Contents

5 Numerical Integration **1**

5.1 Interpolatory Quadrature Rules 1
5.1.1 Introduction 1
5.1.2 Some Classical Formulas 3
5.1.3 Higher Order Newton–Cotes' Formulas 7
5.1.4 Weighted Quadrature Rules 11
Review Questions 14
Problems and Computer Exercises 15
5.2 Quadrature Rules with Free Nodes 18
5.2.1 Gauss–Christoffel Quadrature 22
5.2.2 Applications of Gauss Quadrature 24
5.2.3 Matrix Formulas Related to Gauss Quadrature . . . 30
5.2.4 Symmetric Weight Functions 36
Review Questions 37
Problems and Computer Exercises 37
5.3 Extrapolation Methods 40
5.3.1 Euler–Maclaurin Formula 40
5.3.2 Romberg's Method 41
5.3.3 The Epsilon Algorithm 48
5.3.4 Infinite Intervals 49
5.3.5 Adaptive Quadrature 51
Review Questions 54
Problems and Computer Exercises 54
5.4 Multiple Integrals 56
5.4.1 Product Rules 56
5.4.2 Successive One-Dimensional Quadrature 58
5.4.3 Product Rules 59
5.4.4 Irregular Triangular Grids 60
5.4.5 Monte Carlo Methods 65
Review Questions 66
Problems 67

Bibliography **69**

Index **72**

Chapter 5
Numerical Integration

5.1 Interpolatory Quadrature Rules

5.1.1 Introduction

As is well known, even many relatively simple integrals cannot be expressed in finite terms of elementary functions, and must be evaluated by numerical methods. The problem to calculate the definite integral of a given function over a finite interval is often called **numerical quadrature**, since it relates to the ancient problem of the quadrature of the circle, i.e., constructing a square with equal area to that of a circle.

In this chapter we study the problem of *how to find the parameters in a formula* for the approximate calculation integrals

$$I(f) = \int_a^b f(x)\,dx.$$

Note that $I(f)$ is a linear functional and hence the problem is a special case of approximating a linear functional considered in Sec. 3.3.4. The quadrature rules considered will be of the form

$$\int_a^b f(x)\,dx \approx \sum_{i=1}^{n} w_i f(x_i), \tag{5.1.1}$$

where the **nodes** $x_1 < x_2 < \cdots < x_n$ are distinct and **weights** $w_1, w_2, \ldots, w_n$. Often (but not always) all nodes lie in $[a, b]$.

The weights w_i are usually determined so that the formula (5.1.1) is exact for polynomials of as high degree as possible.

Definition 5.1.1. *A quadrature rule* (5.1.1) *has* **order of accuracy** *(or degree of exactness) equal to d if it is exact for all polynomials of degree $\leq d$, i.e. for all $p \in \mathcal{P}_{d+1}$.*

The coefficients w_i depend only on the distribution of the points $\{x_i\}_{i=1}^n$. Note that the relation

$$\int_a^b dx = \sum_{i=1}^n w_i = (b-a) \tag{5.1.2}$$

follows from the requirement that the formula is exact for $f(x) \equiv 1$. Suppose that the function values $f(x_i)$ is evaluated with an error e_i, such that $|e_i| \leq \epsilon$, for all $i = 1:n$. Then, if $w_i \geq 0$, the related error in the quadrature formula satisfies

$$\Big|\sum_{i=1}^n w_i e_i\Big| \leq \epsilon \sum_{i=1}^n |w_i| \leq \epsilon(b-a). \tag{5.1.3}$$

However, this upper bound does not hold if some weights in the quadrature rules are negative.

In an **interpolatory** quadrature formula the integral is approximated by $\int_a^b w(x)p(x)\,dx$, where $p(x)$ is the unique polynomial of degree $n-1$ interpolating $f(x)$ at the distinct points $x_1, x_2, \ldots, x_n$. By Lagrange's interpolation formula (Theorem 4.2.6)

$$p(x) = \sum_{i=1}^n f(x_i)\ell_i(x), \qquad \ell_i(x) = \prod_{\substack{j=1 \\ j\neq i}}^n \frac{(x-x_j)}{(x_i-x_j)},$$

where $\ell_i(x)$ are the elementary Lagrange polynomials associated with the nodes $x_1, x_2, \ldots, x_n$. It follows that the weights are given by

$$w_i = \int_a^b \ell_i(x)\,dx. \tag{5.1.4}$$

In practice, the coefficients are often more easily computed using the method of undetermined coefficients rather than by integrating $\ell_i(x)$.

An expression for the truncation error is obtained by integrating the remainder (see Theorems 4.2.3 and 4.2.4)

$$\begin{aligned} R_n(f) &= \int_a^b [x_1, \ldots, x_n, x] f \prod_{i=1}^n (x - x_i)\,dx \\ &= \frac{1}{n!}\int_a^b f^{(n)}(\xi_x) \prod_{i=1}^n (x-x_i)\,dx, \quad \xi_x \in [a,b]. \end{aligned} \tag{5.1.5}$$

where the second expression holds if $f^{(n)}$ is continuous in $[a,b]$.

Theorem 5.1.2. *For any given set of nodes $x_1, x_2, \ldots, x_n$ an interpolatory quadrature formula with weights* (5.1.4) *has order of exactness equal to at least $d = n-1$. Conversely, if the formula has degree of exactness $n-1$, then the formula is interpolatory.*

Proof. For any $f \in \mathcal{P}_n$ we have $p(x) = f$, and hence (5.1.4) has degree of exactness at least equal to $n-1$. On the other hand, if the degree of exactness of (5.1.4) is $n-1$, then putting $f = \ell_i(x)$ shows that the weights w_i satisfy (5.1.4), i.e. the formula is interpolatory. □

5.1.2 Some Classical Formulas

Interpolatory quadrature formulas, where the nodes are constrained to be equally spaced, are called **Newton–Cotes'**[1] formulas. These are especially suited for integrating a tabulated function, a task that was more common before the computer age. The midpoint, trapezoidal and Simpson's formula, to be described here, are all special cases of Newton–Cotes' formulas.

The **trapezoidal rule** (cf. Figure 1.2.5) is based on linear interpolation of $f(x)$ at $x_1 = a$ and $x_2 = b$, that is $f(x)$ is approximated by

$$p(x) = f(a) + (x-a)[a,b]f = f(a) + (x-a)\frac{f(b)-f(a)}{b-a}.$$

The integral of $p(x)$ equals the area of a trapezoid with base $(b-a)$ times the average height $\frac{1}{2}(f(a)+f(b))$. Hence

$$\int_a^b f(x)\,dx \approx \frac{(b-a)}{2}(f(a)+f(b)).$$

To increase the accuracy we subdivide the interval $[a,b]$ and assume that $f_i = f(x_i)$ is known on a grid of equidistant points

$$x_0 = a, \quad x_i = x_0 + ih, \quad x_n = b. \tag{5.1.6}$$

where $h = (b-a)/n$ is the **step length**. The trapezoidal approximation for the ith subinterval is

$$\int_{x_i}^{x_{i+1}} f(x)\,dx = T(h) + R_i, \quad T(h) = \frac{h}{2}(f_i + f_{i+1}), \tag{5.1.7}$$

which is the **composite trapezoidal rule**

Assume now that $f''(x)$ is continuous in $[a,b]$. Using the exact remainder in Newton's interpolation formula (see Theorem 4.2.3) we have

$$R_i = \int_{x_i}^{x_{i+1}} (f(x) - p_2(x))\,dx = \int_{x_i}^{x_{i+1}} (x-x_i)(x-x_{i+1})\,[x_i, x_{i+1}, x]f\,dx. \tag{5.1.8}$$

Since $[x_i, x_{i+1}, x]f$ is a continuous function of x and $(x-x_i)(x-x_{i+1})$ has constant (negative) sign for $x \in [x_i, x_{i+1}]$, the mean-value theorem of integral calculus gives

$$R_i = [x_i, x_{i+1}, \xi_i]f \int_{x_i}^{x_{i+1}} (x-x_i)(x-x_{i+1})\,dx, \quad \xi_i \in [x_i, x_{i+1}].$$

[1] Roger Cotes (1682–1716) was a highly appreciated young colleague of Isaac Newton. He was entrusted with the preparation of of the second edition of Newton's *Principia*. He worked and published the coefficients for Newton's formulas for numerical integration for $n \le 11$.

Setting $x = x_i + ht$, and using the Theorem 4.2.4, we get

$$R_i = -\frac{1}{2}f''(\zeta_i)\int_0^1 h^2 t(t-1)h\,dt = -\frac{1}{12}h^3 f''(\zeta_i), \quad \zeta_i \in [x_i, x_{i+1}]. \tag{5.1.9}$$

For another proof of this result using the Peano kernel, see Example 3.2.7.

Summing the contributions for each subinterval $[x_i, x_{i+1}]$, $i = 0 : n.$ gives

$$\int_a^b f(x)\,dx = T(h) + E_T, \quad T(h) = \frac{h}{2}(f_0 + f_n) + h\sum_{i=2}^{n-1} f_i, \tag{5.1.10}$$

where the **global** truncation error is

$$E_T = -\frac{h^3}{12}\sum_{i=0}^{n-1} f''(\zeta_i) = -\frac{1}{12}(b-a)h^2 f''(\xi), \quad \xi \in [a, b]. \tag{5.1.11}$$

(The last equality follows since f'' was assumed to be continuous on the interval $[a, b]$.) This shows that by choosing h small enough we can make the truncation error arbitrary small. In other words we have **asymptotic convergence** when $h \to 0$.

In the **midpoint rule** $f(x)$ is approximated on $[x_i, x_{i+1}]$ by its value $f_{i+1/2} = f((x_i + x_{i+1})/2$ at the midpoint of the interval. This leads to the approximation

$$\int_{x_i}^{x_{i+1}} f(x)\,dx = M(h) + R_i, \quad M(h) = hf_{i+1/2} \tag{5.1.12}$$

The midpoint rule approximation can be interpreted as the area of the trapezium defined by the tangent of f at the midpoint $x_{i-\frac{1}{2}}$.

The remainder term in Taylor's formula gives

$$f(x) - f_{i-\frac{1}{2}} = (x - x_{i-\frac{1}{2}})f'_{i-\frac{1}{2}} + \tfrac{1}{2}(x - x_{i-\frac{1}{2}})^2 f''(\zeta_x), \quad \zeta_x \in [x_{i-1}, x_i].$$

By symmetry the integral over $[x_{i-1}, x_i]$ of the linear term vanishes. We can use the mean value theorem, to show that

$$R_i = \int_{x_i}^{x_{i+1}} \tfrac{1}{2}f''(\zeta_x)(x - x_{i-\frac{1}{2}})^2\,dx = \tfrac{1}{2}f''(\zeta_i)\int_{-\frac{1}{2}}^{\frac{1}{2}} h^3 t^2\,dt = \frac{h^3}{24}f''(\zeta_i).$$

Although it uses just *one function value* the midpoint rule, like the trapezoidal rule, is exact when $f(x)$ is a linear function. Summing the contributions for each subinterval we obtain the **composite midpoint rule**

$$\int_a^b f(x)\,dx = R(h) + E_M, \quad R(h) = h\sum_{i=0}^{n-1} f_{i+1/2}, \tag{5.1.13}$$

(Compare the above approximation with the Riemann sum in the *definition* of a definite integral.) For the global error we have

$$E_M = \frac{(b-a)h^2}{24}f''(\zeta), \quad \zeta \in [a, b]. \tag{5.1.14}$$

The trapezoidal rule is called a **closed rule** because values of f at both endpoints are used. It is not uncommon that f has an integrable singularity at an endpoint. In that case an **open rule**, like the midpoint rule, can still be applied.

If $f''(x)$ has constant sign in each subinterval then the error in the midpoint rule is approximately half as large as that for the trapezoidal rule and has the opposite sign. However, the trapezoidal rule is more economical to use when a sequence of approximations for $h, h/2, h/4, \ldots$ is to be computed, since about half of the values needed for $h/2$ were already computed and used for h, etc. indeed, it is easy to verify the following useful relation between the trapezoidal and midpoint rules:

$$T(h/2) = \frac{1}{2}(T(h) + M(h)). \tag{5.1.15}$$

If the magnitude of the error in the function values does not exceed $\frac{1}{2}U$, then for the trapezoidal and midpoint rules the magnitude of the propagated error in the approximation is bounded by $(b-a)\frac{1}{2}U$, independent of h. Note that this holds *for any quadrature formula* (5.1.1), *provided that all weights* w_i *are positive.*

If the rounding error is negligible and h sufficiently small, then it follows from (5.1.11) that the error in $T(h/2)$ is about 1/4-th of that in $T(h)$. Hence the magnitude of the error in $T(h/2)$ can be estimated by $\frac{1}{3}|T(h/2) - T(h)|$, or more conservatively by $|T(h/2) - T(h)|$. (A more systematic use of Richardson extrapolation is made in Romberg's method; see Sec. 5.3.2.)

Example 5.1.1.

Compute approximately $\displaystyle\int_0^{0.8} \frac{\sin x}{x}\, dx$. As an exercise the reader should check some of the midpoint and trapezoidal sums given below, which are correct to ten decimals. (Use (5.1.15).)

h	$M(h)$	$T(h)$
0.8	0.77883 66846	0.75867 80454
0.4	0.77376 69772	0.76875 73650
0.2	0.77251 27162	0.77126 21711
0.1		0.77188 74437

The correct value, to six decimals, is 0.772096. Verify that in this example the error is approximately proportional to h^2 for both $M(h)$ and $T(h)$. We estimate the error in $T(0.1)$ to be $\frac{1}{3}6.26 \cdot 10^{-4} \leq 2.1 \cdot 10^{-4}$.

From the error analysis above we note that the error in the midpoint rule is roughly half the size of the error in the trapezoidal rule and of opposite sign. Hence it seems that the linear combination

$$S(h) = \frac{1}{3}(T(h) + 2M(h)). \tag{5.1.16}$$

should be a better approximation. This is indeed the case and (5.1.16) is equiva-

lent to **Simpson's rule**[2], one of the most famous classical formulas for numerical integration.

Another way to derive Simpson's rule is to approximate $f(x)$ by a piecewise polynomial of third degree. It is convenient to shift the origin to the midpoint of the interval and consider the integral over the interval $[x_i-h, x_i+h]$. From Taylor's formula we have

$$f(x) = f_i + (x-x_i)f_i' + \frac{(x-x_i)^2}{2}f_i'' + \frac{(x-x_i)^3}{3!}f_i''' + O(h^4),$$

where the remainder is zero for all polynomials of degree 3 or less. Integrating term by term, the integrals of the second and fourth term vanishes giving

$$\int_{x_i-h}^{x_i+h} f(x)\,dx = 2hf_i + 0 + \frac{1}{3}h^3 f_i'' + 0 + O(h^5).$$

Using $h^2 f_i'' = (f_{i-1} - 2f_i + f_{i+1}) + O(h^4)$ (see (4.7.5)) we have that

$$\begin{aligned}\int_{x_i-h}^{x_i+h} f(x)\,dx &= 2hf_i + \frac{1}{3}h(f_{i-1} - 2f_i + f_{i+1}) + O(h^5) \\ &= \frac{1}{3}h(f_{i-1} + 4f_i + f_{i+1}) + O(h^5),\end{aligned} \tag{5.1.17}$$

where the remainder term is zero for all third-degree polynomials. We now determine the error term for $f(x) = (x-x_i)^4$, which is

$$R_T = \frac{1}{3}h(h^4 + 0 + h^4) - \int_{x_i-h}^{x_i+h} x^4\,dx = (2/3 - 2/5)h^5 = \frac{4}{15}h^5.$$

It follows that an *asymptotic* error estimate is

$$R_T = h^5\frac{4}{15}\frac{f^{(4)}(x_i)}{4!} + O(h^6) = \frac{h^5}{90}f^{(4)}(x_i) + O(h^6).$$

A strict error estimate for Simpson's rule is more difficult to obtain. As for the midpoint formula the midpoint x_i can be considered as a double point of interpolation; see Problem 3. The general error formula (5.1.5) then gives

$$R(f) = \frac{1}{4!}\int_{x_{i-1}}^{x_{i+1}} f^{(4)}(\xi_x)(x-x_{i-1})(x-x_i)^2(x+x_{i+1})\,dx.$$

where $(x-x_{i-1})(x-x_i)^2(x+x_{i+1})$ has constant sign on $[x_{i-1}, x_{i+1}]$. If $2h$ is the length of the interval of integration Using the mean value theorem gives the error

$$-\frac{1}{90}f^{(4)}(\xi)h^5, \quad |\xi| < h. \tag{5.1.18}$$

[2]The English mathematician Thomas Simpson (1710–1761) is best known for his work on interpolation and quadrature. He also worked on probability theory.

The remainder can also be obtained from Peano's error representation It can be shown (see Stoer [32, p. 152 ff]) that for Simpson's rule

$$Rf = \int_{\mathbf{R}} f^{(4)}(u)K(u)\,du,$$

where the kernel equals

$$K(u) = -\frac{1}{72}(h-u)^3(3u+h)^2, \quad 0 \le u \le h,$$

and $K(u) = K(|u|)$ for $u < 0$, $K(u) = 0$ for $|u| > h$. This again gives (5.1.18).

In the **composite Simpson's formula** one divides the interval $[a, b]$ into an *even* number $n = 2m$ steps of length h, and use the formula (5.1.17) on each of m double steps, giving

$$\int_a^b f(x)\,dx = \frac{h}{3}(f_0 + 4U + 2E + f_n) + R_T, \tag{5.1.19}$$

where

$$U = f_1 + f_3 + \cdots + f_{n-1}, \qquad E = f_2 + f_4 + \cdots + f_{n-2}.$$

The remainder is

$$R_T = \sum_{i=0}^{m-1} \frac{h^5}{90} f^{(4)}(\xi_i) = \frac{(b-a)}{180} h^4 f^{(4)}(\xi), \qquad \xi \in [a, b]. \tag{5.1.20}$$

This shows that wee have gained *two orders of accuracy* compared to the trapezoidal rule, without using more function evaluations. This is why Simpson's rule is such a popular general-purpose quadrature rule.

5.1.3 Higher Order Newton–Cotes' Formulas

The classical Newton–Cotes' quadrature rules, are interpolatory rules obtained for $w(x) = 1$ and equidistant points in $[0, 1]$. There are two classes: **closed formulas**, where the end points of the interval belong to the nodes; **open formulas**, where all nodes lie strictly in the interior of the interval. The closed Newton–Cotes' formulas are usually written

$$\int_0^{nh} f(x)\,dx = h\sum_{j=0}^{n} w_j f(jh) + R_n(f) \quad w_j = w_{n-j}, \tag{5.1.21}$$

where, in principle, the weights w_i can be determined from (5.1.4). By (5.1.2) they satisfy

$$\sum_{j=0}^{n} h w_j = nh. \tag{5.1.22}$$

(Note that we here sum over $n+1$ points in contrast to our previous notation.) The closed Newton–Cotes' rule for $n = 1$ and $n = 2$ are equivalent to the trapezoidal rule and Simpson's rule, respectively.

In general it can be shown that the closed Newton–Cotes' formula integrate all polynomials of degree d exactly, where $d = n$ for n odd and $d = n + 1$ for n even. The extra accuracy for n even is, as in Simpson's rule, due to symmetry. For $n \leq 7$ the coefficients w_i are positive, but for $n = 8$ and $n \geq 10$ negative coefficients appear. Such formulas may still be useful, but since $\sum_{j=0}^{n} h|w_j| > nh$, they are less robust with respect to errors in the function values f_i.

Similarly the open Newton–Cotes' formulas are usually written as

$$\int_0^{nh} f(x)\,dx = h\sum_{i=1}^{n-1} w_i f(ih) + R_{n-1,n}(h), \quad w_{-j} = w_{n-j}.$$

The simplest open Newton–Cotes' formula for $n = 2$ is the midpoint rule with step size $2h$. The open formulas have order $d = n - 1$ for n even and $n - 2$ for n odd. For the open formulas negative coefficients occur already for $n = 4$ and $n = 6$.

The Peano kernels for both the open and the closed formulas can be shown to have constant sign (see Steffensen [31]). Thus the local truncation error can be written as

$$R_n(h) = c_{n,d}h^{d+1}f^{(d)}(\zeta), \quad \zeta \in [0, nh], \tag{5.1.23}$$

It is easily shown that the Peano kernels for the corresponding composite formulas also have constant sign.

The Newton–Cotes' closed formulas for $n \leq 6$ and open formulas for $n \leq 5$, with error terms, are given in Tables 5.1.1 and 5.1.2, respectively. Note that the sign of the error coefficients in the open rules are opposite the sign in the closed rules. Higher order Newton–Cotes' formulas are given in Abramowitz and Stegun [1, pp. 886–887],

Table 5.1.1. *The coefficients $w_i = Ac_i$ in the n-points closed Newton–Cotes' formulas.*

n	d	A	c_0	c_1	c_2	c_3	c_4	c_5	c_6	c_n
1	1	1/2	1	1						−1/12
2	3	1/3	1	4	1					−1/90
3	3	3/8	1	3	3	1				−3/80
4	5	2/45	7	32	12	32	7			−8/945
5	5	5/288	19	75	50	50	75	19		−275/12 096
6	7	1/140	41	236	27	272	27	236	41	−9/1400

We now show how the classical Newton–Cotes formulas for $w(x) = 1$ can be derived using the operator methods developed in Sec. 3.3. Let m, n be given integers and let h be a positive step size. In order to utilize the symmetry of the problem easier, we move the origin to the midpoint of the interval of integration. If we set

$$x_j = jh, \qquad f_j = f(jh), \quad j = -n/2 : 1 : n/2,$$

Table 5.1.2. *The coefficients $w_i = Ac_i$ in the n-points open Newton–Cotes' formulas.*

n	d	A	c_1	c_2	c_3	c_4	c_5		c_n.
2	1	2	1						1/24
3	1	3/2	1	1					1/4
4	3	4/3	2	−1	2				14/45
5	3	5/24	11	1	1	11			95/144
6	5	3/10	11	−14	26	−14	11		41/140
7	5	7/1440	611	−453	562	562	−453	611	5257/8640

the Newton–Cotes formula now reads

$$\int_{-mh/2}^{mh/2} f(x)\,dx = h \sum_{j=-n/2}^{n/2} w_j f_j + R_{m,n}(h), \quad w_{-j} = w_j. \tag{5.1.24}$$

Note that j, $n/2$ and $m/2$ are not necessarily integers. For a Newton–Cotes formula $n/2 - j$ and $m/2 - j$ are evidently integers. Hence $(m - n)/2$ is an integer too, but there may be other formulas, perhaps almost as good, where this is not the case. The coefficients $w_j = w_{j;m,n}$ are to be determined so that the remainder $R_{m,n}$ vanishes if $f \in \mathcal{P}_q$, with q as large as possible for given m, n. The left hand side of (5.1.24), divided by h, reads in operator form,

$$(e^{hDm/2} - e^{-hDm/2})(hD)^{-1} f(x_0),$$

which is an even function of hD. By (3.3.38), hD is an odd function of δ. It follows that the left hand side is an even function of δ, hence we can, for every m, write

$$(e^{hDm/2} - e^{-hDm/2})(hD)^{-1} \quad \mapsto A_m(\delta^2) = a_{1m} + a_{2m}\delta^2 + \ldots + a_{k+1,m}\delta^{2k} \ldots \tag{5.1.25}$$

We truncate after (say) δ^{2k}; the first neglected term is then $a_{k+2,m}\delta^{2k+2}$. We saw in Sec. 3.3.4 how to bring a truncated δ^2-expansion to $B(E)$-form

$$b_1 + b_2(E + E^{-1}) + b_3(E^2 + E^{-2}) + \ldots + b_k(E^k + E^{-k}).$$

by matrix multiplication with a matrix M of the form given in (3.3.45). By comparison with (5.1.24), we conclude that $n/2 = k$, that the indices j are integers, and that $w_j = b_{j+1}$ (if $j \geq 0$). *If m is even, this becomes a Newton–Cotes formula.* If m is odd, it may still be a useful formula, but it does not belong to the Newton–Cotes family, because $(m - n)/2 = m/2 - k$ is no integer.

