·177 (−1)
2 × 5 pt Newton–Cotes	9	0·93 (−2)	2·08 (−4)	0·180 (−1)
Lobatto	5	1·82 (−2)	4·35 (−4)	0·468 (−1)
5 pt Newton–Cotes	5	7·26 (−2)	35·90 (−4)	0·122 (0)
2 × 3 pt Simpson	5	8·54 (−2)	59·37 (−4)	0·127 (0)
3 pt Simpson	3	27·74 (−2)	498·62 (−4)	0·502 (0)

This work was supported by the National Aeronautics and Space Administration, contract NSR52–112–002.

Appendix

Relative errors

We wish to evaluate the integral to a relative accuracy ϵ; that is, if E is the error in the quadrature and I is the integral then we require $|E/I|$ to be less than ϵ. If the integrand always has the same sign the problem is straightforward; we adopt the same principle in §2 and require that in each subinterval (p, q)

$$\left|\frac{E_{pq}}{I_{pq}}\right| < \epsilon + \sum_{r=a}^{s=p} \left[\epsilon - \left|\frac{E_{rs}}{I_{rs}}\right|\right] \left|\frac{I_{rs}}{I_{pq}}\right| \qquad \text{(A.1)}$$

This allows $|E_{pq}/I_{pq}|$ to be as large as possible while still keeping $|\Sigma E_{pq}/I| < \epsilon$. When the integrand changes sign the problem is more difficult because I may be small and because I_{pq} may be close to zero. This last problem can usually be overcome by jumping to the next subinterval if I_{pq} is found to be small. On the other hand if any I_{pq} is negative then $R = \Sigma|E_{pq}|/|\Sigma I_{pq}|$ may be larger than ϵ. In this case the calculation can be repeated after replacing ϵ in (A.1) by ϵ^2/R.

The use of (A.1) has been found to be satisfactory in practice.

Table 2

Number of function evaluations required to evaluate $\int_0^1 f(x)dx$ to a specified accuracy

$f(x)$	ACCURACY ϵ	CCR-METHOD	ATLAS ROUTINE	4-PT GAUSS
	0·5 (−3)	125	122	216
	0·5 (−4)	137	181	238
$\frac{1}{1-0\cdot998x^4}$	0·5 (−5)	133[b]	311	260
	0·5 (−6)	241	548	414
	0·5 (−7)	277	a	480
	0·5 (−8)	397	a	678
	0·5 (−3)	41	23	62
	0·5 (−4)	53	32	84
	0·5 (−5)	61	52	84
$\frac{1}{1+100x^2}$	0·5 (−6)	61	92	106
	0·5 (−7)	97	157	194
	0·5 (−8)	145	248	216
	0·5 (−9)	193	432	260
	0·5 (−10)	253	742	348

a No convergence
b This number is correct although smaller than the number above it. In both cases the interval was finally subdivided in exactly the same way; in the upper case the failure of an early error test was detected using 4 additional function evaluations.

References

Clenshaw, C. W., and Curtis, A. R. (1960). A method for numerical integration on an automatic computer, *Num. Math.*, Vol. 12, pp. 197–205.

Davis, P. J., and Rabinowitz, P. R. (1954). On the estimation of quadrature errors for analytical functions, *M.T.A.C.*, Vol. 8, p. 193.

O'Hara, H. (1969). *The efficient and reliable computation of definite integrals.* Thesis, Queen's University of Belfast.

O'Hara, H., and Smith, F. J. (1968). Error estimation in the Clenshaw–Curtis quadrature formula, *Computer Journal*, Vol. 11, pp. 213–219.

Ralston, A. (1965). *A first course in numerical analysis*, McGraw-Hill, pp. 111–114.

Sard, A. (1949). Best approximate integration formulas; best approximation formulas, *American J. Math.*, Vol. 71, pp. 80–91.

Stern, M. (1967). Optimal quadrature formulae, *Computer Journal*, Vol. 9, pp. 396–403.

Wright, K. (1966). Series methods for integration, *Computer Journal*, Vol. 9, pp. 191–199.

Book Review

Information Theory for Systems Engineers, by L. P. Hyvärinen, 1968; 205 pages. (No. 5 in *Lecture Notes in Operations Research and Mathematical Economics.*) (Berlin: *Springer-Verlag*, $3.80.)

Being based on lectures given at the IBM European Systems Research Institute, this book is composed in terms of the interests of computer users and computer designers. Its level is very well within the scope of a modern undergraduate course in computer science or communication engineering, since it assumes only a basic knowledge of calculus and of the theory of probability and statistics. The author points out that information may be *syntactic* (commonly known to communication engineers as *selective*), *semantic* or *pragmatic.* Any process which does not destroy syntactic information is reversible: this is a formulation which, incidentally, could be shown to be parallel with reversible operations in thermodynamics which do not increase entropy. But data-processing is often concerned with reducing the quantity of data in order to make it easier to grasp what remains, a process of reducing the syntactic information in order to increase the pragmatic information.

The treatment of coding is rather sketchy: it is largely confined to the types of parity check which are used within the structure of computer systems and barely mentions data transmission. The types of burst-correcting codes most suitable for transmission are not mentioned, but k-out-of-m codes (e.g. 2 out of 5 and 4 out of 8) are recommended for protection against bursts of errors. There are also more sophisticated methods of coding decimal inventory numbers than the simple complement digit described in this book.

In summary, the book is useful to a computer user who wishes to understand the customary parity check procedures but is not sufficient for a computer designer or programmer who wishes to devise more sophisticated error-protection.

D. A. Bell (Hull)