If $n = m$ a formula is of the closed type. Its remainder term is the first neglected term of the operator series, truncated after δ^{2k}, $2k = n = m$ (and multiplied by h). Hence the remainder of (5.1.24) can be estimated by $a_{2+m/2}\delta^{m+2} f_0$.

or (better)

$$R_{m,m} \sim (a_{m/2+2}/m)H(hD)^{m+2}f_0.$$

where we call $H = mh$ the "bigstep".

If the integral is computed over $[a, b]$ by means of a sequence of "bigsteps", each of length H, an estimate of the *global* error has the same form, except that H is replaced by $b-a$, and f_0 is replaced by $\max_{x\in[a,b]} |f(x)|$. The exponent of hD in an error estimate that contains H or $b-a$, is known as the *global order of accuracy* of the method.

If $n < m$, a formula of the open type is obtained. Among the open formulas we shall only consider the case that $n = m-2$, which are the open Newton–Cotes formula. The operator expansion is truncated after δ^{m-2}, and we obtain

$$R_{m-2,m} \sim (a_{m/2+1}/m)H(hD)^{m}f_0.$$

Formulas with $n > m$ are rarely mentioned in the literature (except for $m = 1$). We do not understand why; it is rather common that an integrand has a smooth continuation outside the interval of integration.

Example 5.1.2.

The coefficients a_{im} in the expansion (5.1.25) can be computed by means of the Cauchy+FFT method. In this way extensive algebraic calculations are avoided[3]. It can be shown that the exact coefficients are rational numbers, though it is sometimes hard to estimate in advance the order of magnitude of the denominators. The algorithm must be used with judgment.

The coefficients are first obtained in floating point representation. The transformation to rational form is obtained by a continued fraction algorithm, described in Example 3.4.1.

For the case $m = 8$ the result reads,

$$A_8(\delta^2) = 8 + \frac{64}{3}\delta^2 + \frac{688}{45}\delta^4 + \frac{736}{189}\delta^6 + \frac{3956}{14175}\delta^8 - \frac{2368}{467775}\delta^{10} + \dots \qquad (5.1.26)$$

The closed integration formula becomes

$$\int_{-x_4}^{x_4} f(x)dx = \frac{4h}{14175}\Big(-4540f_0 + 10496(f_1 + f_{-1}) - 928(f_2 + f_{-2}) + 5888(f_3 + f_{-3)}) + 989(f_4 + f_{-4})\Big) + R, \qquad (5.1.27)$$

$$R \sim \frac{296}{467775}Hh^{10}f^{(10)}(x_0). \qquad (5.1.28)$$

It goes without saying that this is not how Newton and Cotes found their methods. Our method may seem complicated, but the Matlab programs for this are rather short, and to a large extent useful for other purposes. The computation of about 150 Cotes-coefficients and 25 remainders ($m = 2 : 14$), took less than two seconds on a PC. This includes the calculation of several alternatives for rational

[3]These could, however, be carried out using a system like Maple.

approximations to the floating-point results. For a small number of the 150 coefficients the judicious choice among the alternatives took, however, much more than 2 (human) seconds; this detail is both science and art.

It was mentioned that, *if m is odd*, (5.1.25) does not provide formulas of the Newton–Cotes family, since $(m-n)/2$ is no integer, nor are the indices j in (5.1.24) integers. So, the operator associated with the right hand side of (5.1.24) is of the form

$$c_1(E^{1/2}+E^{-1/2})+c_2(E^{3/2}+E^{-3/2})+c_3(E^{5/2}+E^{-5/2})+\dots.$$

If it is divided algebraically by $\mu=\frac{1}{2}(E^{1/2}+E^{-1/2})$, however, it becomes of the $B(E)$-form (say)

$$b'_1+b'_2(E+E^{-1})+b'_3(E^2+E^{-2})+\ldots+b_k(E^k+E^{-k}).$$

If m is odd we therefore expand

$$(e^{hDm/2}-e^{-hDm/2})(hD)^{-1}/\mu, \qquad \mu=\sqrt{1+\delta^2/4},$$

into a δ^2-series, with coefficients a'_j. Again this can be done numerically by the Cauchy+FFT method. For each m two truncated δ^2-series, one for the closed an one for the open case, are then transformed into $B(E)$-expressions numerically by means of the matrix M, as described above. The expressions are then multiplied *algebraically* by $\mu=\frac{1}{2}(E^{1/2}+E^{-1/2})$. We then have the coefficients of a Newton–Cotes formula with m odd.

The asymptotic error is

$$a'_{m/2+1}H(hD)^{m+1} \quad \text{and} \quad a'_{m/2-1}H(hD)^{m-1}$$

for the closed type, and open type, respectively ($2k=m-1$). The global orders of accuracy for Newton–Cotes methods with odd m are thus the same as for the methods, where m is one less.

5.1.4 Weighted Quadrature Rules

Newton–Cotes' quadrature rules consist of approximating the integrand by a polynomial and then integrate the polynomial exactly. Thus the accuracy depends on how well the function $f(x)$ can (locally) be approximated by a polynomial. A sufficient condition that the method converges as $h\to 0$ is that the integrand be continuous, but to get rapid convergence more is required.

If the integrand becomes infinite at a point, some modification is *necessary*. Even if some low-order derivative of the function is infinite at some point in or near the interval of integration, one *should* make such a modification. It is not uncommon that, when using a constant step-size, a single step taken close to a point where, for example, the derivative of the integrand is infinite, gives a larger error than all other steps combined.

It is often advantageous to consider quadrature rules of the form

$$\int_a^b f(x)w(x)\,dx \approx \sum_{i=1}^{n} w_i f(x_i). \tag{5.1.29}$$

Here $w(x) \geq 0$ is a given **weight function** (or density function) chosen so that $f(x)$ can be well approximated by a polynomial. To assure that the integral (5.1.29) is well defined when $f(x)$ is a polynomial, we assume in the following that the integrals

$$\mu_k = \int_a^b x^k w(x)\,dx, \quad k = 1, 2, \ldots, , \tag{5.1.30}$$

are defined for all $k \geq 0$, and $\mu_0 > 0$. The limits (a, b) of integration are here allowed to be infinite. Since the formula should be exact for $f(x) = 1$ it holds that

$$\mu_0 = \int_a^b 1 \cdot w(x)\,dx = \sum_{i=1}^{n} w_i. \tag{5.1.31}$$

The quantity μ_k is called the kth (ordinary) **moment** with respect to the weight function $w(x)$. For an interpolatory quadrature formula the weights are given by

$$w_i = \int_a^b \ell_i(x)w(x)\,dx. \tag{5.1.32}$$

Example 5.1.3.

Newton–Cotes formulas with weight functions other than $w(x) = 1$ are useful, e.g., when the integrand has a singularity. Such formulas can be derived by the method of undetermined coefficients. Consider the formula

$$\frac{1}{\sqrt{2h}} \int_0^{2h} x^{-1/2} f(x)\,dx \approx C_0 f(0) + C_1 f(h) + C_2 f(2h),$$

which is to be exact for any second-degree polynomial $f(x)$. Equating the left and right hand sides for $f(x) = 1, x, x^2$ we obtain

$$C_0 + C_1 + C_2 = 2, \qquad \frac{1}{2}C_1 + C_2 = \frac{2}{3}, \qquad \frac{1}{4}C_1 + C_2\frac{2}{5}.$$

This linear system is easily solved, giving $C_0 = 12/15$, $C_1 = 16/15$, $C_2 = 2/15$.

There are also other possibilities to treat integrals, where the integrand has a **singularity** or is "almost singular".

Example 5.1.4.

In the integral

$$I = \int_0^1 \frac{1}{\sqrt{x}} e^x\,dx$$

the integrand is infinite at the origin. By the substitution $x = t^2$ we get $I = 2\int_0^1 e^{t^2}\,dt$, which can be treated without difficulty.

Another possibility is to use integration by parts.

$$I = \int_0^1 x^{-1/2}e^x\,dx = 2x^{1/2}e^x\big|_0^1 - 2\int_0^1 x^{1/2}e^x\,dx$$
$$= 2e - 2\frac{2}{3}x^{3/2}e^x\big|_0^1 + \frac{4}{3}\int_0^1 x^{3/2}e^x\,dx = \frac{2}{3}e + \frac{4}{3}\int_0^1 x^{3/2}e^x\,dx.$$

The last integral has a mild singularity at the origin. If one wants high accuracy, then it is advisable to integrate by parts a few more times before the numerical treatment.

It is often profitable to investigate whether or not one can transform or modify the given problem in some way to make it more suitable for numerical integration. Below we give give some selected examples.

Example 5.1.5. (Simple Comparison Problem)

In $I = \int_{0.1}^1 x^{-3}e^x\,dx$ the integrand is infinite near the left end point. If we write

$$I = \int_{0.1}^1 x^{-3}\Big(1 + x + \frac{x^2}{2}\Big)\,dx + \int_{0.1}^1 x^{-3}\Big(e^x - 1 - x - \frac{x^2}{2}\Big)\,dx$$

the first integral can be computed analytically. The second integrand can be treated numerically. The integer and its derivatives are of moderate size. Note, however, the cancellation in the evaluation of the integrand.

For integrals over an infinite interval one can try some substitution which maps the interval $(0,\infty)$ to $(0,1)$, e.g., $t = e^{-x}$ of $t = 1/(1+x)$. However, in such cases one must be careful not to introduce an unpleasant singularity into the integrand instead.

Example 5.1.6.

Consider the integral $I = \int_0^\infty (1+x^2)^{-4/3}\,dx$. If one wants five decimal digits in the result then $\int_R^\infty$ is not negligible until $R \approx 10^3$. But one can expand the integrand in powers of x^{-1} and integrate term-wise,

$$\int_R^\infty (1+x^2)^{-4/3}\,dx = \int_R^\infty x^{-8/3}(1+x^{-2})^{-4/3}\,dx$$
$$= \int_R^\infty \Big(x^{-8/3} - \frac{4}{3}x^{-14/3} + \frac{14}{9}x^{-20/3} - \cdots\Big)$$
$$= R^{-5/3}\Big(\frac{3}{5} - \frac{4}{11}R^{-2} + \frac{14}{51}R^{-4} - \cdots\Big).$$

If this expansion is used, then one needs only apply numerical integration to the interval $[0, 8]$.

Example 5.1.7.

With the substitution $t = 1/(1+x)$ the integral in the previous example becomes

$$I = \int_0^1 (t^2 + (1-t)^2)^{-4/3} t^{2/3}\, dt.$$

The integrand now has an infinite derivative at the origin. This can be eliminated by making the substitution $t = u^3$, to get

$$I = \int_0^1 (u^6 + (1-u^3)^2)^{-4/3} 3u^4\, du,$$

which can be computed with, for example, a Newton–Cotes' method.

If the integrand is **oscillating**, then with ordinary integration methods one must choose a step size which is small with respect to the wave length; this is often an irritating limitation in many applications. The techniques previously mentioned (simple comparison problem, special integration formula, etc.) are sometimes effective in such situations. In addition, the following method can be used on integrals of the form

$$I = \int_0^\infty f(x) \sin(g(x))\, dx,$$

where $g(x)$ is an increasing function, and both $f(x)$ and $g(x)$ can be approximated by a polynomial. Set

$$I = \sum_{n=0}^{\infty} (-1)^n u_n, \qquad u_n = \int_{x_n}^{x_{n+1}} f(x) |\sin(g(x))|\, dx,$$

where $x_0, x_1, x_2, \ldots$ are the successive zeros of $\sin(g(x))$. The convergence of this alternating series can then be improved with the help of repeated averaging, see Sec. 3.2.1.

Review Questions

1. Why is a weight function $w(x) > 0$ included in many quadrature rules?
2. What is meant by the order of accuracy of a quadrature formula? Name three classical quadrature methods and give their order of accuracy.
3. What is meant by a composite quadrature rule? What is the difference between local and global error?
4. Give an account of the theoretical background of the classical Newton–Cotes rules.
5. Describe some possibilities for treating integrals, where the integrand has a **singularity** or is "almost singular".

Problems and Computer Exercises

1. (a) Derive the closed Newton–Cotes rule for $m = 3$,

$$I = \frac{3h}{8}(f_0 + 3f_1 + 3f_2 + f_3) + R_T, \qquad h = (b-a)/3,$$

also known as Simpson's (3/8)-rule.

(b) Derive the open Newton–Cotes rule for $m = 4$,

$$I = \frac{4h}{3}(2f_1 - f_2 + 2f_3) + R_T, \qquad h = (b-a)/4.$$

(c) Find asymptotic error estimates for the formulas in (a) and (b) by applying them to suitable polynomials.

2. (a) Show that Simpson's formula is the unique quadrature formula of the form

$$\int_{-h}^{h} f(x)\,dx \approx h(a_{-1}f(-h) + a_0 f(0) + a_1 f(h))$$

that is exact whenever $f \in \mathcal{P}_4$. Try to find several derivations of Simpson's formula, with or without the use of difference operators.

(b) Find the Peano kernel $K_2(u)$, such that $Rf = \int_{\mathbf{R}} f''(u)K_2(u)\,du$, and find the best constants c, p, such that

$$|Rf| \le ch^p \max|f''(u)|, \quad \forall f \in C^2[-h,h].$$

If you are going to deal with functions that are not in C^3, would you still prefer Simpson's formula to the trapezoidal rule?

3. The quadrature formula

$$\int_{x_{i-1}}^{x_{i+1}} f(x)\,dx \approx h\big(af(x_{i-1}) + bf(x_i) + cf(x_{i+1})\big) + h^2 df'(x_i),$$

can be interpreted as a Hermite interpolatory formula with a *double point* at x_i. Show that $d = 0$ and that this formula is identical to Simpson's rule. Then show that the error can be written as

$$R(f) = \frac{1}{4!}\int_{x_{i-1}}^{x_{i+1}} f^{(4)}(\xi_x)(x - x_{i-1})(x - x_i)^2(x - x_{i+1})\,dx,$$

where $f^{(4)}(\xi_x)$ is a continuous function of x. Deduce the error formula for Simpson's rule. Setting $x = x_i + ht$, we get

$$R(f) = \frac{h^4}{24}f^{(4)}(\xi_i)\int_{-1}^{1}(t+1)t^2(t-1)h\,dt = \frac{h^5}{90}f^{(4)}(\xi_i).$$

4. A second kind of Newton–Cotes" open quadrature rule uses the midpoints of the equidistant grid $x_i = ih$, $i = 1 : n$, i.e.

$$\int_{x_0}^{x_n} f(x)\,dx = \sum_{i=1}^{n} w_i f_{i-1/2}, \quad x_{i-1/2} = \tfrac{1}{2}(x_{i-1} + x_i).$$

(a) For $n = 1$ we get the midpoint rule. Determine the weights in this formula for $n = 3$ and $n = 5$. (Use symmetry!)

(b) What is the order of accuracy of these two rules?

5. Derive Simpson's formula with end corrections, i.e. a formula of the form

$$\int_{-h}^{h} f(x)\,dx \approx h\big(af(-h) + bf(0) + af(h)\big) + h^2 c(f'(-h) - f'(h)),$$

that is exact for polynomials of degree five. What is the corresponding composite formula for the interval $[a, b]$ with $b - a = 2nh$?

6. Compute the integral

$$\frac{1}{2\pi}\int_0^{2\pi} e^{\frac{1}{\sqrt{2}}\sin x}\,dx$$

by the trapezoidal rule, using $h = \pi/2$ and $h = \pi/4$.

7. Compute the integral $\displaystyle\int_0^{\infty} (1 + x^2)^{-4/3}\,dx$ with five correct decimals. Expand the integrand in powers of x^{-1} and integrate term-wise over the interval $[R, \infty]$, for a suitable value of R. Then use a Newton–Cotes' rule on the remaining interval $[0, R]$.

8. Write a program for the derivation of a formula for integrals of the form $I = \int_0^1 x^{-1/2} f(x)\,dx$ that is exact for $f \in \mathcal{P}_n$ and uses the values $f(x_i)$, $i = 1 : n$, by means of the power basis.

(a) Compute the coefficients b_i for $n = 6 : 8$ with equidistant points, $x_i = (i - 1)/(n - 1)$, $i = 1 : n$. Apply the formulas to the integrals

$$\int_0^1 x^{-1/2} e^{-x}\,dx; \qquad \int_0^1 \frac{dx}{\sin\sqrt{x}}; \qquad \int_0^1 (1 - t^3)^{-1/2}\,dt.$$

In the first of the integrals compare with the result obtained by series expansion in Problem 3.1.1. In the last integral a substitution is needed for bringing it to the right form.

(b) Do the same for the case, where the step size $x_{i+1} - x_i$ grows proportionally to i; $x_1 = 0$; $x_n = 1$. Is the accuracy significantly different compared to (a), for the same number of points?

(c) Make some very small random perturbations of the x_i, $i = 1 : n$ in (a), (say) of the order of 10^{-13}. Of which order of magnitude are the changes in the coefficients b_i , and the changes in the results for the first of the integrals?

9. Propose a suitable plan (using a computer) for computing the following integrals, for $s = 0.5, 0.6, 0.7, \ldots, 3.0$:

(a) $\int_0^\infty (x^3 + sx)^{-1/2}\,dx$; (b) $\int_0^\infty (x^2+1)^{-1/2}e^{-sx}\,dx$, error $< 10^{-6}$;

(c) $\int_\pi^\infty (s+x)^{-1/3}\sin x\,dx$.

10. It is not true that any degree of accuracy can be obtained by using a Newton–Cotes' formula of sufficiently high order. To show this, Compute approximations to the integral

$$\int_{-4}^{4} \frac{dx}{1+x^2} = 2\tan^{-1} 4 \approx 2.6516353\ldots.$$

using the closed Newton–Cotes' formula with $n = 2, 4, 6, 8$. Which formula gives the smallest error?

11. For expressing integrals appearing in the solution of certain integral equations the following modification of the midpoint rule is often used:

$$\int_{x_0}^{x_n} K(x_j, x)y(x)\,dx = \sum_{i=0}^{n-1} m_{ij}y_{i+1/2},$$

where $y_{i+1/2} = y(\frac{1}{2}(x_i + x_{i+1}))$ and m_{ij} is the moment integral

$$m_{ij} = \int_{x_i}^{x_{i+1}} K(x_j, x)\,dx.$$

Derive an error estimate for this formula.

12. (a) Suppose that you have found a truncated δ^2-expansion, (say) $A(\delta^2) \equiv a_1 + a_2\delta^2 + \ldots + a_{k+1}\delta^{2k}$. Then an equivalent symmetric expression of the form $B(E) \equiv b_1 + b_2(E + E^{-1}) + \ldots + b_{k+1}(E^k + E^{-k})$ can be obtained as $b = M_{k+1}a$, where a, b are column vectors for the coefficients, and M_{k+1} is the $(k+1) \times (k+1)$ submatrix of the matrix M given in (3.2.45).
Use this for deriving (5.1.27) from (5.1.26). How do you obtain the remainder term? If you obtain the coefficients as decimal fractions, multiply them by 14175/4 in order to check that they agree with (5.1.27).

(b) Use Cauchy+FFT for deriving (5.1.26), and the open formula and the remainder for the same interval.

(c) Set $z_n = \nabla^{-1}y_n - \Delta^{-1}y_0$. We have, in the literature, seen the interpretation that $z_n = \sum_{j=0}^{n} y_j$ if $n \geq 0$. It seems to require some extra conditions to be true. Investigate if the conditions $z_{-1} = y_{-1} = 0$ are necessary and sufficient. Can you suggest better conditions? (The equations $\Delta\Delta^{-1} = \nabla\nabla^{-1} = 1$ mentioned earlier are assumed to be true.)

13. (a) Write a program for the derivation of quadrature formulas and error estimates according to Example 5.1.2 for $m = n-1, n, n+1$. Test the formulas and the error estimates for some m, n on some simple (though not too simple)

examples. Some of these formulas are listed in Handbook of Mathematical Functions [1, Sec. 25.4]. In particular, check the closed Newton–Cotes' 9-point formula ($n = 8$). .

(b) Sketch a program for the case that $h = 1/(2n + 1)$, with the computation of f at $2m$ symmetrical points.

(c) Abramowitz–Stegun [1, Sec. 25.4] gives several Newton–Cotes formulas of closed and open types, with remainders. Try to reproduce and extend their tables with techniques related to Example 5.2.2.

5.2 Quadrature Rules with Free Nodes

Previously we have assumed that all nodes x_i of the quadrature formula are given. A natural questions is whether we can do better by a judicious choice of the free nodes. This question is answered in the following theorem, which shows that by a careful choice of grid points the order of accuracy of the quadrature rule can substantially improved.

Theorem 5.2.1 (Gautschi [16] Theorem 3.2.1).

Let k be an integer such that $0 \le k \le n$. Consider the quadrature rule (5.1.1) *and let*

$$s(x) = (x - x_1)(x - x_2) \cdots (x - x_n) \tag{5.2.1}$$

be the corresponding **node polynomial**. *Then the quadrature rule has degree of exactness equal to $d = n + k - 1$, if and only if the following two conditions are satisfied:*

(a) *The quadrature rule* (5.1.1) *is interpolatory, i.e. the coefficients C_i are given by* (5.1.4).

(b) *The node polynomial satisfies*

$$\int_a^b p(x)s(x)w(x)\,dx = 0, \tag{5.2.2}$$

for all polynomials $p \in \mathcal{P}_k$.

Proof. We first prove the *necessity* of the conditions (a) and (b). Since the degree of exactness is $d = n+k-1 \ge n-1$, the condition (a) follows immediately. Further, for any $p \in \mathcal{P}_k$ the product $p(x)s(x)$ is in $\mathcal{P}_{n+k}$. Hence

$$\int_a^b p(x)s(x)w(x)\,dx = \sum_{j=1}^{n} w_k f(x_k)s(x_k) = 0.$$

since $s(x_k) = 0$, $k = 1 : n$, so that (b) holds.

To prove the *sufficiency*, let $p(x)$ be any polynomial of degree $n + k - 1$. Let $q(x)$ and $r(x)$ be the quotient and remainder, respectively, in the division

$$f(x) = q(x)s_n(x) + r(x).$$

Then $q(x)$ and r are polynomials of degree $k-1$ and $n-1$, respectively, and it holds that

$$\int_a^b p(x)w(x)\,dx = \int_a^b q(x)s_n(x)w(x)\,dx + \int_a^b r(x)w(x)\,dx.$$

Here the first integral is zero because of the orthogonality property of $s(x)$. For the second we have

$$\sum_{i=1}^n w_i p(x_i) = \sum_{i=1}^n w_i q(x_i)s_n(x_i) + \sum_{i=1}^n w_i r(x_i) = \sum_{i=1}^n w_i r(x_i),$$

since $s_n(x_i) = 0$, $i = 1:n$. But

$$\int_a^b r(x)w(x)\,dx = \sum_{i=1}^n w_i r(x_i),$$

since the weights were chosen such that the formula was interpolatory and therefore exact for all polynomials of degree $n-1$. □

In the previous section we derived Newton–Cotes' quadrature rules using Lagrange interpolation or operator series. We now outline another general technique, the method of undetermined coefficients, for determining approximate quadrature formulas of maximum order.

Let L be a linear functional and consider approximation formulas of the form

$$Lf \approx \tilde{L}f = \sum_{i=1}^p a_i f(x_i) + \sum_{j=1}^q b_j f(z_j), \tag{5.2.3}$$

where the x_i are p *given nodes*, while the z_j are q *free nodes*. The latter are to be determined together with the weight factors a_i, b_j. The altogether $p+2q$ parameters in the formula are to be determined, if possible, so that the formula becomes exact for all polynomials of degree less than $p+2q$.

We introduce the two node polynomials

$$r(x) = (x-x_1)\cdots(x-x_p), \qquad s(x) = (x-z_1)\cdots(x-z_q), \tag{5.2.4}$$

of degree p and q, respectively.

Let $\phi_1, \phi_2, \ldots, \phi_N$ be a basis of the space of polynomials of degree less than N. We assume that the quantities $L\phi_k$, $k = 1:p+2q$ are known. Then we obtain the *non-linear* system,

$$\sum_{i=1}^p \phi_k(x_i)a_i + \sum_{j=1}^q \phi_k(z_j)b_j = L\phi_k(x), \quad k = 1,2,\ldots,p+2q, \tag{5.2.5}$$

This is a non-linear system in z_j, but of a very special type. Note that the free nodes z_j appear in a symmetric fashion; the system (5.2.5) is invariant with respect

to permutations of the free nodes together with their weights. We therefore first ask for their **elementary symmetric functions**, i.e. for the coefficients g_j of the node polynomial

$$s(x) = \phi_{q+1}(x) - \sum_{j=1}^{q} s_j \phi_j(x) \tag{5.2.6}$$

that has the free nodes $z_1, z_2, \dots z_q$ as zeros. We change the basis to the set

$$\phi_1(x), \dots \phi_q(x), s(x)\phi_1(x), \dots, s(x)\phi_{p+q}(x).$$

In the system (5.2.5), the equations for $k = 1 : q$ will not be changed, but the equations for $k = 1 + q : p + 2q$ become,

$$\sum_{i=1}^{p} \phi_{k'}(x_i)s(x_i)a_i + \sum_{j=1}^{q} \phi_{k'}(z_j)s(z_j)b_j = L(s\phi_{k'}), \quad 1 \le k' \le p + q. \tag{5.2.7}$$

Here the second sum disappears since $s(z_j) = 0$, for all j. (This is the nice feature of this treatment!) Further by (5.2.6)

$$L(s\phi_{k'}) = L(\phi_{k'}\phi_{q+1}) - \sum_{j=1}^{q} L(\phi_{k'}\phi_j)s_j, \quad 1 \le k' \le p + q. \tag{5.2.8}$$

We thus obtain the following *linear* system for the computation of the $q + p$ quantities, s_j, and $A_i = s(x_i)a_i$:

$$\sum_{j=1}^{q} L(\phi_{k'}\phi_j)s_j + \sum_{i=1}^{p} \phi_{k'}(x_i)A_i = L(\phi_{k'}\phi_{q+1}), \quad k' = 1 : p + q. \tag{5.2.9}$$

The weights of the fixed nodes are $a_i = A_i/s(x_i)$. The free nodes z_j are then determined by finding the q roots of the polynomial

$$s(x) = \phi_{q+1}(x) - \sum_{j=1}^{q} s_j \phi_j(x) = 0.$$

(Methods for computing roots of a polynomial are given in Sec. 6.5.) Finally, with a_i and z_j known, the weights b_j are obtained by the solution of the first q equations of the system (5.2.5). which are linear in b_j.

Let $p = 0$, $[a, b] = [0, b]$, (b may be infinite) and consider the monomial basis. The reader is advised to verify that, when $p > 0$ the matrix becomes a kind of combination of a Hankel matrix and a Vandermonde matrix. In this case the condition number of of the linear system (5.2.5) increases exponentially with $p+2q$ and the free nodes and corresponding weights may become rather inaccurate. It is usually found, however, that unless the condition number is so big that the solution breaks down completely, the computed solution will satisfy equation (5.2.5) with a small residual. That is what really matters for the application of formula (5.2.3).

Example 5.2.1.

Consider the linear functional $L(f) = \int_0^1 f(x)\,dx$. Set $p = 0$, $q = 3$ and choose the monomial basis $\phi_i(x) = x^{i-1}$. Introducing the node polynomial

$$s(x) = (x - z_1)(x - z_2)(x - z_3) = x^3 - s_3x^2 - s_2x - s_1,$$

the linear system (5.2.8) becomes

$$\begin{pmatrix} 1 & 1/2 & 1/3 \\ 1/2 & 1/3 & 1/4 \\ 1/3 & 1/4 & 1/5 \end{pmatrix} \begin{pmatrix} s_1 \\ s_2 \\ s_3 \end{pmatrix} = \begin{pmatrix} 1/4 \\ 1/5 \\ 1/6 \end{pmatrix}.$$

The exact solution is $s_1 = 1/20$, $s_2 = -3/5$, and $s_3 = 3/2$. The free nodes thus are the zeros of $s(x) = x^3 - 3x^2/2 + 3x/5 - 1/20$, which are $z_2 = 1/2$ and $z_{1,3} = 1/2 \pm \sqrt{3/20}$. The weights b_1, b_2, b_3 are then found by solving (5.2.5) for $k = 1 : 3$.

For the purpose of error estimation, we can add the two equations

$$\sum_{i=1}^{p} \phi_k(x_i)\, a_i + \sum_{j=1}^{q} \phi_k(z_j)\, b_j + (k-1)!c_{k-1} = L\phi_k, \qquad (5.2.10)$$
$$k = p + 2q + 1, p + 2q + 2,$$

The remainder term of the method is of the form $c_N(Lf_N - \tilde{L}f_N)$, where f_N is any monic polynomial of degree N. Normally $N = p + 2q$, but this is inadequate if $c_{p+2q} = 0$. This exceptional case actually happens, if a certain kind of symmetry is present. The formula is then more accurate than expected, and we take $N = p + 2q + 1$ instead. This is why c_k is to be computed for two values of k.

For the determination of the error constant we compute, according to the comments to (5.2.10), the difference between the right hand side and the left hand side of (5.2.9), and divide by $(k')!$, for $k' = p + q + 1$, $p + q + 2$.

From a pure mathematical point of view all bases are equivalent, but equation (5.2.5) may be better conditioned with some bases than with others, and this turns out to be an important issue when $p + 2q$ is large. The simplest choice of basis is

$$\phi_k(x) = x^{k-1}, \quad x \in (0, b),$$

(b may be infinite). For this choice the condition number of (5.2.5), will increase exponentially with $p + 2q$.

In the case $[a, b] = [-b, b]$, where the weight function $w(x)$ and the given nodes x_i are symmetrical with respect to the origin it holds that $L(\phi_k(x)) = 0$, when k is even. Then the weights a_i and b_i, and the free nodes z_j will also be symmetrically located. If $p = 2p'$ is even, the number of parameters will be reduced to $p' + q$ by the transformation

$$x = \sqrt{\xi}, \quad \xi \in [0, b^2].$$

Note that $w(x)$ will be replaced by $w(\sqrt{\xi})/\sqrt{\xi}$. If p is odd, one node is at the origin, and one can proceed in an analogous way. This should also reduce the condition number approximately to its square root, and it is possible to derive in a numerically stable way formulas with about twice as high order of accuracy as in the unsymmetric case.

5.2.1 Gauss–Christoffel Quadrature

By Theorem 5.2.1 if the n nodes in a quadrature formula are chosen so that the node polynomial $s(x) = (x - x_1)(x - x_2) \cdots (x - x_n)$ satisfies

$$\int_a^b p(x)s(x)w(x)\,dx = 0, \quad \forall\, p(x) \in \mathcal{P}_n, \tag{5.2.11}$$

then the corresponding interpolatory quadrature rule has the maximum possible order of accuracy $2n - 1$. These formulas are called **Gauss' quadrature** formulas associated with the weight function w. The construction of such quadrature rules is closely related to the theory of orthogonal polynomials. For the weight function $w(x) \equiv 1$ they were derived in 1814 by Gauss [11]. Formulas for more general weight functions were given by Christoffel[4] [5] in 1858, which is why these are referred to as **Gauss–Christoffel quadrature** formulas.

We denote by

$$(f, g) = \int_a^b f(x)g(x)w(x)\,dx, \tag{5.2.12}$$

the inner product with respect to the weight function $w(x) \geq 0$ and the interval $[a, b]$. The corresponding norm is $(f, f) = \|f\|_2^2$. This inner product has the important property that

$$(xf, g) = (f, xg). \tag{5.2.13}$$

We recall the assumption that $w(x) \geq 0$ is a weight function on $[a, b]$ such that moments

$$\mu_k = (x^k, 1) = \int_a^b x^k w(x)\,dx.$$

are defined for all $k \geq 0$, and $\mu_0 > 0$.

The zeros of these polynomials then determine the nodes in the corresponding Gaussian formula. The weights are then determined by integrating the elementary Lagrange polynomials (5.1.4)

$$w_i = \int_a^b \ell_i(x)w(x)\,dx, \quad \ell_i(x) = \prod_{\substack{j=1 \\ j\neq i}}^{n} \frac{(x - x_j)}{(x_i - x_j)}.$$

In Sec. 5.2.3 we will discuss a more stable algorithm that determines the nodes and weights directly from the coefficients in the recurrence relation (**??**).

The condition (5.2.11) for the node polynomial can now be interpreted to mean that $s(x)$ is orthogonal to all polynomials in $\mathcal{P}_n$. We shall now prove some important results from the general theory of orthogonal polynomials.

[4] Elvin Bruno Christoffel (1829–1900) worked mostly in Strasbourg. He is best known for his work in geometry and tensor analysis, which Einsten later used in his theory of relativity.

Theorem 5.2.2.

The roots x_i, $i = 1 : n$, of the orthogonal polynomial polynomial φ_{n+1} of degree n, associated with the weight function $w(x) \geq 0$ on $[a, b]$, are real, distinct and contained in the open interval (a, b).

Proof. Let $a < x_1 < x_2 \cdots < x_m$, be the roots of φ_{n+1} of odd multiplicity, which lie in (a, b). At these roots φ_n changes sign and therefore the polynomial $q(x)\varphi_{n+1}$, where

$$q(x) = (x - x_1)(x - x_2) \cdots (x - x_m),$$

has constant sign in $[a, b]$. Hence,

$$\int_a^b \varphi_{n+1} q(x) w(x)\, dx > 0.$$

But this is possible only if the degree of $q(x)$ is equal to n. Thus $m = n$ and the theorem follows. □

Corollary 5.2.3.

If $x_1, x_2, \ldots, x_n$ are chosen as the n distinct zeros of the orthogonal polynomial φ_{n+1} of degree n in the family of orthogonal polynomials associated with $w(x)$, then the formula

$$\int_a^b f(x) w(x)\, dx \approx w_1 f_1 + w_2 f_2 + \ldots + w_n f_n, \tag{5.2.14}$$

$$w_i = \int_a^b \ell_i(x) w(x)\, dx, \tag{5.2.15}$$

is exact for polynomials of degree $2n - 1$.

Apart from having optimal degree of exactness equal to $2n - 1$, Gaussian quadrature rules have several important properties, which we now outline.

Theorem 5.2.4.

All weights in a Gaussian quadrature rule are real, distinct and positive.

Proof. Let

$$\ell_i(x) = \prod_{\substack{j=1 \\ j \neq i}}^{n} \frac{(x - x_j)}{(x_i - x_j)}, \quad i = 1 : n,$$

be the Lagrange polynomials. Then the quadrature formula (5.2.14) is exact for $p(x) = (\ell_i(x))^2$, which is of degree $2(n-1)$. Further $\ell_i(x_j) = 0$, $j \neq i$, and therefore

$$\int_a^b (\ell_i(x))^2 w(x)\, dx = w_i (\ell_i(x_i))^2 = w_i.$$

Since $w(x) > 0$ it follows that $w_i > 0$. □

Gaussian quadrature formulas can also be derived by Hermite interpolation on the nodes x_k, *each counted as a double node*, and requiring that coefficients of the derivative terms should be zero. This interpretation gives a convenient expression for the error term in Gaussian quadrature.

Theorem 5.2.5.
The remainder term in Gauss' quadrature is given by the formula

$$\frac{f^{(2n)}(\xi)}{(2n)!}\int_a^b \Big[\prod_{i=1}^n (x-x_i)\Big]^2 w(x)\,dx = c_n f^{(2n)}(\xi), \quad a<\xi<b. \tag{5.2.16}$$

The constant c_n can be determined by applying the formula to some polynomial of degree $2n$.

Proof. Denote by $q(x)$ the polynomial of degree $2n-1$ which solves the Hermite interpolation problem (see Sec. 4.3.1)

$$q(x_i) = f(x_i), \qquad q'(x_i) = f'(x_i), \qquad i = 1:n.$$

The Gauss quadrature formula is exact for $q(x)$, and hence

$$\int_a^b q(x)w(x)\,dx = \sum_{i=1}^n w_i q(x_i) = \sum_{i=1}^n w_i f(x_i).$$

Thus

$$\sum_{i=1}^n w_i f(x_i) - \int_a^b f(x)w(x)\,dx = \int_a^b (q(x)-f(x))w(x)\,dx.$$

Using the remainder term (4.3.4) in Hermite interpolation gives

$$f(x)-q(x) = \frac{f^{(2n)}(\xi)}{(2n)!}(\varphi_n(x))^2, \quad \varphi_n(x) = \prod_{i=1}^n (x-x_i).$$

and the theorem now follows. □

5.2.2 Applications of Gauss Quadrature

For the uniform weight distribution $w(x) = 1$ on $[-1,1]$ the relevant orthogonal polynomials are the **Legendre polynomials** $P_n(x)$. As a historical aside, Gauss derived his quadrature formula by considering the continued fraction

$$\tfrac{1}{2}\ln\left(\frac{z+1}{z-1}\right) = \tfrac{1}{2}\int_{-1}^1 \frac{dx}{z-x} = \frac{1}{z-}\,\frac{1/3}{z-}\cdots \tag{5.2.17}$$

$$= \frac{1}{z} + \frac{1}{3z^3} + \frac{1}{5z^5} + \cdots \tag{5.2.18}$$

The nth convergent of this continued fraction is a rational function with a numerator of degree $n-1$ in z and denominator of degree n, which is the $(n-1,n)$ Padé approximation to the function. Decomposing this fraction in partial fractions the residues and the poles can be taken as nodes of a quadrature formula. The denominators are precisely the **Legendre polynomials**. Using the accuracy properties of the Padé approximants Gauss showed that the formula will have order $2n-1$. For more on this interesting connections between Padé approximants and orthogonal polynomials see Brezinski [3].

Since the weight distribution is symmetric about the origin the Legendre polynomials have the symmetry property

$$P_n(-x) = (-1)^n P_n(x).$$

They satisfy the three-term recurrence formula $P_0(x) = 1$, $P_1(x) = x$,

$$P_{n+1}(x) = \frac{2n+1}{n+1} x P_n(x) - \frac{n}{n+1} P_{n-1}(x), \quad n \geq 1. \tag{5.2.19}$$

giving

$$P_2(x) = \tfrac{1}{2}(3x^2-1), \qquad P_3(x) = \tfrac{1}{2}(5x^3-3x), \ldots \tag{5.2.20}$$

The Legendre polynomials have leading coefficient

$$A_n = \frac{1}{2^n n!} 2n(2n-1)(2n-2)\ldots(n+1).$$

and $\|P_n\| = 2/(2n+1)$.

The Legendre polynomials can also be defined by

$$P_0(x) = 1, \qquad P_n(x) = \frac{1}{2^n n!} \frac{d^n}{dx^n}\big((x^2-1)^n\big), \qquad n = 1, 2, \ldots \tag{9.3.21}$$

Since $(x^2-1)^n$ is a polynomial of degree $2n$, $P_n(x)$ is a polynomial of degree n. The extreme values are

$$|P_n(x)| \leq 1, \quad x \in [-1, 1].$$

There seems to be no easy proof for this result; see Henrici [1964, p. 219].

Example 5.2.2.

Derive a two-point Gauss quadrature rule for $\int_{-1}^{1} f(x)\,dx$. Here $w(x) = 1$, and the relevant orthogonal polynomials are the Legendre polynomials $P_{m+1}(x)$. For $m = 1$ we have $P_2(x) = \frac{1}{2}(3x^2-1)$, and hence $x_0 = -3^{-1/2}$, $x_1 = 3^{-1/2}$. The weights can be determined by application of the formula to $f(x) = 1$ and $f(x) = x$, respectively, i.e.,

$$w_0 + w_1 = 2, \qquad -3^{-1/2} w_0 + 3^{-1/2} w_1 = 0,$$

with solution $w_0 = w_1 = 1$. Hence the formula

$$\int_{-1}^{1} f(x)\,dx \approx f(-3^{-1/2}) + f(3^{-1/2})$$

Table 5.2.1. *Abscissas and weight factors for Gauss–Legendre quadrature from Abramowitz–Stegun* [1, Table 25.4].

x_i	w_i
$n = 4$	
±0.33998 10435 84856	0.65214 51548 62546
±0.86113 63115 94053	0.34785 48451 37454
$n = 5$	
0.00000 00000 00000	0.56888 88888 88889
±0.53846 93101 05683	0.47862 86704 99366
±0.90617 98459 38664	0.23692 68850 56189
$n = 6$	
±0.23861 91860 83197	0.46791 39345 72691
±0.66120 93864 66265	0.36076 15730 48139
±0.93246 95142 03152	0.17132 44923 79170

Table 5.2.2. *Summary of Gaussian quadrature rules*

interval $[a, b]$	weight function $w(x)$	abscissas zeros of	polynomials
$[-1, 1]$	1	$P_n(x)$	Legendre
$[-1, 1]$	$(1-x^2)^{-1/2}$	$T_n(x)$	Chebyshev 1st kind
$[-1, 1]$	$(1-x)^\alpha(1+x)^\beta$	$J_n(x;\alpha,\beta)$	Jacobi
$[-1, 1]$	$(1-x^2)^{1/2}$	$U_n(x)$	Chebyshev, 2nd kind
$[0, \infty]$	e^{-x}	$L_n(x)$	Laguerre
$[-\infty, \infty]$	e^{-x^2}	$H_n(x)$	Hermite

is exact for polynomials of third degree.

In order to use the Gaussian quadrature rule the abscissas and weight factors must be known numerically. Note that for $w(x) = 1$ and the interval $[-1, 1]$ the abscissas are symmetric with respect to the origin. The two-point formulas was given above; for the three-point formula see Problem 1 below. In Table 5.2.1 we give abscissas and weights for some higher order Gauss–Legendre formulas using $n = m + 1$ points.

The **Jacobi polynomials** $J_n(x;\alpha,\beta)$ arise from the weight function

$$w(x) = (1-x)^\alpha(1+x)^\beta, \quad x \in [-1, 1], \quad \alpha, \beta > -1,$$

They are special cases of Gauss hypergeometric function $F(a, b, c : x)$

$$F(-n, \alpha + 1 + \beta + n, \alpha + 1; x).$$

(see (3.1.12)). The Jacobi polynomials are usually defined so that the coefficient A_n of x^n in $J_n(x; \alpha, \beta)$ is given by

$$A_n = \frac{1}{2^n n!} \frac{\Gamma(2n + \alpha + \beta + 1}{\Gamma(n + \alpha + \beta + 1}.$$

We obtain the Gauss–Legendre quadrature formula as the special case when $\alpha = \beta = 0$. Further the case $\alpha = \beta = -1/2$, which corresponds to $w(x) = 1/\sqrt{1 - x^2}$, give the Gauss–Chebyshev quadrature formula. These and some other important Gaussian quadrature rules are summarized in Table 5.2.2.

The above rules are given for the standard interval $[-1, 1]$. The corresponding formula for an integral over the interval $[a, b]$ is obtained by the change of variable $t = \frac{1}{2}((b - a)x + (a + b))$, which maps the interval $[a, b]$ onto $[-1, 1]$, so that

$$\int_a^b f(t)dt = \frac{b - a}{2} \int_{-1}^1 g(x)\, dx, \qquad g(x) = f\left(\frac{1}{2}\big((b - a)x + (a + b)\big)\right).$$

If $f(t)$ is a polynomial then $g(x)$ will be a polynomial of the same degree, since the transformation is linear. Hence the order of accuracy of the formula is not affected.

Two other important cases of Gauss quadrature rules are the following: For the weight function

$$w(x) = e^{-x}, \quad 0 \le x < \infty,$$

the corresponding orthogonal polynomials are the **Laguerre polynomials**, which satisfy

$$L_n(x) = e^x \frac{d^n}{dx^n}(x^n e^{-x}).$$

The **Hermite polynomials** are orthogonal with respect to the weight function

$$w(x) = e^{-x^2}, \quad -\infty < x < \infty.$$

They satisfy the recurrence relation $H_0(x) = 1$, $H_1(x) = 2x$,

$$H_{n+1}(x) = 2xH_n(x) - 2nH_{n-1}(x).$$

The Hermite polynomials can also be defined by the formula.

$$H_n(x) = (-1)^n e^{x^2} \frac{d^n}{dx^n} e^{-x^2}$$

It can be verified that these polynomials are identical to those defined by the recurrence relation.

In some situations we want some of the abscissas x_i in the quadrature formula to be fixed; the rest are to be chosen freely to maximize the order of accuracy. In the

most common cases the preassigned abscissas are at the endpoints of the interval. We consider here quadrature rules of the form

$$Lf = \sum_{i=1}^{n} w_i f(x_i) + \sum_{j=1}^{m} b_j f(z_j) + E(f) \tag{5.2.21}$$

where z_j, $j = 1 : m$ are fixed nodes and the x_i, a_i and b_j are to be determined. By a generalization of Theorem 5.2.5 the remainder term is given by the formula

$$E(f) = \frac{f^{(2n+m)}(\xi)}{(2n)!} \int_a^b \prod_{i=1}^{m}(x - z_i)\Big[\prod_{i=1}^{n}(x - x_i)\Big]^2 w(x)\,dx, \quad a < \xi < b. \tag{5.2.22}$$

In **Gauss–Lobatto** quadrature $m = 2$, and both endpoints are used as abscissas, $z_1 = a$, $z_2 = b$. Taking $[a, b] = [-1, 1]$ and and the weight function $w(x) = 1$, the quadrature formula has the form

$$\int_{-1}^{1} f(x)\,dx = b_1 f(-1) + b_2 f(1) + \sum_{i=1}^{n} w_i f(x_i) + E_L. \tag{5.2.23}$$

where

$$b_1 = b_2 = 2/((n + 2)(n + 1)).$$

The remaining n abscissas are the zeros $P'_{n+1}(x)$, where $P_n(x)$ denotes the Legendre polynomial. They lie symmetric with respect to the origin. The corresponding weights are given by

$$w_i = b_1/(P_{n+1}(x_i))^2,$$

and satisfy $w_i = w_{n+1-i}$. Because two points are fixed we lose two degrees of accuracy and the Lobatto rule (5.2.23) is exact only for polynomials of order $2m-1$. If $f(x) \in C^{2m}[-1, 1]$ then the error term is given by

$$E_L(f) = -\frac{(n + 2)(n + 1)^3 2^{2n+3}(n!)^4}{(2n + 3)[(2n + 2)!]^3} f^{(2n+2)}(\xi), \quad \xi \in (-1, 1). \tag{5.2.24}$$

Nodes and weights for Lobatto quadrature are found in Abramowitz–Stegun [1, Table 25.6].

Example 5.2.3.

The simplest Gauss–Lobatto rule is Simpson's rule with one interior node. Taking $n = 2$ the interior nodes are the zeros of $\phi_2(x)$, where

$$\int_{-1}^{1} (1 - x^2)\phi_2(x)p(x)\,dx = 0, \quad \forall p \in P_2.$$

Thus, ϕ_2 is, up to a constant factor, the Jacobi polynomial $J_2(x, 1, 1) = (x^2 - 1/5)$. Hence the interior nodes are $\pm 1/\sqrt{5}$ and the quadrature formula becomes

$$\int_{-1}^{1} f(x)\,dx = \frac{1}{6}(f(-1) + f(1)) + \frac{5}{6}(f(-1/\sqrt{5}) + f(1/\sqrt{5})) + R(f), \tag{5.2.25}$$

where $R(f) = 0$ for $f \in P_6$.

In **Gauss–Radau** quadrature rules one of the endpoints ± 1 is taken as abscissa, $z_0 = -1$, say. The quadrature formula has the form

$$\int_{-1}^{1} f(x)\,dx = \frac{2}{(n+1)^2} f(-1) + \sum_{i=1}^{n} w_i f(x_i) + E_{R1}. \tag{5.2.26}$$

The n free abscissas are the zeros of

$$\frac{P_n(x) + P_{n+1}(x)}{x - 1},$$

where $P_m(x)$ are the Legendre polynomials. The corresponding weights are given by

$$w_i = \frac{1}{(n+1)^2} \frac{1 - x_i}{(P_n(x_i))^2}.$$

The Gauss–Radau quadrature rule is exact for polynomials of order $2n + 1$. If $f(x) \in C^{2m-1}[-1, 1]$ then the error term is given by

$$E_{R1}(f) = \frac{(n+1)2^{2n+1}}{[(2n+1)!]^3} (n!)^4 f^{(2n+1)}(\xi_1), \quad \xi_1 \in (-1, 1). \tag{5.2.27}$$

A similar formula can be obtained with the fixed point $+1$ by making the substitution $t = -x$. From the error term (5.2.22) it follows that if the derivative $f^{(n+1)}(x)$ has constant sign in $[a, b]$, then the error will have opposite sign. This can be used to obtain lower and upper bounds for the true integral.

A drawback with Gaussian rules is that as we increase the order of the formula all interior abscissas change, except that at the origin. Hence we cannot use the function values computed for the lower order formula. For this reason Kronrod [21] considered the following problem: Given an n-point Gaussian quadrature rule

$$G_n \approx \sum_{i=0}^{n-1} w_i f(x_i),$$

find a new formula using the n old abscissas x_i and $n + 1$ new abscissas y_i

$$K_{2n+1} \approx \sum_{i=0}^{n-1} A_i f(x_i) + \sum_{i=0}^{n} B_i f(y_i).$$

The new abscissas and the weights A_i and B_i are to be chosen so that the rule K_{2n+1} is exact for polynomials of degree $3n + 1$.

The two rules (G_n, K_{2n+1}) are called a **Gauss–Kronrod** pair. Note that the number of new function evaluations are the same as for the Gauss rule G_{n+1}. The error can be estimated by the difference $|G_n - K_{2n+1}|$, but this usually severely overestimates the error.

Gauss–Kronrod rules is one of most effective methods for calculating integrals. Often one takes $n = 7$ and uses the Gauss–Kronrod pair (G_7, K_{15}), together with the realistic but still conservative error estimate $(200|G_n - K_{2n+1}|)^{1.5}$, see Kahaner, Moler, and Nash [20].

A Kronrod extension of the Gauss–Lobatto rule (5.2.25) has been given by Gander and Gautschi [9]:

$$\int_{-1}^{1} f(x)\,dx = \frac{11}{210}(f(-1)+f(1)) + \frac{72}{245}(f(-\sqrt{2/3})+f(\sqrt{2/3})) + \frac{125}{294}(f(-1/\sqrt{5})+f(1/\sqrt{5})) + \frac{16}{35}f(0)) + R(f). \quad (5.2.28)$$

This rule is exact for all $f \in \mathcal{P}_{10}$. Note that the Kronrod points $\pm\sqrt{2/3}$ and 0 interlace the previous nodes.

5.2.3 Matrix Formulas Related to Gauss Quadrature

We collect here some classical results of Gauss, Christoffel, Chebyshev, Stieltjes and others, with a few modern aspects and a notations appropriate for our purpose.

Let $\{p_1, p_2, \ldots, p_n\}$ be a basis for the space $\mathcal{P}_n$, of polynomials of degree $n-1$, where p_j be a polynomial of exact degree $j-1$. We introduce the row vector

$$\pi(x) = [p_1(x),\, p_1(x), \ldots, p_n(x)], \quad (5.2.29)$$

containing these basis functions. The **modified moments** with respect to the basis $\pi(x)$ are

$$\nu_k = (p_k, 1) = \int_a^b p_k(x)w(x)\,dx, \quad k = 1:n, \quad (5.2.30)$$

We define the two symmetric matrices

$$G = \int \pi(x)^T\pi(x)w(x)\,dx, \qquad \hat{G} = \int x\pi(x)^T\pi(x)w(x)\,dx. \quad (5.2.31)$$

associated with the basis defined by π. Here G is the **Gram matrix**[5] with elements $g_{ij} = (p_i, p_j) = (p_j, p_i)$,

$$G = \begin{pmatrix} (p_1,p_1) & (p_1,p_2) & \ldots & (p_1,p_n) \\ (p_2,p_1) & (p_2,p_2) & \ldots & (p_2,p_n) \\ \vdots & \vdots & \ddots & \vdots \\ (p_n,p_1) & (p_n,p_2) & \ldots & (p_n,p_n) \end{pmatrix}. \quad (5.2.32)$$

Two particularly interesting bases are the power basis and the orthonormal basis defined, respectively, by

$$\theta(x) = (1, x, x^2, \ldots, x^{n-1}), \quad (5.2.33)$$

$$\varphi(x) = (\phi_1(x), \phi_2(x), \ldots, \phi_n(x)), \quad (5.2.34)$$

[5] Jørgen Pedersen Gram (1850–1916) graduated from Copenhagen University and then worked as company director for a life insurance company. He introduced the Gram determinant in connection with his study of linear independence and his name is also associated with Gram–Schmidt orthogonalization.

where the components of φ are orthonormal polynomials with respect to the weight function w.

Example 5.2.4.

For the power basis $\theta(x)$ we have $g_{ij} = (x^{i-1}, x^{j-1}) = \mu_{i+j-2}$. So the matrices G and $\hat{G}$ become Hankel matrices,

$$G = \begin{pmatrix} \mu_0 & \mu_1 & \cdots & \mu_{n-1} \\ \mu_1 & \mu_2 & \cdots & \mu_n \\ \vdots & \vdots & \cdots & \vdots \\ \mu_{n-1} & \mu_n & \cdots & \mu_{2n-2} \end{pmatrix}, \qquad \hat{G} = \begin{pmatrix} \mu_1 & \mu_2 & \cdots & \mu_n \\ \mu_2 & \mu_3 & \cdots & \mu_{n+1} \\ \vdots & \vdots & \cdots & \vdots \\ \mu_n & \mu_{n+1} & \cdots & \mu_{2n-1} \end{pmatrix}.$$

In particular, for $w(x) \equiv 1$, and $[a, b] = [0, 1]$ we have $\mu_k = \int_0^1 x^{k-1}\, dx = 1/k$ and G is the notoriously ill-conditioned *Hilbert matrix*, for which the spectral condition number grows like $0.014 \cdot 10^{1.5n}$.

Let u, v, be two polynomials in $\mathcal{P}_n$ and set

$$u(x) = \pi(x)u_\pi, \qquad v(x) = \pi(x)v_\pi,$$

where u_π, v_π, are column vectors with the coefficients in the representation of u, v with respect to the basis defined by π. Note that $(u, v) = u_\pi^T G v_\pi$. For $u = v \neq 0$ we find that

$$u_\pi^T G u_\pi = (u, u) > 0,$$

hence the Gram matrix G is positive definite. (The matrix $\hat{G}$ is, however, usually indefinite.)

A polynomial of degree n that is orthogonal to all polynomials of degree less than n can be written in the form

$$\phi_{n+1}(x) = xp_n(x) - \pi(x)c_n, \quad c_n \in \mathbf{R}^n, \tag{5.2.35}$$

Here c_n is determined by the linear equations

$$-\int \pi(x)^T \pi(x) c_n w(x)\, dx + \int x\pi(x)^T p_n(x) w(x)\, dx = 0,$$

or in matrix form

$$Gc_n = \hat{g}_n, \tag{5.2.36}$$

where $\hat{g}_n$ is the last column of the matrix $\hat{G}$. Further, there are coefficients c_{kj} depending on the basis only, such that

$$xp_j(x) = \sum_{k=1}^{j+1} c_{k,j} p_k(x), \quad j = 1 : n-1.$$

Together with (5.2.35) this can be summarized in the vector equation

$$x\pi(x) = \pi(x)(C, c_n) + (0, 0, \ldots, \phi_{n+1}(x)). \tag{5.2.37}$$

Here $C \in \mathbf{R}^{n\times(n-1]}$ is an upper Hessenberg matrix, which depends on the basis only, while c_n also depends on the weight function. If the basis $\pi(x)$ is some family of orthogonal polynomials (with respect to another weight function than w) C is a tridiagonal matrix, obtained by means of the three-term recurrence relation for this family.

After multiplication of (5.2.37) by $\pi(x)^T w(x)$ and integration we obtain by (5.2.31)

$$G\overline{C} = \hat{G}, \quad \overline{C} = (C, c_n). \tag{5.2.38}$$

where the last column of this equation is the same as equation (5.2.36). Let G^*, C^* be defined like G, C, with n increased by one. Note that G and C are principal submatrices of G^* and C^*. Then $\hat{G}$ equals the n first rows of the product G^*C^*. So no integrations are needed for g_n, except for the matrix G.

Theorem 5.2.6.

Denote by R the matrix of coefficients of the expansions of the general basis functions $\pi(x) = [p_1(x), p_1(x), \ldots, p_n(x)]$ into the orthonormal basis polynomials, i.e.

$$\pi(x) = \varphi(x)R. \tag{5.2.39}$$

Then $G = R^T R$, i.e. R is the upper triangular Cholesky factor of the Gram matrix G. Note that this factorization up to the mth row is the same for all $n \geq m$. Further $\hat{G} = R^T JR$, where J is a symmetric tridiagonal matrix.

Proof. R is evidently an upper triangular matrix. Further, we have

$$\begin{aligned} G = \int \pi(x)^T \pi(x) w(x)\, dx &= \int R^T \varphi(x)^T \varphi(x) R w(x)\, dx \\ &= R^T I R = R^T R, \end{aligned}$$

since the elements of $\varphi(x)$ is an orthonormal system. This shows that R is the Cholesky factor of G. We similarly find that

$$\hat{G} = R^T JR, \quad J = \int x\varphi(x)^T \varphi(x) w(x)\, dx,$$

so J clearly is a symmetrical matrix. J is a particular case of $\hat{G}$ and from (5.2.38) and $G = I$ it follows that $J = \overline{C}$, a Hessenberg matrix. Hence J is a symmetric tridiagonal matrix. □

From (5.2.38) and Theorem 5.2.6 it follows that

$$\hat{G} = G\overline{C} = R^T R\overline{C} = R^T JR,$$

Since R is nonsingular we have $R\overline{C} = JR$, or

$$J = R\overline{C}R^{-1}. \tag{5.2.40}$$

It follows that, for every choice of basis, the spectrum of $\overline{C}$ equals the spectrum of J. We shall see that it is equal to the set of zeros of the orthogonal polynomial ϕ_n.

In particular, for the power basis $p_j(x) = x^{j-1}$, the Hessenberg matrix C is a **shift matrix**; the only non-zero elements are ones in the first main subdiagonal. Further, with $c_n^T = (a_1, a_2, \ldots, a_n)$, (5.2.35) reads

$$\phi_{n+1}(x) = x^n - \sum_{k=1}^{n} a_k x^{k-1},$$

and

$$\overline{C} = \begin{pmatrix} 0 & & & & a_1 \\ 1 & 0 & & & a_2 \\ & 1 & \ddots & & \vdots \\ & & \ddots & 0 & a_{n-1} \\ & & & 1 & a_n \end{pmatrix} \in \mathbf{R}^{n\times n},$$

which (after a permutation of rows and columns) is the companion matrix of the polynomial $\phi_{n+1}(x)$ (see Sec. 6.5.1). *Thus the eigenvalues λ_j, $j = 1 : n$, of $\overline{C}$ are the zeros of $\phi_{n+1}(x)$*, and hence the nodes for the Gauss–Christoffel quadrature formula. The row eigenvector corresponding to λ_j is

$$\theta(\lambda_j) = (1, \lambda_j, \lambda_j^2, \ldots, \lambda_j^{n-1}), \tag{5.2.41}$$

i.e. it holds that

$$\theta(\lambda_j)\overline{C} = \lambda_j \theta(\lambda_j), \quad j = 1 : n. \tag{5.2.42}$$

This yields a diagonalization of $\overline{C}$, since, by the general theory of orthogonal polynomials (see Theorem 5.2.4) the roots are simple roots, located in the interior of the smallest interval that contains the weight distribution.

To summarize, we have shown that if C and the Gram matrix G are known, then c_n can be computed by performing the Cholesky decomposition $G = R^T R$ and then solving $R^T R c_n = \hat{g}_n$ for c_n. The zeros of $\phi_{n+1}(x)$ are then equal to the eigenvalues of $\overline{C} = (C, c_n)$, or equivalently the eigenvalues of the symmetric tridiagonal matrix $J = R\overline{C}R^{-1}$. *This is true for any basis $\pi(x)$.* Note that J can be computed by solving the matrix equation $JR = R\overline{C}$ or

$$R^T J = (R\overline{C})^T. \tag{5.2.43}$$

Here R^T is a lower triangular matrix and the right hand side a lower Hessenberg matrix. This and the tridiagonal structure of J considerably simplifies the calculation of J.

For the power basis $\theta(x)$ we saw in Example 5.2.4 that G is a Hankel matrix. Hankel matrices play an important role in the classical theory of orthogonal polynomials, Gauss–Christoffel quadrature, the moment problem, continued fractions, etc. They are less interesting for practical computations since the condition number of H increases rapidly with n. This is due to the by now familiar fact that, when n is large, x^n can be accurately approximated by a polynomial of lower degree. The power basis is thus not a good basis for spaces of polynomials. Similarly the moments for the power basis are not in general a good starting point for the numerical computation of the matrix J.

In particular, for the orthonormal basis, for which $G = I$, and $\hat{G} = G^{-1}\hat{G} = J$, we obtain

$$\varphi(\lambda_j)J = \lambda_j\varphi(\lambda_j), \quad j = 1:n. \tag{5.2.44}$$

where

$$J = \begin{pmatrix} \beta_1 & \gamma_1 & & & 0 \\ \gamma_1 & \beta_2 & \gamma_2 & & \\ & \gamma_2 & & \ddots & \\ & & \ddots & \ddots & \gamma_{n-1} \\ 0 & & & \gamma_{n-1} & \beta_n \end{pmatrix}, \tag{5.2.45}$$

is a symmetric tridiagonal **Jacobi matrix** with nonzero off-diagonal elements. It is well known from linear algebra that such a matrix has n real distinct eigenvalues. Further, the eigenvectors can always be chosen mutually orthogonal.

Setting

$$\Phi = (\varphi(\lambda_1)^T, \ldots, \varphi(\lambda_n)^T), \qquad \Phi_{ij} = (\phi_{i-1}(\lambda_j), \qquad \Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_n),$$

we obtain by (5.2.44) and the symmetry of J the important formula

$$J\Phi = \Phi\Lambda. \tag{5.2.46}$$

It also follows from (5.2.44) that the equation

$$x\varphi(x)^T = J\varphi(x)^T + \gamma_n\phi_{n+1}(x)e_n, \quad e_n = (0, \ldots, 0, 1)^T, \tag{5.2.47}$$

where γ_n is to be chosen so that $\|\phi_{n+1}\| = 1$, holds when $x = \lambda_j$, and $\phi_{n+1}(\lambda_j) = 0$, $j = 1:n$. Since the degree of φ is less than n, it is easily shown that the *equation* (5.2.47) *holds for all* x. As a by-product we obtain the important three term recurrence (**??**)

Let V be an orthogonal matrix that diagonalizes J, i.e.

$$JV = V\Lambda, \quad V^TV = VV^T = I,$$

where Λ is the diagonal in (5.2.46). It follows that $V = \Phi D$ for some diagonal matrix $D = \operatorname{diag}(d_i)$, and

$$V = \Phi D^2\Phi^T = VV^T = I,$$

that is

$$\sum_{k=1}^{n} \phi_i(\lambda_k)d_k^2\phi_j(\lambda_k) = \delta_{ij} = (\phi_i, \phi_j), \quad i, j = 1:n.$$

This equality holds also for $i = n+1$, because $\phi n + 1(\lambda_k) = 0$, for all k, and $(\phi_{n+1}, \phi_j) = 0$, $j = 1:n$.

Since every polynomial p of degree less than $2n$ can be expressed as a linear combination of polynomials of the form $\phi_i\phi_j$ (in infinitely many ways) it follows that

$$\sum_{k=1}^{n} d_k^2 p(\lambda_k) = \int p(x)w(x)\,dx, \tag{5.2.48}$$

for any polynomial p of degree less than $2n$. This yields the **Gauss–Christoffel quadrature rule**:

$$\int f(x)w(x)\,dx = \sum_{k=1}^{n} d_k^2 f(\lambda_k) + R, \tag{5.2.49}$$

where $R = \int (f(x) - p(x))w(x)\,dx$, *for any polynomial* p *of degree less than* $2n$, *such that* $p(\lambda_k) = f(\lambda_k)$, $k = 1 : n$.

The familiar form for the remainder term

$$R = k_n f^{(2n)}(\xi)/(2n)!, \tag{5.2.50}$$

is obtained by choosing a Hermite interpolation polynomial for p and then applying the mean value theorem. The constant k_n is independent of f. The choice $f(x) = A_n^2 x^{2n} + \cdots$ gives $k_n = A_n^{-2}$. A recurrence relation for the leading coefficient A_j is obtained by (**??**). We obtain

$$A_0 = \mu_0^{-1/2}, \quad A_{k+1} = A_k/\gamma_k. \tag{5.2.51}$$

The mean value form for R may be inappropriate, when the interval is infinite. Some other estimate of the above integral for R may then be more adequate.

A simple formula for the weights d_k^2, due to Golub and Welsch [18], is obtained by matching the first rows of the equality $V = \Phi D$. Since the elements in the first row are all equal to the constant $\phi_1 = \mu_0^{-1/2}$, we obtain

$$e_1^T V = \mu_0^{-1/2} d^T, \qquad d_k^2 = \mu_0 v_{1,k}^2, \quad k = 1 : n. \tag{5.2.52}$$

The well known fact that the weights are positive and their sum equals μ_0, follows immediately from this simple formula for the weights. We summarize these results in the following theorem:

Theorem 5.2.7.

Let J *be the symmetric tridiagonal* $n \times n$ *matrix that contains the coefficients in the three term recurrence relation for the orthonormal system of polynomials associated with the weight function* $w(x) > 0$. *Let* f *be an analytic function in a domain that contains the spectrum of* J.

Then the following concise formula, is exact when f *is a polynomial of degree less than* $2n$,

$$\frac{1}{\mu_0} \int f(x)w(x)\,dx \approx e_1^T V^T f(\Lambda) V e_1, \tag{5.2.53}$$

where $f(\Lambda) = \operatorname{diag}(f(\lambda_1, \ldots, f(\lambda_n)$.

Proof. The result follows from the Gauss–Christoffel rule (5.2.49) and (5.2.52). □

When the three-term recurrence relation for the orthonormal polynomials associated with the weight function $w(x)$ is known, the Gauss–Christoffel rule can

elegantly be obtained as follows. The eigenvalues of J are the nodes of the Gauss–Christoffel rule and the weights are obtained from (5.2.52) as the first components of the corresponding eigenvectors.. These quantities can be computed in a stable and efficient way by the QR-algorithm; see Volume II, Sec. 9.7.4. We remark that Golub [17] has shown how to extend this scheme to the computation of nodes and weights for Gauss–Radau and Gauss–Lobatto quadrature rules.

When the coefficients in the three-term relation cannot be obtained by theoretical analysis or numerical computation, we consider the matrices $\overline{C}$ and G as given data about the basis and weight function. As described above R, c_n, and J can then be computed by means of (5.2.40). The nodes and weights are then computed according to the previous case. Note that R and J are determined simultaneously for all $k \leq n$; just take the submatrices of the largest ones.

The computations are most straightforward for the power basis, i.e. with the moments of the weight function as the initial data. Unfortunately, the condition number of this problem increases rapidly with n, which results in inaccurate nodes and weights. Nevertheless, as long as the Choleski factorization of the Gram matrix G does not break down because of a negative pivot, the values of the integral give by the Gauss–Christoffel formula may be much more accurate than the nodes and weights obtained.

5.2.4 Symmetric Weight Functions

In many important cases the weight function $w(x)$ is symmetric about the origin. Then the moments of odd order are zero, and the orthogonal polynomials of odd (even) degree are odd (even) functions. The eigenvalues will appear in pairs, $\pm\lambda_k$. If n is odd, there is also a simple zero eigenvalue. The weights are symmetric so that the weights corresponding to the two eigenvalues $\pm\lambda_i$ are the same.

We shall see that in the symmetric case the eigenvalue problem for the tridiagonal matrix $J \in \mathbf{R}^{n\times n}$ can be reduced to a singular value problem for smaller bidiagonal matrix B, where

$$B \in \begin{cases} \mathbf{R}^{n/2\times n/2}, & \text{if } n \text{ even;} \\ \mathbf{R}^{(n+1)/2\times(n-1)/2}, & \text{if } n \text{ odd.} \end{cases}$$

We permute rows and columns in J, by an odd-even permutation, e.g., if $n = 7$ then $(1, 2, 3, 4, 5, 6, 7) \mapsto (1, 3, 5, 7, 2, 4, 6)$, and

$$\tilde{J} = T^{-1}JT = \begin{pmatrix} 0 & B \\ B^T & 0 \end{pmatrix}, \quad B = \begin{pmatrix} \beta_1 & 0 & 0 \\ \beta_2 & \beta_3 & 0 \\ 0 & \beta_4 & \beta_5 \\ 0 & 0 & \beta_6 \end{pmatrix},$$

where T be the permutation matrix effecting the odd-even permutation. Then, if the orthogonal matrix V diagonalizes J, i.e. $J = V\Lambda V^T$, then $\tilde{V} = T^{-1}V$, diagonalizes $\tilde{J} = T^TJT$, i.e. $\tilde{J} = T^{-1}JT = T^{-1}V\lambda V^TT$. Note that the first row of V is just a permutation of $\tilde{V}$. We can therefore substitute $\tilde{V}$ for V in equation (5.2.52) that gives the weights in the Gauss–Christoffel formula.

The following relationship between the SVD and a Hermitian eigenvalue problem, exploited by Lanczos [22, Chap. 3] can easily be verified.

Theorem 5.2.8.
Let the singular value decomposition of $B \in \mathbf{R}^{m\times n}$ $(m \geq n)$ be $B = P\Sigma Q^T$, where

$$\Sigma = \operatorname{diag}(\Sigma_1,\ 0), \quad \Sigma_1 = \operatorname{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n),$$

and

$$P = (P_1,\ P_2) \in \mathbf{C}^{m\times m}, \quad P_1 \in \mathbf{C}^{m\times n}, \quad Q \in \mathbf{C}^{n\times n}.$$

Then the symmetric matrix $C \in \mathbf{R}^{(m+n)\times(m+n)}$ has the eigendecomposition

$$C = \begin{pmatrix} 0 & B \\ B^T & 0 \end{pmatrix} = V \begin{pmatrix} \Sigma_1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -\Sigma_1 \end{pmatrix} V^T, \tag{5.2.54}$$

where $V \in$ is orthogonal

$$V = \frac{1}{\sqrt{2}} \begin{pmatrix} P_1 & \sqrt{2}\,P_2 & P_1 \\ Q & 0 & -Q \end{pmatrix}^T. \tag{5.2.55}$$

Hence the eigenvalues of C are $\pm\sigma_1, \pm\sigma_2, \ldots, \pm\sigma_r$, and zero repeated $(m-n)$ times.

The QR-algorithm for symmetric tridiagonal matrices can be adopted to compute the singular values σ_i and the first components of the matrix P of singular vectors of the bidiagonal matrix B; see Vol. II, Sec. 9.7.6.

Review Questions

1. What are orthogonal polynomials? Give a few examples of families of orthogonal polynomials together with the three-term recursion formula, which its members satisfy.
2. Formulate and prove a theorem concerning the location of zeros of orthogonal polynomials.
3. Give an account of Gauss quadrature formulas: accuracy, how the nodes and weights are determined. What important properties are satisfied by the weights?
4. What is the orthogonality property of the Legendre polynomials?

Problems and Computer Exercises

1. Prove that the three-point quadrature formula

$$\int_{-1}^{1} f(x)\,dx \approx \frac{1}{9}\big(5f(-\sqrt{3/5}) + 8f(0) + 5f(\sqrt{3/5})\big),$$

is exact for polynomials of degree 5. Apply it to the computation of

$$\int_0^1 \frac{\sin x}{1+x}\,dx,$$

and estimate the error in the result.

2. (a) Calculate the Hermite polynomials H_n for $n \leq 4$ using the recurrence relation.

(b) Express, conversely, $1, x, x^2, x^3, x^4$ in terms of the Hermite polynomials.

3. Determine the orthogonal polynomials $\phi_n(x)$, $n = 1, 2, 3$, with leading coefficient 1, for the weight function $w(x) = 1 + x^2$, $x \in [-1, 1]$.

(b) Give a two-point Gaussian quadrature formula for integrals of the form

$$\int_{-1}^1 f(x)(1+x^2)\,dx,$$

which is exact when $f(x)$ is a polynomial of degree three.

Hint: Either use the method of undetermined coefficients taking advantage of symmetry, or the three term recurrence relation in Theorem **??**.

4. (W. Gautschi) (a) Construct the quadratic polynomial ϕ_2 orthogonal on $[0, \infty]$ with respect to the weight function $w(x) = e^{-x}$. *Hint*: Use $\int_0^\infty t^m e^{-t}\,dt = m!$.

(b) Obtain the two-point Gauss–Laguerre quadrature formula

$$\int_0^\infty f(x)e^{-x}\,dx = w_1 f(x_1) + w_2 f(x_2) + E_2(f),$$

including a representation for the remainder $E_2(f)$.

(c) Apply the formula in (b) to approximate

$$I = \int_0^\infty (x+1)^{-1} e^{-x}\,dx.$$

Use the remainder term to estimate the error, and compare your estimate with the true error ($I = 0.596347361\ldots$).

5. Show that the formula

$$\int_{-1}^1 f(x)(1-x^2)^{-1/2}\,dx = \frac{\pi}{n}\sum_{k=1}^n f\Big(\cos\frac{2k-1}{2n}\pi\Big)$$

is exact for all polynomials of degree $2n-1$.

6. Derive the Gauss–Lobatto quadrature rule in Example 5.2.3, with two interior points by using the Ansatz

$$\int_{-1}^1 f(x)\,dx = w_1(f(-1)+f(1)) + w_2(f(-x_1)+f(x_1)),$$

and requiring that it be exact for $f(x) = 1, x^2, x^4$.

7. Compute an approximate value of

$$\int_{-1}^{1} x^4 \sin^2 \pi x \, dx = 2 \int_{0}^{1} x^4 \sin^2 \pi x \, dx,$$

using the 5 point Gauss–Legendre quadrature rule on $[0, 1]$ for the weight function $w(x) = 1$. For nodes and weights see Table 5.2.1. (The true value of the integral is 0.11407 77897 39689.)

8. Let $\mu_j = \int_a^b x^j w(x)\, dx$ be the jth **moment** of the weight distribution w. Show that the system of equations

$$\begin{pmatrix} \mu_0 & \mu_1 & \cdots & \mu_{n-1} \\ \mu_1 & \mu_2 & \cdots & \mu_n \\ \vdots & \vdots & \cdots & \vdots \\ \mu_{n-1} & \mu_n & \cdots & \mu_{2n-2} \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_{n-1} \end{pmatrix} = - \begin{pmatrix} \mu_n \\ \mu_{n+1} \\ \vdots \\ \mu_{2n-1} \end{pmatrix}$$

has as solution the coefficient of a polynomial $x^n + \sum_{j=1}^{n} a_j x^{j-1}$, which is a member of the family of orthogonal polynomials associated with the weight function w.

9. (a) Determine exactly the Lobatto formulas with given nodes at -1 and 1, (and the remaining nodes free), for the weight functions $w(x) = (1 - x^2)^{-\frac{1}{2}}$, $x \in [-1, 1]$. Determine for this weight function also the nodes and weights for the Gauss quadrature formula (i.e. when all nodes are free).

Hint: Set $x = \cos\phi$, and formulate equivalent problems on the unit circle. Note that you obtain (at least) two different discrete orthogonality properties of the Chebyshev polynomials this way.

(b) Lobatto–Kronrod pairs are useful when a long interval has been divided into several shorter intervals (cf. Example 5.2.28). Determine Lobatto–Kronrod pairs (exactly) for $w(x) = (1 - x^2)^{-\frac{1}{2}}$.

10. Apply the formulas in Problem 9 to the case $w(x) = 1$, $x \in [-1, 1]$ and some of the following functions:

(a) $f(x) = e^{kx}$, $k = 1, 2, 4, 8, \ldots$; (b) $f(x) = 1/(k + x)$, $k = 1, 2, 1.1, 1.01$;

(c) $f(x) = k/(1 + k^2x^2)$, $k = 1, 4, 16, 64$.

Compare the actual errors with the error estimates.

11. Write a MATLAB function for the evaluation of the Sievert[6] integral,

$$S(x, \theta) = \int_0^{\theta} e^{-x/\cos\phi}\, d\phi,$$

for any $x \geq 0$, $x \leq \theta \leq 90°$, with at least six decimals relative accuracy. There may be useful hints in Abramowitz–Stegun [1, § 27.4].

[6] Sievert was a Swedish radio-physicist, who was so great that doses of radiation are measured in millisievert, or even microsievert, all over the world.

5.3 Extrapolation Methods

5.3.1 Euler–Maclaurin Formula

Although Newton–Cotes' rules of high orders of accuracy are known, they have the drawback that they do not provide a convenient way of estimating the error. Also, for high order rules negative weights appear. In this section we will derive formulas of high order, based on Euler–Maclaurin's formula (see Sec. 3.5), which do not share these drawbacks.

According to Theorem 3.5.2, if $f \in C^{2r+2}[a,b]$, then

$$T(a:h:b)f - \int_a^b f(x)\,dx = \frac{h^2}{12}\big(f'(b) - f'(a)\big) - \frac{h^4}{720}\big(f'''(b) - f'''(a)\big) + \ldots + \frac{B_{2r}h^{2r}}{(2r)!}\big(f^{(2r-1)}(b) - f^{(2r-1)}(a)\big) + R_{2r+2}(a,h,b)f.$$

Here $x_i = a + ih$, $x_n = b$, and $T(a:h:b)f$ denotes the trapezoidal sum

$$T(a:h:b)f = \sum_{i=1}^{n} \frac{h}{2}\big(f(x_{i-1}) + f(x_i)\big).$$

The remainder $R_{2r+2}(a,h,b)f$ is $O(h^{2r+2})$ is represented by an integral with a kernel of constant sign in (3.5.8). The estimation of the remainder is very simple in certain important particular cases. Note that although the expansion contains derivatives at the boundary points only, the remainder requires tha $|f^{(2r+2)}|$ is integrable on the interval $[a,b]$.

One easily shows the following simple and useful relation of the trapezoidal sum to the midpoint sum

$$R(a,h,b)f = \sum_{i=1}^{n} hf(x_{i-1/2}) = 2T(a:\tfrac{1}{2}h:b)f - T(a:h:b)f. \tag{5.3.1}$$

From this one easily derives the expansion

$$R(a,h,b)f = \int_a^b f(x)\,dx - \frac{h^2}{24}\big(f'(b) - f'(a)\big) + \frac{7h^4}{5760}\big(f'''(b) - f'''(a)\big) + \ldots + \Big(\frac{1}{2^{2r-1}} - 1\Big)\frac{B_{2r}h^{2r}}{(2r)!}\big(f^{(2r-1)}(b) - f^{(2r-1)}(a)\big) + \ldots,$$

which has the same relation to the midpoint sum as the Euler–Maclaurin Formula has to the trapezoidal sum.

The Euler–Maclaurin formulas can be used for highly accurate numerical integration when the values of derivatives of f are known at $x = a$ and $x = b$. It also possible to use difference approximations to estimate the derivatives needed.

A variant with uncentered differences, is **Gregory's**[7] **quadrature formula**

$$\int_a^b f(x)\,dx = h\frac{E^n-1}{hD}f_0 = h\left(\frac{f_n}{-\ln(1-\nabla)} - \frac{f_0}{\ln(1+\Delta)}\right)$$
$$= T(a;h;b) + h\sum_{j=1}^{\infty} a_{j+1}(\nabla^j f_n + (-\Delta)^j f_0),$$

where $T(a:h:b)$ is the trapezoidal sum, as defined in the Euler–Maclaurin Formula. The operator expansion must be truncated at $\nabla^k f_n$ and $\Delta^l f_0$, where $k \le n$, $l \le n$. Concerning the interpretation of ∇^{-1} and Δ^{-1}, see Problem 3.2.13(d).

5.3.2 Romberg's Method

The Euler–Maclaurin formula is the theoretical basis for the application of repeated Richardson extrapolation (see Sec. 3.5.2) to the results of the trapezoidal rule. This method, introduced in [29], is known as **Romberg's method**. It is one of the most widely used methods, because it allows a simple strategy for the automatic determination of a suitable step size and order. A thorough analysis of Romberg's method was carried out by Bauer, Rutishauser and Stiefel [2, 1963] that we shall refer to for proof details.

Let $f \in C^{2m+2}[a,b]$ be a real function to be integrated over $[a,b]$. Set $x_i = a + ih$, $x_n = b$, and denote by

$$T(h)f = \sum_{i=1}^{n} \frac{h}{2}\big(f(x_{i-1}) + f(x_i)\big).$$

the trapezoidal. Then by the Euler–Maclaurin's formula it follows that

$$T(h) - \int_a^b f(x)\,dx = c_2h^2 + c_4h^4 + \cdots + c_mh^{2m} + \tau_{m+1}(h)h^{2m+2},$$

where $c_k = 0$ if $f \in \mathcal{P}_k$. This suggests the use of Repeated Richardson extrapolation applied to the trapezoidal sums computed with step lengths

$$h_0 = \frac{b-a}{n_0}, \quad h_1 = \frac{h_0}{n_0}, \quad \ldots, \quad h_m = \frac{h_{m-1}}{n_m}, \tag{5.3.2}$$

where $n_1, n_2, \ldots, n_m$ are strictly increasing positive integers. Romberg used the special sequence

$$h_i = (b-a)/2^i.$$

In this case Richardson extrapolation can be used with headings $\Delta/3$, $\Delta/15$, $\Delta/63, \ldots$.

[7] James Gregory (1638–1675), Scotch mathematician. This formula was discovered long before the Euler–Maclaurin formula, and seems to have been primarily used for numerical quadrature. It can be used also for summation, but the variants with central differences are typically more efficient.

By (5.3.1) we have the relation

$$T(h/2) = \frac{1}{2}(T(h) + R(h)), \quad R(h)f = \sum_{i=1}^{n} hf(x_{i-1/2}) \tag{5.3.3}$$

where $R(h)$ is he midpoint sum. This makes it possible to reuse the function values that have been computed earlier.

For practical numerical calculations the values of the coefficients c_k are not needed, but they are used, e.g., in the derivation of an error bound, see Theorem 5.3.1. It is also important to remember that the coefficients depend on derivatives of increasing order; the success of repeated Richardson extrapolations is thus related to the behavior in $[a, b]$ of the higher derivatives of the integrand.

According to the discussion of repeated Richardson extrapolation in Sec. 3.5.2, one continues the process, until two values *in the same row* agree to the desired accuracy. If no other error estimate is available, $\min_k |T_{m,k} - T_{m,k-1}|$ is usually chosen as an estimate of the truncation error, even though it is usually a strong overestimate. A feature of the Romberg algorithm is that it also contains exits with lower accuracy at a lower cost.

If the use of the basic asymptotic expansion is doubtful, then the uppermost diagonal of the extrapolation scheme should be ignored, except for its element in the first column. Such a case is detected by inspection of the difference quotients in a column. If for some k, where $T_{k+2,k}$ has been computed and the modulus of the relative irregular error of $T_{k+2,k} - T_{k+1,k}$ is less than (say) 20%, *and*, most important, the difference quotient $(T_{k+1,k} - T_{k,k})/(T_{k+2,k} - T_{k+1,k})$ is is very different from its theoretical value q^{p_k}, then the uppermost diagonal is to be ignored (except for its first element).

Example 5.3.1. *A numerical illustration to Romberg's method.*

Use Romberg's method to compute the integral (cf. Example 5.1.1)

$$\int_0^{0.8} \frac{\sin x}{x}\, dx.$$

The correct value, to ten decimals, is 0.7720957855.

The midpoint and trapezoidal sums computed using IEEE double precision are given below

h	$R(h)f$	$T(h)f$
0.8	0.77883 66846 1730	0.75867 80454 4976
0.4	0.77376 69771 8681	0.76875 73650 3353
0.2	0.77251 27161 1197	0.77126 21711 1017
0.1		0.77188 74436 5335

It can be verified that in this example the error is approximately proportional to h^2 for both $R(h)$ and $T(h)$. We estimate the error in $T(0.1)$ to be $\frac{1}{3}6.26 \cdot 10^{-4} \leq 2.1 \cdot 10^{-4}$.

The trapezoidal sums are then copied to the first column of the Romberg scheme, and repeated extrapolation is applied using the following MATLAB program with $tol = 0.5 \cdot 10^{-10}$ and $q = 5$.

```
function [I, T, md] = romberg(f,a,b,tol,q)
%
% Romberg's method for computing the integral of f over [a,b].
% Stop when two adjacent values in the same column differ by
% less than tol.
%
  T = zeros(q+2,q+1);
  h = b - a;  m = 1; P = 1;
  T(1,1) = h*(feval(f,a) + feval(f,b))/2;
  for i = 2:q+1
    h = h/2; m = 2*m;
    % Compute midpoint sum
    s = 0;
    for j = 1:2:m
      s = s + feval(f, a+j*h)
    end
    R(i-1,1) = 2*h*s;
    T(i,1) = (T(i-1,1) + R(i-1,1))/2;
    % Richardson extrapolation
    jmax = min(i-1,q);
    for j = 1:jmax
      T(i,j+1) = T(i,j) + (T(i,j) - T(i-1,j))/(2^(2*j) - 1)
    end
    % Check accuracy
    [md, jb] = min(abs(T(i,1:jmax) - T(i-1,1:jmax)));
    I = T(i,jb);
    if md <= tol
      T = T(1:i,1:jmax+1); % return active part of T
      return
    end
  end
\vspace{-4mm}
```

The result is given in the table below:

m	T_{m1}	$\Delta/3$	T_{m2}	$\Delta/15$	T_{m3}	T_{44}
1	0.7586780454					
		33597732				
2	0.7687573650		0.7721171382			
		8349354		13355		
3	0.7712621711		0.7720971065		0.7720957710	
		2084242		826		
4	0.7718874437		0.7720958678		0.7720957853	0.7720957855

T_{i1}	T_{i2}	T_{i3}	T_{i4}	T_{i5}
0.758678045450				
0.768757365034	0.772117138228			
0.771262171110	0.772097106469	0.772095771018		
0.771887443653	0.772095867834	0.772095785259	0.772095785485	
0.772043703883	0.772095790626	0.772095785479	0.772095785482	0.772095785482

Since none of the differences $|T_{44} - T_{43}| = 2 \cdot 10^{-10}$, the termination criterion mentioned above requires that the row with $m = 5$ must be computed. Then, the termination criterion is satisfied with a wide margin, since $|T_{55} - T_{54}| = 2.8 \cdot 10^{-12}$, and the irregular errors are less than 10^{-12}. T_{55} is even better than this error bound indicates; the correct result agrees with $T_{55} = 0.772095785482$ to all twelve displayed decimal places.

In cases where the cost of evaluating $F(h)$ is proportional to $1/h$, the standard sequence

$$h_i = (b-a)/n_i, \quad \text{with} \quad n_i = \{1, 2, 4, 8, 16. \ldots\}$$

has the drawback that step sizes decrease rapidly. Bulirsch [4] has proposed the alternative sequence

$$n_i = \{1, 2, 3, 4, 6, 8, 12, 16, 24, \ldots\},$$

for which similar savings can be realized.

In the general case, with $T_{i0} = T(h_i)$ and step lengths given by (5.3.2), repeated Richardson extrapolation using the Neville interpolation scheme takes the form

$$T_{ik} = T_{i,k-1} + \frac{T_{i,k-1} - T_{i-1,k-1}}{(h_{i-k}/h_i)^2 - 1}, \quad 1 \le k \le i \le m.$$

Sometimes rational extrapolation is preferred. This gives rise to a recursion of similar form (see Stoer and Bulirsch [32, Sec. 3.4])

$$T_{ik} = T_{i,k-1} + \frac{T_{i,k-1} - T_{i-1,k-1}}{(h_{i-k}/h_i)^2 \Big[1 - \dfrac{T_{i,k-1} - T_{i-1,k-1}}{T_{i,k-1} - T_{i-1,k-2}}\Big] - 1}, \quad 1 \le k \le i \le m.$$

Theorem 5.3.1. *Error bound for Romberg's method.*

The items T_{mk} in Romberg's method are estimates of the integral $\int_a^{a+h} f(x)\,dx$, that can be expressed as a linear functional,

$$T_{mk} = (b-a) \sum_{j=1}^{n} \alpha_{m,j}^{(k)} f(a + jh), \tag{5.3.4}$$

where $n = 2^{m-1}$, $h = (b-a)/n$, and

$$\sum_{j=1}^{n} \alpha_{m,j}^{(k)} = 1, \qquad \alpha_{m,j}^{(k)} > 0. \tag{5.3.5}$$

The remainder functional for T_{mk} is zero for $f \in \mathcal{P}_{2k}$, and its Peano kernel is positive in the interval (a, b). The truncation error of T_{mk} reads

$$T_{mk} - \int_a^b f(x)dx = r_k h^{2k}(b-a)f^{(2k)}(\tfrac{1}{2}(a+b)) + O(h^{2k+2}(b-a)f^{(2k+2)})$$
$$= r_k h^{2k}(b-a)f^{(2k)}(\xi), \quad \xi \in (a,b), \tag{5.3.6}$$
$$r_k = 2^{k(k-1)}|B_{2k}|/(2k)!, \quad h = 2^{1-m}(b-a).$$

Proof. Sketch: Equation (5.3.4) follows directly from the construction of the Romberg scheme. (It is for theoretical use only; the recursion formulas are better for practical use.) The first formula in (5.3.5) holds, because T_{mk} is exact if $f = 1$. The second formula is easily proved for low values of k. The general proof is more complicated; see [2, Theorem 4].

The Peano kernel for $m = k = 1$ (trapezoidal rule) was constructed in Sec. 3.2. For $m = k = 2$ (Simpson's rule), see Sec. 5.1.2. The general case is more complicated. Recall that, by Corollary 3.3.9 of Peano's Remainder Theorem, a remainder formula with a mean value $\xi \in (a, a+H)$, exists iff the Peano kernel does not change sign.

Bauer, Rutishauser and Stiefel [2, pp. 207–210], constructed a recursion formula for the kernels, and succeeded in proving that they are all positive, by an ingenious use of the recursion. The expression for r_k is also derived there, although with a different notation; see also Problem 3. □

From (5.3.5) it follows that if the magnitude of the irregular error in $f(a+jh)$ is at most ϵ, then the magnitude of the inherited irregular error in T_{mk} is at most $\epsilon(b-a)$.

There is another way of finding r_k. Note that for each value of k, the error of T_{kk} for $f(x) = x^{2k}$ can be determined numerically. Then r_k can be obtained from (5.3.6). T_{mk} is the same formula as T_{kk}, although with a different h.

Sometimes several of the uppermost diagonals are to be ignored. It was mentioned that for the integration of a class of periodic functions the trapezoidal rule is superconvergent. In this case all the difference quotients in the first column are much larger than $q^{p_1} = q^2$. According to the rule just formulated, every element of the Romberg scheme, outside the first column should be ignored. It is all right; *in superconvergent cases Romberg's method is of no use*; it deteriorates the excellent results that the trapezoidal rule has produced. The value $T_{m,k}$ is usually accepted as an estimate of a_0 when $|T_{m,k} - T_{m-1,k}| < \delta$, where δ is the permissible error. Thus one extrapolates until two values *in the same column* agree to the desired accuracy. In most situations, the magnitude of the difference between two values in the same column gives, if h is sufficiently small, with a large margin a bound for the truncation error in the lower of the two values. One cannot, however, get a guaranteed error bound in all situations. Often instead the subdiagonal error criterion $|T_{m,m-1} - T_{m,m}| < \delta$ is used, and T_{mm} taken as the numerical result.

The remainder for the closed Newton–Cotes formulas (with an odd number of

points, i.e., for $k > 0$ in our case), reads

$$d_k h^{2^k+2}(b-a)f^{(2^k+2)}(\xi);$$

for $k = 0$ we have the trapezoidal rule with remainder $d_0 h^2 H f^{(2)}(\xi)$. It follows that for $k = \{0, 1, 2\}$ both methods give, with $k' = \{2, 3, 5\}$, function values, exact results for $f \in \mathcal{P}_{k'}$.[8]

By working algebraically in the Romberg scheme, we obtain the following relations between Romberg's and Newton–Cotes' methods:

$$\begin{aligned}
T_{11} &= \tfrac{1}{2}(b-a)\left(f(a)+f(b)\right),\\
T_{21} &= \frac{1}{4}(b-a)\left(f(a)+f(\tfrac{1}{2}(a+b))+f(\tfrac{1}{2}(a+b))+f(b)\right)\\
&= \tfrac{1}{2}(b-a)\left(\tfrac{1}{2}f(a)+f(\tfrac{1}{2}(a+b))+\tfrac{1}{2}f(b)\right),\\
T_{22} &= \frac{1}{3}(4T_{21}-T_{11}) = \frac{1}{6}(b-a)\left(f(a)+4f(\tfrac{1}{2}(a+b))+f(b)\right). \qquad (5.3.7)
\end{aligned}$$

We see that T_{22} is the same as Simpson's formula. It can also be shown in this way that T_{33} is the same as the five point closed Newton–Cotes formula.

Table 5.3.1. *Data concerning some Romberg and Newton–Cotes formulas.*

$m = k$	n	order T_{kk}	order C_n	error const. r_k	error const. c_n
1	1	2	2	1/12	1/12
2	2	4	4	1/180	1/180
3	4	6	6	2/945	2/945
4	8	8	10	16/4725	296/467775

This equivalence can also be proved by the following argument. By Corollary 3.3.8, there is only one linear combination of the values of the function f at $n+1$ given points that can yield $\int_a^b f(x)\,dx$ exactly for all polynomials $f \in \mathcal{P}_{n+1}$. It follows that the methods of Cotes and Romberg T_{kk} are identical for $k = 0, 1, 2$, but for $k > 2$, $2^k + 2 > 2k + 2$, and the methods are not identical. For $k = 3$ (9 function values), Cotes is exact in $\mathcal{P}_{10}$, while T_{33} is exact in $\mathcal{P}_8$. For $k = 4$ (17 function values), Cotes is exact in $\mathcal{P}_{18}$, while T_{44} is exact in $\mathcal{P}_{10}$; see Table 5.3.1. This sounds like an advantage for Cotes, but one has to be sceptical about formulas that use equidistant points in polynomial approximation of very high degree; see Problem 5 and the discussion of Runge's phenomena in Chapter 4.

Note that the remainder of T_{44} is

$$r_4 h^8 (b-a) f^{(8)}(\xi) \approx r_4 (b-a)\Delta^8 f(a),$$

where $\Delta^8 f(a)$ uses the same function values as T_{44} and C_8. So we can use $r_4(b-a)\Delta^8 f(a)$ as an asymptotically correct error estimate for T_{44}.

[8] For $k = 2, 3$, the results are exact even in $\mathcal{P}_{k'+1}$, due to the symmetry discussed in Example 3.2.7.

When the values in a row of a Richardson scheme converge fast, it is worth to *try, e.g., Aitken extrapolation to this row*, in order to improve the error estimate of the diagonal element $T_{m,m}$. It is important that the irregular errors of the values are small compared to the last Richardson correction. The theoretical support to this is usually rather poor, and the row should therefore contain at least four items, so that one can obtain two Aitken accelerated values. These should not be accepted as results, but they provide two error estimates for $T_{m,m}$. The largest absolute value of these error estimates indicates the order of magnitude of the error of $T_{m,m}$, but it is not a guaranteed error bound. If you want to calculate the indefinite integral of $f(x)$, it may be irritating that the improvements are made only at the endpoints; H may be too big for the application of interpolation of the results afterwards or for graphical output. An idea how to get denser output is developed in [23].

If the function to be integrated has a singularity in the interval. then the expansion no longer is a series in h^2 and Romberg's metod has to be modified.

For example, if the integrand $f(x)$ has an algebraic end-point singularity,

$$f(x) = x^{\beta} h(x), \quad -1 < \beta \leq 0,$$

where $h(x) \in C^{p+1}[a, b]$, then an asymptotic expansion of the form

$$R = \sum_{q=1}^{p} a_q k^{-\beta-q} + \sum_{q=1}^{p} b_q k^{-q} + O(k^{-p-1}) \tag{5.3.8}$$

can be shown to hold for a trapezoidal sum. Similar, but more complicated, expansions can be obtained for other classes of singularities.

The case when the error expansion for the trapezoidal sum has the form

$$T(h) = I + \sum_{m=1}^{n} a_m e_m(h) + R_n(h), \tag{5.3.9}$$

where $e_j(h)$, $j = 1, 2, \ldots$ are known functions satisfying $\lim_{h\to 0} e_{m+1}(h)/e_m(h) = 0$ and the error term satisfies $R_n(h) = O(e_{n+1}(h))$ has been treated by Håvie [19].

Suppose we have computed the trapezoidal sums $T_0^{(k)} = T(h_k)$, for a sequence of steplengths $h_0 > h_1 > h_2 \cdots > h_n > 0$. We want to compute the "best" possible approximation $T_n^{(0)}$ to $I = \lim_{h\to 0} T(h)$, defined by the equations

$$T(h) = T_n^{(0)} + \sum_{m=1}^{n} a'_m e_m(h_k), \quad k = 0 : n. \tag{5.3.10}$$

Håvie showed how the approximations can be computed by a special recurrence relation. Set

$$E_1^{(n)} = \frac{e_1(h_{n+1})T_n - e_1(h_n)T_{n+1}}{e_1(h_{n+1}) - e_1(h_n)}.$$

Replacing T_n and T_{n+1} by their expansion, we obtain

$$e_{1,i}^{(n)} = \frac{e_1(h_{n+1})e_i(h_n) - e_1(h_n)e_i(h_{n+1})}{e_1(h_{n+1}) - e_1(h_n)}.$$

The same process can be repeated for eliminating $e_{1,2}^{(n)}$ in the expansion of $E_1^{(n)}$, and so on. This gives the **E-algorithm**

$$E_k^{(n)} = \frac{e_{k-1.k}^{(n+1)} E_{k-1}^{(n)} - e_{k-1.k}^{(n)} E_{k-1}^{(n+1)}}{e_{k-1.k}^{(n+1)} - e_{k-1.k}^{(n)}}. \tag{5.3.11}$$

The auxiliary quantities $e_{k.i}^{(n)}$ are recursively computed by a similar rule

$$e_{k,i}^{(n)} = \frac{e_{k-1.k}^{(n+1)} e_{k-1,i}^{(n)} - e_{k-1.k}^{(n)} e_{k-1,i}^{(n+1)}}{e_{k-1.k}^{(n+1)} - e_{k-1.k}^{(n)}}, \tag{5.3.12}$$

with $e_{0.i}^{(n)} = e_i(h_n)$. The algorithm can be interpreted in terms of Gaussian elimination for solving the system

$$E_k^{(n)} + b_1 g(h_{n+i}) + \cdots + b_k g(h_{n+i}) = T_{n+i}, \quad i = 0 : k,$$

for the unknown $E_k^{(n)}$.

5.3.3 The Epsilon Algorithm

Richardson extrapolation as used in Romberg's method can only be used to accelerate the rate of convergence if the exponents in the asymptotic expansions are known *explicitly*. In cases when the exponents are unknown a nonlinear extrapolation scheme, like the ϵ-algorithm (see Sec. 3.3.5) has to be used. This is the most important convergence acceleration scheme besides Richardson extrapolation in numerical quadrature.

In the ϵ algorithm a two-dimensional array of numbers $\epsilon_k^{(p)}$ is computed by the recurrence relation,

$$\epsilon_{k+1}^{(p)} = \epsilon_{k-1}^{(p+1)} + \frac{1}{\epsilon_k^{(p+1)} - \epsilon_k^{(p)}}. \tag{5.3.13}$$

using the following boundary conditions

$$\epsilon_{-1}^{(p)} = 0, \quad p = 1, 2, 3, \ldots,$$
$$\epsilon_0^{(p)} = s_p, \quad p = 0, 1, 2, \ldots$$

Example 5.3.2.

Consider the integral

$$\int \sqrt{x}\, dx = 2/3.$$

If Romberg's method is applied to this integral the convergence is very slow. In contrast the ϵ-algorithm is well adapted to accelerating convergence when an asymptotic error expansion of the form (5.3.8) holds.

In the figure below the results from Romberg's method applied to the trapezoidal rule for. This is compared with the results from applying the ϵ-algorithm to the same trapezoidal sums used in Romberg's method. Already for $\epsilon_5^{(0)}$ the order of magnitude of the error is the same as the accumulated rounding error using IEEE double precision.

5.3.4 Infinite Intervals

In general the trapezoidal rule is second order accurate, unless $f'(a) = f'(b)$, but there exist *interesting exceptions*. Suppose that the function f is infinitely differentiable for $x \in \mathbf{R}$, and that f *has* $[a, b]$ *as an interval of periodicity*, i.e.,

$$f(x + (b - a)) = f(x), \quad \forall \quad x \in \mathbf{R}.$$

Then $f^{(k)}(b) = f^{(k)}(a)$, for $k = 0, 1, 2, \ldots$, hence *every term in the Euler–Maclaurin expansion is zero* for the integral over *the whole period* $[a, b]$. One could be led to believe that the trapezoidal rule gives the exact value of the integral, but this is usually not the case; for most periodic functions f, $\lim_{r\to\infty} R_{2r+2} f \neq 0$; *the expansion converges, of course, though not necessarily to the correct result.*

On the other hand, the convergence as $h \to 0$ for a fixed (though arbitrary) r is a different story; the error bound (5.3.6) shows that

$$|R_{2r+2}(a, h, b)f| = O(h^{2r+2}).$$

Since r is arbitrary, this means that *for this class of functions, the trapezoidal error tends to zero faster than any power of* h, *as* $h \to 0$. We may call this **superconvergence**. The application of the trapezoidal rule to an integral over $[0, \infty)$ of a function $f \in C^\infty(0, \infty)$ often yields similar features, sometimes even more striking.

Suppose that the periodic function $f(z)$, $z = x + iy$, is analytic in a strip, $|y| < c$, around the real axis. It can then be shown that the error of the trapezoidal rule is $O(e^{-\eta/h})$ as $h \downarrow 0$; η is related to the width of the strip. A similar result will be obtained in Example 5.3.3, for an annulus instead of a strip.

As a rule, this discussion does *not* apply to periodic functions which are defined by periodic continuation of a function originally defined on $[a, b]$ (like the Bernoulli functions). They usually become non-analytic at a and b, and at all points $a + (b - a)n$, $n = 0, \pm 1, \pm 2, \ldots$.

The **Poisson summation formula** is, even better than the Euler–Maclaurin formula for the quantitative study of the trapezoidal truncation error on an infinite interval. For convenient reference we now formulate the following surprising result:

Theorem 5.3.2. *Suppose that the trapezoidal rule (or, equivalently, the rectangle rule) is applied with* constant *step size* h *to* $\int_{-\infty}^{\infty} f(x)\,dx$. *The Fourier transform of* f *reads*

$$\hat{f}(\omega) = \int_{-\infty}^{\infty} f(x)e^{-i\omega t}\,dt.$$

Then the integration error decreases like $2\hat{f}(2\pi/h)$ *as* $h \downarrow 0$.

Example 5.3.3.

For the normal probability density, we have

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(t/\sigma)^2}, \qquad \hat{f}(\omega) = e^{-\frac{1}{2}(\omega\sigma)^2}.$$

The integration error is thus approximately $2\exp(-2(\pi\sigma/h)^2)$. Roughly speaking, the number of correct digits is doubled if h is divided by $\sqrt{2}$, e.g., the error is approximately $5.4\,10^{-9}$ for $h = \sigma$, and $1.4\,10^{-17}$ for $h = \sigma/\sqrt{2}$.

The application of the trapezoidal rule to an integral over $[0, \infty)$ of a function $f \in C^\infty(0, \infty)$ often yields similar features, sometimes even more striking. Suppose that, for $k = 1, 2, 3, \ldots$,

$$f^{(2k-1)}(0) = 0 \quad \text{and} \quad f^{(2k-1)}(x) \to 0, \quad x \to \infty,$$

and $\int_0^\infty |f^{(2k)}(x)|\,dx < \infty$. (Note that for any function $g \in C^\infty(-\infty, \infty)$ the function $f(x) = g(x) + g(-x)$ satisfies such conditions at the origin.) Then all terms of the Euler–Maclaurin expansion are zero, and one can be misled to believe that the trapezoidal sum gives $\int_0^\infty f(x)\,dx$ exactly for any step size h! We have already seen an example of this in Example 3.5.3. See also Theorem 5.3.2 and Problem 3. The explanation is that the remainder $R_{2r+2}(a, h, \infty)$ will typically not tend to zero, as $r \to \infty$ for fixed h. On the other hand: if we consider the behavior of the truncation error as $h \to 0$ for given r, we find that it is $o(h^{2r})$ for any r, just like the case of a periodic unction.

For a finite subinterval of $[0, \infty)$, however, the remainder is still typically $O(h^2)$, and for each step the remainder is typically $O(h^3)$. So, there is an *enormous cancellation of the local truncation errors*, when a C^∞-function, with vanishing odd-order derivatives at the origin, is integrated by the trapezoidal rule over $[0, \infty)$.

Example 5.3.4.

Infinite intervals of integration occur often in practical problems. For integrals of the form $\int_{-\infty}^\infty f(x)\,dx$, the trapezoidal rule (or the midpoint rule) often gives good accuracy if one integrates over the interval $[-R_1, R_2]$, assuming that $f(x)$ and its lower derivatives are small for $x \le -R_1$ and $x \ge R_2$.

The correct value to six decimal digits of the integral $\int_{-\infty}^\infty e^{-x^2}\,dx$ is $\pi^{1/2} = 1.772454$. For $x \pm 4$, the integrand is less than $0.5\,10^{-6}$. Using the trapezoidal rule for the integral over $[-4, 4]$ we get the estimate 1.772453 with $h = 1/2$, an amazingly good result. (The values of the function have been taken from a six-place table.) The truncation error in the value of the integral is here less than 1/10,000 of the truncation error in the largest term of the trapezoidal sum—a superb example of "cancellation of truncation error". The error that is which is committed when we

replace ∞ by 4 can be estimated in the following way:

$$|R| = 2\int_4^\infty e^{-x^2}\,dx = 2\int_{16}^\infty e^{-t}0.5t^{-1/2}\,dt$$
$$= 2\cdot 0.516^{-1/2}\int_{16}^\infty e^{-t}0.5\,dt = \frac{1}{4}e^{-16} < 10^{-7}.$$

5.3.5 Adaptive Quadrature

It is often the case that the integrand $f(x)$ (or its derivatives) has strongly varying orders of magnitude in different parts of the interval of integration $[a, b]$. One should then choose *different step sizes in different parts of the integration interval*. Since

$$\int_a^b = \int_a^{c_1} + \int_{c_1}^{c_2} + \cdots + \int_{c_1}^b,$$

the integrals on the right hand side can be treated as independent subproblems. Indeed, it is possible to perform the subdivision recursively in several levels. In **adaptive quadrature methods** step sizes are automatically adapts so that the approximation satisfies a prescribed error tolerance

$$\left| I - \int_a^b f(x)\,dx \right| \le \epsilon. \tag{5.3.14}$$

We first remark that evaluation of the integral (**??**) is equivalent to solving

$$\frac{dy}{dx} = f(x), \qquad y(a) = 0, \tag{5.3.15}$$

and taking $I = y(b)$. This is a special case of an initial value problem for an ordinary differential equation, and the methods described in Chapter 13 can be used to solve the problem (5.3.15). These algorithms have been developed to include sophisticated techniques for adaptively choosing step size and order in the integration (see Sec. 13.2), and may therefore be a good choice for handling difficult cases.

We consider first a fixed order adaptive method based on Simpson's rule. For a subinterval $[a, b]$, set $h_=(b-a)$ and compute the trapezoidal approximations

$$T_{00} = T(h), \quad T_{10} = T(h/2), \quad T_{20} = T(h/4).$$

The extrapolated values

$$T_{11} = (4T_{10} - T_{00})/3, \qquad T_{21} = (4T_{20} - T_{10})/3,$$

are equivalent to (the composite) Simpson's rule with step length $h/2$ and $h/4$, respectively. We can also calculate

$$T_{22} = (16T_{21} - T_{11})/15,$$

which is Milne's method with step length $h/4$ with remainder equal to $(2/945)(h/4)^6(b-a)f^{(6)}(\xi)$.

For T_{22} we use the error estimate $R_j = |T_{22} - T_{21}|$, which often is a crude overestimate.

We *accept* the approximation I_j if

$$|T_{21} - T_{11}| < \frac{h_j \epsilon}{b-a}, \tag{5.3.16}$$

that is we require the error to be *less than* $\epsilon/(b-a)$ *per unit step.* Otherwise we *reject* the approximation, and subdivide the interval in two intervals $[a_j, \frac{1}{2}(a_j + b_j)]$, $[\frac{1}{2}(a_j + b_j), b_j]$. The same rule is now applied to these two subintervals.

Note that if the function values computed previously are we have saved, these can be reused for the new intervals. Only We start with one interval $[a, b]$ and carry on subdivisions until the error criterion in (5.3.16) is satisfied for all intervals. Since the total error is the sum of errors for all subintervals we then have the error estimate

$$R_T < \sum_j \frac{h_j \epsilon}{b-a} = \epsilon$$

as required.

Many adaptive quadrature schemes exits. Here we shall only illustrate one simple scheme based on a five point closed Newton–Cotes rule, which applies bisection in a locally adaptive strategy. All function evaluations contribute to the final estimate.

Algorithm 5.3.1 *Adaptive Simpson.*

Let f be a given function to be integrated over $[a, b]$ The algorithm adaptsimp uses a recursive to compute an approximation with an error less than a specified tolerance $\tau > 0$. The parameter is is a crude a priori estimation of I, used in the stopping criterion.

```
function [I,nf] = adaptsimp(f,a,b,tol);
% ADAPTSIMP computes the integral of the
% function vector valued function f over [a,b];
% tol is the desired absolute accuracy
% nf is the number of function evaluations
%
% Initial Simpson approximation
ff = feval(f,[a, (a+b)/2, b]); nf = 3;
I1 = (b - a)*[1, 4, 1]*ff'/6;
% Recursive computation
[I,nf] =  adaptrec(f,a,b,ff,I1,tol,nf);

function [I,nf] = adaptrec(f,a,b,ff,I1,tol,nf);
h = (b - a)/2;
fm = feval(f, [a + h/2, b - h/2]); nf = nf + 2;
```

```
% Simpson approximations from left and right subinterval
fL = [ff(1); fm(1); ff(2)];
fR = [ff(2); fm(2); ff(3)];
IL = h*[1, 4, 1]*fL/6;
IR = h*[1, 4, 1]*fR/6;;
% Compute Extrapolated approximation
I2 = IL + IR;
I  = I2 + (I2 - I1)/15;
if  abs(I - I2) > tol
% Refine both subintervals
   [IL,nf] = adaptrec(f,a,a+h,fL,IL,tol/2,nf);
   [IR,nf] = adaptrec(f,b-h,b,fR,IR,tol/2,nf);
   I  = IL + IR;!
 end
```

In many situations it might be preferable to specify a *relative error tolerance*

$$tol = \eta \Big| \int_a^b f(x)\,dx \Big|.$$

Note that in a **locally adaptive** algorithm using a recursive partitioning scheme, the subintervals are processed from left to right until the integral over each subinterval satisfies some error requirement. This means that an a priori initial estimate of the whole integral, needed for use in a relative local error estimate cannot be updated until all subintervals are processed and the computation is finished. Hence, if a relative tolerance is specified then a estimate of the integral is needed before the recursion starts. This is complicated by the fact that the initial estimate might be zero, e.g. if a periodic integrand is sampled at equidistant intervals. Hence a combination of relative and absolute criterion might be preferable.

Example 5.3.5.

This algorithm was used to compute the integral

$$\int_{-4}^{4} \frac{dx}{1+x^2} = 2.65163532733607.$$

with an absolute tolerance 10^{-p}, $p = 4, 5, 6$. The following approximations were obtained.

I	tol	n	error
2.65162 50211	10^{-4}	41	$1.0\,10^{-5}$
2.65163 52064	10^{-5}	81	$1.2\,10^{-7}$
2.65163 5327353	10^{-6}	153	$-1.7\,10^{-11}$

Note that the actual error is much smaller than the required tolerance.

The possibility that a user might try to integrate a non-integrable function (e.g., $f(x) = x^{-1}$ on $[0, 1]$) cannot be neglected. In principle it is not possible to decide whether or not a function $f(x)$ is integrable on the basis of a finite sample $f(x_1), \ldots, f(x_N)$ of function values. Therefore it is necessary to impose

1. an upper limit on the computational effort, i.e. the number of function evaluation.

2. a lower limit on the size of the subregions

This means that premature termination may occur even when the function is close to being non-integrable, e.g., $f(x) = x^{-0.99}$.

So far we have considered adaptive routines, which use fixed quadrature rules on each subinterval but where the partition of the interval depends on the integrand. Such an algorithm is said to be **partition adaptive**. We can also consider **doubly adaptive** integration algorithms. These can choose from a sequence of increasingly higher order rules to be applied to the current subinterval. Such algorithms uses a selection criterion to decide to decide at each stage whether to subdivide the current subinterval or to apply a higher order rule. Doubly adaptive routines copes more efficiently with smooth integrands.

Many variations on the simple scheme outlined above are possible. For example, we could base the method on a higher order Romberg scheme, or even try to choose an optimal order for each subinterval. Adaptive methods work even when the integrand $f(x)$ is badly behaved. However, if f has singularities or unbounded derivatives, the error criterion may never be satisfied. For guard against such cases it is necessary to include some bound of the number of recursion levels that are allowed. It should be kept in mind that although adaptive quadrature algorithms are convenient to use they are in general less efficient than methods which have been specially adapted for a particular problem.

A collection of computer subroutines for adaptive quadrature is given by Piessens et al. [28]. We finally warn the reader that *no automatic quadrature routine can be guaranteed always to work*. Indeed any estimate of $\int_a^b f(x)\,dx$ based solely on the value of $f(x)$ on finitely many points can fail. The integrand $f(x)$ may, for example, be nonzero only on a small subset of $[a, b]$. An adaptive quadrature rule based only on samples $f(x)$ in a finite number of points theoretically may return the value zero in such a case!

Review Questions

1. Give an account of the theoretical background of Romberg's method and its use.

2. Romberg's method uses extrapolation of a sequence of trapezoidal approximations computed for a sequence of step sizes $h_0, h_1, h_2, \ldots$. What sequences have been suggested and what are their relative merits?

Problems and Computer Exercises

1. Is it true that (the short version of) Simpson's formula is a particular case of Gregory's formula? (Simpson lived 1710-1761.)

2. Use Romberg's method to compute the integral $\int_0^4 f(x)\,dx$, using the following (correctly rounded) values of $f(x)$. Need all the values be used?

x	0.0	0.5	1.0	1.5	2.0	2.5	3.0	3.5	4.0
$f(x)$	−4271	−2522	−499	1795	4358	7187	10279	13633	17247

3. (a) Suppose that the form of the error of Romberg's method is known, but the error constant r_k is not known. Determine r_k numerically for $K = 3$ and $k = 4$, by computing the Romberg scheme for $f(x) = x^{2k}$.

(b) Prove the formula for the error constant of Romberg's method.

4. Compute by the Euler–Maclaurin formula, or rather the trapezoidal rule,

$$\text{(a)}\ \int_0^\infty e^{-x^2/2}dx, \qquad \text{(b)}\ \int_0^\infty \frac{dx}{\cosh(\pi x)},$$

as accurately as you can with the normal precision of your computer (or software). Then find out empirically how the error depends on h. Make semi-logarithmic plots on the same screen. How long range of integration do you need?

goodbreak

5. (a) Compute $\int_1^\infty (1+x^2)^{-1}\,dx$. In the notation of Example 5.3.5, compute $\int_1^2$, $\int_2^4$, $\int_4^8, \ldots$; choose yourself where to stop. Use, e.g., Aitken acceleration to find $\int_1^\infty$. Compare with the exact result; and think of an error estimate that can be used if the exact result is not known.

(b)*Romberg+Aitken* Treat in the same way $\int_1^\infty \frac{1}{\sqrt{x+x^3}}$. Compare the computational effort for the computation of the tail $\int_R^\infty$ by acceleration and by series expansion with the same accuracy.

6. Compute the integral

$$\frac{1}{2\pi}\int_0^{2\pi} e^{\frac{1}{\sqrt{2}}\sin x}dx$$

by the trapezoidal rule, using $h = \pi/2$ and $h = \pi/4$ (for hand-held calculator). Continue on a computer with smaller values of h, until the error is on the level of the rounding errors. Observe how the number of correct digits vary with h? Notice that Romberg is of no use in this problem.

7. (a) Show that the trapezoidal rule, with $h = 2\pi/(n+1)$, is exact for all trigonometric polynomials of period 2π—i.e., for functions of the type

$$\sum_{k=-n}^{n} c_k e^{ikt}, \qquad i^2 = -1.$$

—when it is used for integration over a whole period.

(b) Show that if $f(x)$ can be approximated by a trigonometric polynomial of degree n so that the magnitude of the error is less than ϵ, in the interval $(0, 2\pi)$, then the error with the use of the trapezoidal rule with $h = 2\pi/(n+1)$

on the integral $(2\pi)^{-1}\int_0^{2\pi} f(x)\,dx$ is less than 2ϵ.

(c) Use the above to explain the sensationally good result in Problem 2 above, when $h = \pi/4$.

Hint: First estimate how well the function $g(x) = e^{x/\sqrt{2}}$ can be approximated by a polynomial in $\mathcal{P}_8$ for $x \in [-1, 1]$. The estimate found by the truncated Maclaurin expansion is not quite good enough. Theorem 3.1.5 provides a sharper estimate with an appropriate choice of R; remember Scylla and Charybdis.

8. (J. N. Lyness) The integral

$$I(f, g) = \int_0^{nh} f(x)g'(x)\,dx \tag{5.3.17}$$

is called a **Stieltjes integral**. An approximation related to the trapezoidal rule is

$$S_m = \tfrac{1}{2}\sum_{j=0}^{n-1} \big(f(jh) + f((j+1)h)\big)\big(g((j+1)h) - (g(jh)\big),$$

which requires $2(m+1)$ function evaluations. Similarly an analogue to the "mid-point rule" is

$$R_m = \tfrac{1}{2}\sum_{j=0}^{n-1}{}'' f(jh)\big(g((j+1)h) - (g((j-1)h)\big),$$

where the double prime on the summation indicates that the extreme values $j = 0$ and $j = m$ are assigned a weighting factor $\frac{1}{2}$. This rule requires $2(m+2)$ function evaluations, two of which lie outside the interval of integration.

(a) Show that the difference $S - m - R_m$ is of order $O(h^2)$.

9. Apply the programs handed out for Romberg's method (also longromb) and repeated averages on the integral

$$\int_0^{1000} x\cos(x^3)\,dx.$$

Try to obtain the results with 10 decimal places.

5.4 Multiple Integrals

5.4.1 Product Rules

The ideas of numerical quadrature can be generalized to multiple integrals. Consider the two-dimensional integral

$$I = \int_D f(x, y)\,dxdy \tag{5.4.1}$$

For regions D, such as a square, cube, cylinder, etc., which are the Cartesian product of lower dimensional regions, integration rules can be developed by multiplying together the lower dimensional rules. For example, if

$$\int_0^1 f(x)\,dx = \sum_{i=1}^{n} w_i f(x_i)$$

is a one dimensional rule, then

$$\int_0^1 \int_0^1 f(x,y)\,dxdy = \sum_{i,j=1}^{n} w_i w_j f(x_i, y_j)$$

is a two-dimensional rule for a square. Such rules are not necessarily the most economical rules.

Example 5.4.1.

Consider a quadrature rule of the form

$$\int_{-h}^{h} \int_{-h}^{h} f(x,y)\,dxdy = 4h^2 \sum_{i,j=1}^{n} w_i f(x_i, y_i).$$

The product Simpson's rule uses 9 function values, with abscissas and weights given by

(x_i, y_i)	(0,0)	$(\pm h, \pm h)$	$(\pm h, 0)$	$(0, \pm h)$
w_i	4/9	1/36	1/9	1/9

A more efficient rule is the product 2-point Legendre rule, using the four points

$$(x_i, y_i) = \Big(\pm \frac{h}{\sqrt{3}}, \pm \frac{h}{\sqrt{3}} \Big) \quad w_i = 1/4.$$

For both rules the error is $O(h^4)$. Some quadrature rules for circles, triangles, hexagons, spheres, cubes, etc., are given in Abramowitz–Stegun [1, § 25].

Since the amount of work will increase rapidly with the number of dimensions. It is therefore advisable to try to reduce the number of dimensions by applying analytic techniques to parts of the task.

Example 5.4.2.

The following triple integral can be reduced to a single integral:

$$\int_0^\infty \int_0^\infty \int_0^\infty e^{-(x+y+z)} \sin(xz) \sin(yz)\,dxdydz$$

$$\int_0^\infty e^{-x}\,dx \int_0^\infty e^{-y} \sin(yx)dy \int_0^\infty e^{-z} \sin(zx) = \int_0^\infty \Big(\frac{x}{1+x^2} \Big)^2 e^{-x}\,dx,$$

because

$$\int_0^\infty e^{-z} \sin(zx)dz = \int_0^\infty e^{-y} \sin(yx)dz = \frac{x}{1+x^2}.$$

The remaining single integral is simply evaluated by the techniques previously studied.

Often a transformation of variable is needed for such a reduction (see Problem 1 at the end of this section), but sometimes that does not help either. Several approaches are then possible:

(a) numerical integration in one direction at a time—see Sec. 5.4.2;

(b) the use of a rectangular grid, mainly if the boundary of the region is composed of straight lines–see Sec. 5.4.3.

(c) the use of an irregular triangular grid—possible for more general boundaries—see Sec. 5.4.4.

(d) Monte Carlo methods, mainly for problems with complicated boundaries and a large number of dimensions—see Sec. 5.4.5.

5.4.2 Successive One-Dimensional Quadrature

For simplicity we restrict ourselves below to the two-dimensional case, although the ideas are more general. Consider the integral (5.4.1) where D is a domain in the x-y plane. The simplest way to compute an approximation to I is by repeated use of one dimensional quadrature rules. If lines parallel with the x-axis have at most one segment in common with D, then I can be written in the form

$$I = \int_a^b \Big(\int_{c(x)}^{d(x)} f(x,y)dy \Big)\, dx,$$

or

$$I = \int_a^b \varphi(x)\, dx, \qquad \varphi(x) = \int_{c(x)}^{d(x)} f(x,y)dy. \tag{5.4.2}$$

For a sequence of values x_i, $i = 1, \ldots, n$ we can evaluate the function $\varphi(x)$ by the one-dimensional quadrature methods described previously. These function values are then used in another one-dimensional quadrature rule to evaluate I. Note that if D is a more general domain, it might be possible to decompose D into the union of simpler domains on which these methods can be used.

Figure 5.4.1. *Region D of integration.*

Example 5.4.3.
Compute

$$I = \int\int_D \sin^2 y \sin^2 x (1 + x^2 + y^2)^{-1/2}\, dxdy,$$

where

$$D = \{(x, y) \mid x^2 + y^2 \le 1\} \cup \{(x, y) \mid 1 \le x \le 3, |y| \le 0.5\}.$$

is the composite region shown in Fig. 8.4.1. Then

$$I = \int_{-1}^{3} \varphi(x) \sin^2 x\, dx, \tag{5.4.3}$$

$$\varphi(x) = \int_{-c(x)}^{c(x)} \sin^2 y (1 + x^2 + y^2)^{-1/2} dy, \tag{5.4.4}$$

where

$$c(x) = \begin{cases} (1 - x^2)^{1/2}, & x \le \frac{1}{2}\sqrt{3}; \\ \frac{1}{2}, & x \ge \frac{1}{2}\sqrt{3}. \end{cases}$$

Values of $\varphi(x)$ were obtained by the application of Romberg's method to (5.4.4) and numerical integration applied to the integral (5.4.3) yielded the value of $I = 0.13202 \pm 10^{-5}$. Ninety-six values of x were needed, and for each value of x, twenty function evaluations used, on the average. The grid is chosen so that $x = \frac{1}{2}\sqrt{3}$, where $\varphi'(x)$ is discontinuous, is a grid point.

5.4.3 Product Rules

Consider a double integral over a rectangular region $D = \{(x, y) \mid a \le x \le b, c \le y \le d\}$. Decomposing the integral as in (5.4.2) and using one-dimensional quadrature rules we can write

$$I \approx \sum_{i=1}^{n} u_i \varphi(x_i), \qquad \varphi(x_i) \approx \sum_{j=1}^{n} v_j f(x_i, y_j),$$

or, combining the rules

$$I \approx \sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij} f(x_i, y_j), \qquad w_{ij} = u_i v_j. \tag{5.4.5}$$

This is called a **product rule** for the double integral I, and it uses mn function values $f_{ij} = f(x_i, y_j)$.

In particular we can use values of f and an equidistant **rectangular grid** in the (x, y)-plane with grid spacings h and k in the x and y directions, respectively. Let $x_0 = a$, $h = (b-a)/n$, $y_0 = c$, $k = (d-c)/m$, and use the notation $x_i = x_0 + ih$,

$y_j = y_0 + jk$. Then the following formulas can be used, generalizing the compound rectangle rule and trapezoidal rule, respectively:

$$I \approx hk \sum_{i=1}^{M} \sum_{j=1}^{N} f_{i-\frac{1}{2},j-\frac{1}{2}}, \tag{5.4.6}$$

$$I \approx hk \sum_{i=1}^{M} \sum_{j=1}^{N} w_{ij} f_{ij} \tag{5.4.7}$$

Here, for the trapezoidal rule $w_{ij} = 1$ for the interior grid points—i.e., when $0 < i < M$ and $0 < j < N$, $w_{ij} = \frac{1}{4}$ for the four corner points, while $w_{ij} = \frac{1}{2}$ for the other boundary points. Both formulas are exact for bilinear functions, and the error can be expanded in even powers of h, k so that repeated Richardson extrapolation can be used.

Formulas of higher accuracy can also be obtained by using Gaussian quadrature rules in the x and y direction. Note that if the one-dimensional formulas are exact for polynomials of degree d_1 and d_2, respectively, then the product rule will be exact for bivariate polynomials $x^p y^q$ where $p \leq d_1$ and $q \leq d_2$.

Higher accuracy formulas can also be derived by **operator** techniques, based on an operator formulation of Taylor's expansion, see equation (4.8.2),

$$u(x_0 + h, y_0 + k) = e^{(hD_x + kD_y)} u(x_0, y_0). \tag{5.4.8}$$

It is possible to use product rules on non-rectangular regions, if these can be mapped into a rectangle. This can be done, e.g., for a triangle. For nonrectangular regions, the rectangular lattice may also be bordered by triangles or "triangles" with one curved side, which may be treated with the techniques outlined in the next section.

5.4.4 Irregular Triangular Grids

A grid of triangles of arbitrary form is a convenient means for approximating a complicated plane region. It is fairly easy to program a computer to refine a coarse triangular grid automatically; see Fig. 8.4.2. It is also easy to adapt the density of points to the behavior of the function.

Triangular grids are thus more flexible than rectangular ones. On the other hand, the administration of a rectangular grid requires less storage and a simpler program. Sometimes the approximation formulas are also a little simpler. Triangular grids have an important application in the **finite element method** (FEM) for problems in continuum mechanics and other applications of partial differential equations; see Chapter 14.

Let $P_i = (x_i, y_i)$, $i = 1, 2, 3$, be the vertices of a triangle T. Then any point $P = (x, y)$ in the plane can be uniquely expressed by the vector equation

$$P = \theta_1 P_1 + \theta_2 P_2 + \theta_3 P_3, \qquad \theta_1 + \theta_1 + \theta_1 = 1. \tag{5.4.9}$$

Figure 5.4.2. *Refinement of a triangular grid.*

In fact, the θ_i, which are called **barycentric coordinates** of P, are determined from the following nonsingular set of equations:

$$\begin{aligned} \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 &= x, \\ \theta_1 y_1 + \theta_2 y_2 + \theta_3 y_3 &= y, \\ \theta_1 + \theta_2 + \theta_3 &= 1, \end{aligned} \tag{5.4.10}$$

The interior of the triangle is characterized by the inequalities $\theta_i > 0$, $i = 1, 2, 3$. In this case P is the center of mass (centroid) of the three masses $\theta_1, \theta_2, \theta_3$ located at the vertices of the triangle (see Fig. 8.4.3). This explains the term "barycentric coordinates". $\theta_1 = 0$ is the equation for the side P_2P_3, and similarly for the other sides.

Figure 5.4.3. *Center of mass of a triangle.*

If f is a *nonhomogeneous linear function* of P, i.e., if $f(P) = a^T P + b$, then the reader can verify that

$$f(P) = \theta_1 f(P_1) + \theta_2 f(P_2) + \theta_3 f(P_3). \tag{5.4.11}$$

this is a form of *linear interpolation on triangular grids.* In order to obtain *quadratic interpolation*, we define

$$\Delta'' = f(P_i) + f(P_j) - 2f\big(\frac{1}{2}(P_i + P_j)\big), \qquad i \neq j. \tag{5.4.12}$$

Theorem 5.4.1.

The interpolation formula

$$f(P) = \theta_1 f_1 + \theta_2 f_2 + \theta_3 f_3 - 2(\theta_2\theta_3\Delta''_{23} + \theta_3\theta_1\Delta''_{31} + \theta_1\theta_2\Delta''_{12})$$

where $f_i = f(P_i)$, is exact for all quadratic functions.

Proof. The right-hand is a quadratic function of P, since it follows from (5.4.10) that the θ_i are (nonhomogeneous) linear functions of x, y. (See also Problem 8.) It remains to show that the right hand side is equal to $f(P)$ for $P = P_i$, and $P = (P_i + P_j)/2$, $i, j = 1, 2, 3$.

For $P = P_i$, $\theta_i = 1$, $\theta_j = 0$, $i \neq j$, hence the right hand side equals f_i. For $P = (P_i + P_j)/2$,

$$\theta_i = \theta_j = \frac{1}{2}, \quad \theta_k = 0, \qquad k \neq i, k \neq j,$$

and hence the right hand side becomes

$$\frac{1}{2}f_i + \frac{1}{2}f_j + -2 \cdot \frac{1}{2}\Big(f_i + f_j - 2u\big(\frac{1}{2}(P_i + P_j)\big)\Big) = f\big(\frac{1}{2}(P_i + P_j)\big).$$

□

The following theorem is equivalent to a rule which has been used in mechanics for the computation of moments of inertia since the nineteenth century:

Theorem 5.4.2.

Let A be the area of a triangle T, with vertices P_1, P_2, P_3. Then the quadrature formula

$$\int\int_T f(x, y)\, dxdy \tag{5.4.13}$$
$$= \frac{1}{3}A\left(f\Big(\frac{1}{2}(P_1 + P_2)\Big) + f\Big(\frac{1}{2}(P_2 + P_3)\Big) + f\Big(\frac{1}{2}(P_3 + P_1)\Big)\right.$$

is exact for all quadratic functions.

PROOF: By symmetry, $\int_T \int \theta_i\, dxdy$ is the same for $i = 1, 2, 3$. Similarly $\int_T \int \theta_i\theta_j\, dxdy$ is the same for all three (i, j)-combinations. Hence for the quadratic function

$$\begin{aligned}\int_T \int f(x, y)\, dxdy &= a(f_1 + f_2 + f_3) - 2b(\Delta''_{23} + \Delta''_{31} + \Delta''_{12}) \\ &= (a - 4b)(f_1 + f_2 + f_3) \\ &+ 4b\Big(f\big(\frac{1}{2}(P_1 + P_2)\big) + f\big(\frac{1}{2}(P_2 + P_3)\big) + f\big(\frac{1}{2}(P_3 + P_1)\big)\Big),\end{aligned}$$

where

$$a = \int_T \int \theta_1\, dxdy, \qquad b = \int_T \int \theta_1\theta_2\, dxdy.$$

Using θ_1, θ_2 as new variables of integration, we get by (5.4.10) and the relation $\theta_3 = 1 - \theta_1 - \theta_2$,

$$\begin{aligned}x &= \theta_1(x_1 - x_3) + \theta_1(x_1 - x_3) + x_3, \\ y &= \theta_1(y_1 - y_3) + \theta_1(y_1 - y_3) + y_3.\end{aligned}$$

Figure 5.4.4. *Correction for curved boundary segment.*

Hence the functional determinant is equal to

$$\begin{vmatrix} x_1 - x_3 & x_2 - x_3 \\ y_1 - y_3 & y_2 - y_3 \end{vmatrix} = 2A,$$

and (check the limits of integration!)

$$a = \int_{\theta_1=0}^{1} \int_{\theta_2=0}^{1-\theta_1} 2\theta_1 d\theta_1 d\theta_2 = 2A \int_0^1 \theta_1(1-\theta_1)d\theta_1 = \frac{A}{3},$$

$$b = \int_{\theta_1=0}^{1} \int_{\theta_2=0}^{1-\theta_1} 2\theta_1\theta_2 d\theta_1 d\theta_2 = 2A \int_0^1 \theta_1 \frac{(1-\theta_1)^2}{2} d\theta_1 = \frac{A}{3}.$$

The results now follows by insertion of this into (5.4.13). □

A numerical method can be based on Theorem 5.4.1, by covering the domain D by triangles. For each curved boundary segment (Fig. 8.4.4) the correction

$$\frac{4}{3} f(S) A(PRQ) \tag{5.4.14}$$

is to be added, where $A(PRQ)$ is the area of the triangle with vertices P, R, Q. The error of the correction can be shown to be $O(\|Q - P\|^5)$ for each segment, if R is close to the midpoint of the arc PQ. If the boundary is given in parametric form, $x = x(x)$, $y = y(x)$, where x and y are twice differentiable on the arc PQ, then one should choose $t_R = \frac{1}{2}(t_P + t_Q)$. Richardson extrapolation can be used to increase the accuracy, see the examples.

Figure 5.4.5. *The grids for I_4 and I_{16}.*

Example 5.4.4.
Consider the integral

$$I = \int\int_D (x^2 + y^2)^k \, dxdy$$

where D is the region shown in Fig. 8.4.5. Let I_n be the result obtained with n triangles. The grids for I_4 and I_{16} are shown in Fig. 8.4.5. Put

$$R'_n = I_{4n} + \frac{1}{15}(I_{4n} - I_n), \qquad R''_n = R'_{4n} + \frac{1}{63}(R'_{4n} - R'_n).$$

The following results were obtained. In this case the work could be reduced by a factor of 4, because of symmetry.

k	I_4	I_{16}	I_{64}	R'_4	R'_{16}	R''_4
2	0.250000	0.307291	0.310872	0.311111	0.311111	0.311111
3	0.104167	0.161784	0.170741	0.165625	0.171338	0.171429
4	0.046875	0.090678	0.104094	0.093598	0.104988	0.105169

The exact values are 0.311111, 0.171429, and 0.105397. It is seen that R'-values have full accuracy for $k = 2$ and the R''-values have high accuracy even for $k = 4$. In fact, it can be shown that R'-values are exact for any fourth-degree polynomial and R''-values are exact for any sixth-degree polynomial, when the region is covered exactly by the triangles.

Example 5.4.5.
The integral

$$a \int\int (a^2 - y^2)^{-1/2} \, dxdy$$

over a quarter of the unit circle is computed with the grids shown in Fig. 8.4.2, and with boundary corrections according to (5.4.9). The following results, using the notation of the previous example, were obtained and compared with the exact values:

a	I_8	I_{32}	R'_8	Exact
2	0.351995	0.352077	0.352082	0.352082
4	0.337492	0.337608	0.337615	0.337616
6	0.335084	0.335200	0.335207	0.335208
8	0.334259	0.334374	0.334382	0.334382

Note, however, that Richardson extrapolation may not always give improvement, e.g., when the rate of convergence of the basic method is *more rapid* than usual.

We mention also that some progress has been made in developing quadrature rules of optimal order for rectangles and triangles. In one dimension this led to Gaussian quadrature rules. In two dimensions the problem is much more difficult. Non-product rules for simple regions like a circle, equilateral triangle, regular hexagon, etc., can be found in Abramowitz and Stegun [1, pp. 891–895]. For a thorough treatment of multiple integrals the reader is referred to the book by Stroud [33].

5.4.5 Monte Carlo Methods

Quasi-Monte Carlo methods for numerical integration; see Niederreiter [27] Low discrepancy sequences

Lattice rules are equal weight rules for integration of periodic functions over the d-dimensional unit cube $[0,1]^d$. Thus the problem is to approximate the integral

$$If = \int_0^1 \cdots \int_0^1 f(x_1, \ldots, x_d)\, dx_1 \ldots dx_d, \tag{5.4.15}$$

by a rule

$$Q_N f = \frac{1}{N} \sum_{j=0}^{N-1} f\left(\left\{ \frac{j}{N} g \right\} \right), \tag{5.4.16}$$

where g is an d-dimensional integer vector that does not have N as a factor and by $\{x\} = \{x_1, \ldots, x_d$ we denote the vector whose jth component is the fractional part of x_j.

For numerical integration in high dimensions the number of function values needed to obtain an acceptable approximation tends to increase exponentially in the number of dimensions d. This is often referred to as *the curse of dimensionality*, a phrase coined by Richard Bellman. The exponential increase is clearly inevitable with any form of product integration rule. Recently it has been shown that the curse can be lifted by using a class of randomly shifted lattice rules by Ian H. Sloane.

One of the most important application of the Monte Carlo method described in Section 1.4.2 is in the numerical calculation of multiple integrals. If we use product rules to evaluate a multiple integral in d dimensions the work will depend exponentially on d. This means that the problem may quickly becomes intractable when d increases. On the other hand, for the Monte Carlo method the complexity always is proportional to $1/\epsilon$, where ϵ is the required tolerance *independent of the dimension* d. Hence the Monte Carlo method can be said to break "the curse of dimension" inherent in other approaches!

We shall briefly describe some ideas used in integration by the Monte Carlo method. For simplicity, we first consider integrals in *one* dimension, even though the Monte Carlo method cannot really compete with traditional numerical methods for this problem.

Let $R_1, R_2, \ldots, R_n$ be a sequence of random numbers rectangularly distributed on $[0,1]$, and set

$$I = \int_0^1 f(x)\, dx \approx I_1 = \frac{1}{n} \sum_{i=1}^{n} f(R_i).$$

This generalizes to multiple integrals. For example, to approximate a two dimensional integral over the domain $0 \le x, y \le 1$, we sample the integrand $f(x, y)$ in points (R_{2i-1}, R_{2i}), for $i = 1, 2, \ldots, n$. The technique can be applied to an integral over a general region D, provided that we can sample the integrand f randomly over D.

One can show that the expectation of the variable I_1 is I and that the standard deviation of this estimate decreases in proportion to $n^{-1/2}$. This is very slow even compared to the trapezoidal rule—where the error decreases as n^{-2}. To get one extra decimal place of accuracy we must increase the number of points by a factor of 100. To get three digit accuracy the order of one million points may be required! However, if we consider, e.g., a six-dimensional integral this is not exorbitant. Using a product rule with 10 subdivisions in each dimension would also require 10^6 points.

The above estimate is a special case of a more general one. Suppose X_i $i = 1, 2, \ldots, n$, has density function $g(x)$. Then

$$I_2 = \frac{1}{n} \sum_{i=1}^{n} \frac{f(X_i)}{g(X_i)}$$

has expected value I, since

$$E\left(\frac{f(X_i)}{g(X_i)}\right) = \int_0^1 \frac{f(x)}{g(x)} f(x)\, dx = \int_0^1 f(x)\, dx = I.$$

If one can find a frequency function $g(x)$ such that $f(x)/g(x)$ fluctuates less than $f(x)$, then I_2 will have smaller variance than I_1. This procedure is called **importance sampling**; it has proved very useful in particle-physics problems, where important phenomena (e.g., dangerous radiation which penetrates a shield) are associated with certain events of low probability.

We have previously mentioned the method of using a simple comparison problem. The Monte Carlo variant of this method is called the **control variate method**. Suppose that $\varphi(x)$ is a function whose integral has a known value K, and suppose that $f(x) - \varphi(x)$ fluctuates much less than $f(x)$. Then

$$I = K + \int_0^1 (f(x) - \varphi(x))\, dx \approx K + I_3, \qquad I_3 = \frac{1}{n} \sum_{i=1}^{n} (f(R_i) - \varphi(R_i)),$$

where I_3 has less variance than I_1.

Review Questions

1. How is bilinear interpolation performed? What is the order of accuracy?
2. Define barycentric coordinates, and give the formula for linear interpolation on a triangular grid.

3. Describe the methods for numerical integration with rectangular or triangular grids.

Problems

1. Let D be the unit circle. Introduce polar coordinates in the integral

$$I = \int\int_D \frac{y\sin(ky)}{x^2+y^2}\,dxdy$$

and reduce it analytically to a single integral.

2. Let E be the ellipse $\{(x,y) \mid (x/a)^2 + (y/b)^2 \le 1\}$. Transform

$$I = \int\int_E f(x,y)\,dxdy$$

into an integral over a rectangle in the (r,t)-plane with the transformation $x = ar\cos t$, $y = br\sin t$.

3. Compute by bilinear interpolation $u(0.5, 0.25)$ when

$$u(0,0) = 1, \quad u(1,0) = 2, \quad u(0,1) = 3, \quad u(1,1) = 5.$$

4. Show that, using the notation for equidistant rectangular grids, the formula

$$\int_{x_0-h}^{x_0+h}\int_{y_0-k}^{y_0+k} f(x,y)\,dxdy = \frac{4hk}{6}(f_{1,0} + f_{0,1} + f_{-1,0} + f_{0,-1} + 2f_{0,0})$$

is exact for all cubic polynomials.

5. Is a quadratic polynomial uniquely determined, given six functions values at the vertices and midpoints of the sides of a triangle?

6. Show that the boundary correction of (5.4.9) is exact if $f \equiv 1$, and if the arc is a parabola where the tangent at R is parallel to PQ.

7. Formulate generalizations to several dimensions of the integral formula of Theorem 5.4.1, and convince yourself of their validity.

Hint: The formula is most simply expressed in terms of the values in the vertices and in the centroid of a simplex.

8. (a) Write a program which uses the Monte Carlo method to compute $\int_0^1 e^x\,dx$. Take 25, 100, 225, 400 and 635 points. Plot the error on a loglog-scale. How does the error depend (approximately) on the number of points?

(b) Compute the integral in (a) using the control variate method. Take $\varphi(x) = 1 + x + x^2/2$. Use the same number of points as in (a).

Notes and References

A comprehensive treatment of the numerical evaluation of integrals is given in Davis and Rabinowitz [6]. Alternatively the Newton–Cotes and other quadrature rules can be derived using computer algebra systems, see [10].

For a history of Gauss-type quadrature rules, see Gautschi [13]. Gaussian quadrature rules were derived by Gauss in 1814 using a continued fraction expansion related to the hypergeometric series. In 1826 Jacobi showed that the nodes were the zeros of the Legendre polynomials and that they were real, simple and in $[-1, 1]$. The convergence of Gaussian quadrature methods was first studied by Stieltjes in 1884. A software package in the public domain by Gautschi [14] includes routines for generating Gauss-type formulas and orthogonal polynomials not only for classical but also for essentially arbitrary weight functions. The presentation in Sec. 5.2.3 is inspired by the work of Gautschi [12], [15], Golub and co-authors. Related ideas can be traced to Mysovskih [26].

The classical reference on orthogonal polynomials is Szegö [35]. Tables of abscissas and weights for Gaussian quadrature rules with various weight functions are given in Abramowitz and Stegun [1, Sec. 25] and in Gautschi [13]. A computer package for computing the tridiagonal Jacobi matrix and generating the corresponding Gauss quadrature rule has been developed by Gautschi [14]. Maple programs for Gauss quadrature rules ar given by von Matt [25].

The idea of adaptive Simpson quadrature is old and treated fully by Lyness [24]. Further schemes, computer programs and examples are given in Davis and Rabinowitz [6]. For a recent discussion of error estimates and reliability of different codes see Espelid [8].

Multivariate integration formulas and lattice rules are discussed in [36].

Bibliography

[1] Milton Abramowitz and Irene A. Stegun (eds.). *Handbook of Mathematical Functions*. Dover, New York, NY, 1965.

[2] F. L. Bauer, H. Rutishauser, and E. Stiefel. New aspects in numerical quadrature. In *Proc. of Symposia in Appl. Math.*, volume 15, pages 199–218. Amer. Math. Soc., 1963.

[3] Claude Brezinski. *Padé-Type Approximations and General Orthogonal Polynomials*. Birkhäuser Verlag, Basel, 1980.

[4] R. Bulirsch. Bemerkungen zur Romberg-Integration. *Numer. Math.*, 6:6–16, 1858.

[5] E. B. Christoffel. Über die Gaussische Quadratur und einige Verallgemenierung derselbern. *J. Reine Angew. Math.*, 55:61–82, 1858.

[6] P. J. Davis and Philip Rabinowitz. *Methods of Numerical Integration*. Academic Press, Orlando, FL, second edition, 1984.

[7] H. Engels. *Numerical Quadrature and Cubature*. Academic Press, London, 1980.

[8] Terje O. Espelid. Doubly adaptive quadrature routines based on Newton–Cotes rules. *BIT*, 43:319–337, 2003.

[9] Walter Gander and Walter Gautschi. Adaptive quadrature—revisited. *BIT*, 40:84–101, 2000.

[10] Walter Gander and Dominik Gruntz. Derivation of numerical methods using computer algebra. *SIAM Review*, 41:3:577–593, 1999.

[11] C. F. Gauss. Methodus nova integralium valores per approximationem inveniendi. *Comm. Soc. R. Sci. Gött. Recens*, 3:39–76, 1814.

[12] Walter Gautschi. On the construction of Gaussian quadrature rules from modified moments. *Math. Comp.*, 24:245–260, 1970.

[13] Walter Gautschi. A survey of Gauss–Christoffel quadrature formulae. In P. L. Butzer and F. Fehér, editors, *E. B. Christoffel: The influence of his work on mathematics and the physical sciences*, pages 72–147. Birkhäuser, Basel, 1981.

[14] Walter Gautschi. Algorithm 726: ORTHPOL—a package for generating orthogonal polynomials and Gauss-type quadrature rules. *Trans. Math. Software*, 20:21–62, 1994.

[15] Walter Gautschi. Orthogonal polynomials: applications and computation. *Acta Numerica*, 5:45–119, 1996.

[16] Walter Gautschi. *Numerical Analysis, an Introduction*. Birkhäuser, Boston, MA, 1997.

[17] Gene H. Golub. Some modified matrix eigenvalue problems. *SIAM Review*, 15:318–334, 1973.

[18] Gene H. Golub and J. H. Welsch. Calculation of Gauss quadrature rules. *Math. Comp.*, 23:221–230, 1969.

[19] Tore Håvie. Generalized Neville type extrapolation schemes. *BIT*, 19:204–213, 1979.

[20] David Kahaner, Cleve B. Moler, and Stephen Nash. *Numerical Methods and Software*. Prentice-Hall, Englewood Cliffs, NJ, 1989.

[21] A. S. Kronrod. *Nodes and Weights of Quadrature Formulas*. Consultants Bureau, New York, 1965. Translation from Russian.

[22] Cornelius Lanczos. *Linear Differential Operators*. D. Van Nostrand, London, UK, 1961.

[23] Bengt Lindberg. A simple interpolation algorithm for improvement of the numerical solution of a differential equation. *SIAM J. Numer. Anal.*, 9:662–668, 1972.

[24] J. N. Lyness. Notes on the adaptive Simpson quadrature routine. *J. Assoc. Comput. Mach.*, 16:483–495, 1969.

[25] Urs von Matt. Gauss quadrature. In W. Gander and J. Hřebiček, editors, *Solving Problems in Scientific Computing using MAPLE and MATLAB*, pages 251–279. Springer-Verlag, Berlin, 1997.

[26] I. P. Mysovskih. On the construction of cubature formulas with the smallest number of nodes. *Soviet Math. Dokl.*, 9:277–280, 1968.

[27] H. Niederreiter. Quasi-Monte Carlo methods for numerical integration. In G. Hämmerling and H. Brass, editors, *Numerical Integration III*, pages 157–171. Birkhäuser Verlag, 1988.

[28] R. Piessens, E. de Doncker, C. W. Überhuber, and D. K. Kahaner. *QUADPACK, A Subroutine Package for Automatic Integration*. Springer-Verlag, Berlin, 1983.

[29] W. Romberg. Vereinfachte numerische Integration. *Kong. Norske Videnskabers Selskab Forhandlinger*, 28:7, 1955.

[30] I. H. Sloan and S. Joe. *Lattice Methods for Multiple Integration.* Clarendon Press, Oxford, UK, 1994.

[31] J. F. Steffensen. *Interpolation.* Chelsea, New York, second edition, 1950.

[32] Joseph Stoer and Roland Bulirsch. *Introduction to Numerical Analysis.* Springer-Verlag, New York, third edition, 2002.

[33] A. H. Stroud. *Approximate Calculation of Multiple Integrals.* Prentice-Hall, Englewood Cliffs, NJ, 1972.

[34] A. H. Stroud and D. Secrest. *Gaussian Quadrature Formulas.* Prentice-Hall, Englewood Cliffs, NJ, 1966.

[35] Gabor Szegö. *Orthogonal Polynomials*, volume 23 of *Colloq. Publ.* Amer. Math. Soc., Providence, RI, fourth edition, 1975.

[36] Christoph W. Ueberhuber. *Numerical Computation: Methods, Software, and Analysis. Volumes 1 & 2.* Springer-Verlag, Berlin, 1997.

For the derivation of error estimates for numerical integration we shall require the following result on the continuity of divided differences. For this purpose the following representation of divided differences is useful. If $x, x_1, \ldots, x_n$ be $n+1$ are distinct points, then

$$[x_1, \ldots, x_n, x]f = \sum_{j=1}^{n} \frac{[x, x_j]f}{\prod\limits_{\substack{k=1 \\ k \neq j}}^{n} (x_j - x_k)}. \tag{5.4.17}$$

This formula follows by substituting the Lagrange form of the interpolation polynomial into the exact remainder (4.2.19) in Newton's interpolation formula.

Lemma 5.4.3.

Let $f(x)$ be continuous on $[a, b]$ and let $f'(x)$ be continuous in arbitrary small intervals about some distinct fixed points $x_i \in [a, b]$, $i = 1 : n$. Then

$$[x_1, \ldots, x_n, x]f$$

is a continuous function of x in $[a, b]$.

Index

algorithm
    adaptive Simpson, 53

barycentric coordinates, 62

companion matrix, 34
coordinates
    barycentric, 62

epsilon algorithm, 49
Euler-Maclaurin's formula, 42, 43

FEM, *see* finite element method
finite element method, 61
Fourier transform, 42

Gauss' quadrature, 22–38
    remainder, 26
Gauss–Christoffel quadrature, 22–38
Gauss–Hermite quadrature, 29
Gauss–Jacobi quadrature, 28
Gauss–Kronrod quadrature, 30
Gauss–Laguerre quadrature, 29
Gauss–Legendre quadrature, 27
Gauss–Lobatto quadrature, 30
Gauss–Radau quadrature, 30
Gram matrix, 32
Gregory's quadrature formula, 42
grid
    irregular triangular, 61–66
    rectangular, 61

Hankel matrix, 32
Hermite interpolation, 26
Hermite polynomials, 29, 39
Hilbert matrix, 32

importance sampling, 67
integral
    over infinite interval, 51
    with singularity, 13–15
integration
    by parts, 13
interpolatory quadrature formula, 2

Jacobi polynomials, 28

Laguerre polynomials, 29
Legendre polynomials, 27
linear interpolation
    on triangular grid, 62
Low discrepancy sequences, 66

midpoint rule
    composite, 5
modified moments, 31
moment, 13
multiple integrals, 57–67

Newton–Cotes'
    9-point formula, 18
    quadrature rule, 8–22, 47
node polynomial, 18, 20, 21
node polynomials, 19
numerical quadrature
    Newton–Cotes, 47

order of accuracy, 1
orthogonal polynomials, 22–24
oscillating integrand, 15

Poisson summation formula, 42, 50

quadratic interpolation
    on triangular grid, 62
quadrature

Monte Carlo methods, 66–67
quadrature rule
adaptive, 52
closed, 5
midpoint, 5
Newton–Cotes', 8–22
open, 5
product, 60–61
Simpson's, 6–8
successive one-dimensional, 59–60
trapezoidal, 3
Quasi-Monte Carlo methods, 66

Richardson extrapolation, 42
Romberg's method, 41–49
error bound, 45

shift matrix, 34
Simpson's formula, 16
singular integrand, 13
Stieltjes integral, 57

transform
Fourier, 42
trapezoidal rule
composite, 4
superconvergence, 50
triangular grid
linear interpolation on, 62
quadratic interpolation on, 62
refinement of, 61
truncation error
global, 4
local, 4

weight function, 12
weighted quadrature rules, 12