

4

Integration

In this chapter we show you how to carry out numerical integrations. We will derive the Simpson and trapezoid rules because they are so easy and give a good idea of how integration algorithms work, but we will just sketch the other algorithms and quote some error estimates.

4.1 PROBLEM: INTEGRATING A SPECTRUM

An experiment has measured $dN(t)/dt$, the number of particles entering a counter, per unit time, as a function of time. Your **problem** is to integrate this spectrum to obtain the number of particles $N(1)$ that entered the counter in the first second:

$$N(1) = \int_0^1 \frac{dN}{dt}(t) dt. \quad (4.1)$$

As an explicit assessment, in §4.8 we integrate an exponential spectrum. Nonetheless, the methods work equally well for integrating a table of numbers.

4.2 MODEL: QUADRATURE, SUMMING BOXES

The integration of a function may require some cleverness to do analytically, but it is relatively straightforward on a computer. The Riemann definition of an integral is the limit of the sum over boxes as the width h of the box

Table 4.1 Elementary weights for uniform-step integration rules

Name	Degree	Elementary Weights
Trapezoid	1	$(h/2, h/2)$
Simpson's	2	$(h/3, 4h/3, h/3)$
$\frac{3}{8}$	3	$(3h/8, 9h/8, 9h/8, 3h/8)$
Milne	4	$(14h/45, 64h/45, 24h/45, 64h/45, 14h/45)$

approaches zero:

$$\int_a^b f(x)dx = \lim_{h \rightarrow 0} \left[h \sum_{i=1}^{(b-a)/h} f(x_i) \right]. \quad (4.2)$$

For our problem, the function f is the spectrum as a function of time, $f(x) = dN(t)/dt$.

A traditional way to measure the area numerically is to take a piece of graph paper and count the number of boxes or *quadrilaterals* lying below the curve of the function. For this reason numerical integration is also called *numerical quadrature*, even when it gets beyond the box-counting stage.

The integral of a function $f(x)$ is approximated numerically by the equivalent of a sum over boxes:

$$\int_a^b f(x)dx \approx \sum_{i=1}^N f(x_i)w_i. \quad (4.3)$$

In this example, f is evaluated at N points in the interval $[a, b]$, and the function $f_i \equiv f(x_i)$ are summed with each term in the sum weighted by w_i . While, in general, the sum in (4.3) will give the exact integral only when $N \rightarrow \infty$, for certain classes of functions it may be exact for finite N .

The different integration algorithms amount to different ways of choosing the points and weights. Generally, the precision increases as N gets larger, with roundoff error eventually limiting the increase. Because the "best" approximation depends on the specific behavior of the function $f(x)$, there is no universally best approximation. In fact, some of the automated integration schemes found in subroutine libraries will switch from one method to another until they find one that works well.

In the simplest integration schemes, the integrand is approximated by a few terms in the Taylor series expansion of f , and the terms are integrated. Unless the integrand has some very unusual behaviors in each interval, successive terms should yield higher and higher precision. In these so-called Newton-Cotes methods, the total interval is divided into equal subintervals as

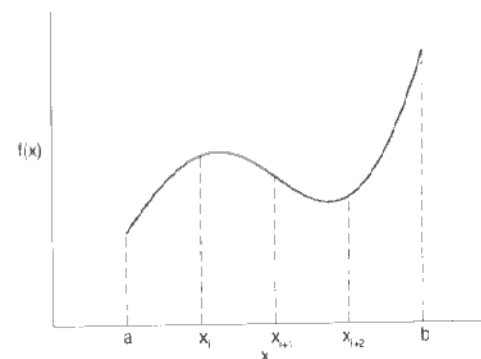


Fig. 4.1 The integral $\int_a^b f(x)dx$ is the area under the graph of $f(x)$ from a to b . Here we break up the area into four regions of equal widths.

shown in Fig. 4.1, with the integrand evaluated at equally spaced points x_i . These algorithms include the trapezoid rule (first-order) and the Simpson rule (second-order). The corresponding weights are given in Table 4.1. These rules are easy to apply and are the logical choice if the integrand is tabulated along evenly spaced points (or if it is tabulated along points that can be mapped onto equally spaced points).

More accurate integration rules are possible if the points are not constrained to be equally spaced. *Gaussian quadrature* methods have the ingenious ability to exactly¹ integrate the product of a function times a $(2N - 1)$ degree polynomial, with only N function evaluations. This is no harder than evaluating the sum (4.3), and explains why so many computational scientists carry their Gaussian quadrature routines with them wherever they go.

In general, results with Gaussian quadrature will be superior to results with equally spaced points as long as there are no singularities in the integrand or its derivative. In the latter cases, the use of Simpson's rule may help avoid a catastrophe, but

you would be better off to remove singularities analytically from your integration before attempting any numerical quadrature.

You may be able to do this by breaking the interval down into several subintervals, so the singularity is at an endpoint where a Gauss point never falls, or by a change of variable. For example

$$\int_{-1}^1 |x|f(x)dx = \int_{-1}^c f(-x)dx + \int_0^1 f(x)dx, \quad (4.4)$$

¹"Exactly," except for roundoff error.

$$\int_0^1 x^{1/3} dx = \int_0^1 3y^2 dy, \quad (y = x^{1/3}), \quad (4.5)$$

$$\int_0^1 \frac{f(x) dx}{\sqrt{1-x^2}} = 2 \int_0^1 \frac{f(1-y^2) dy}{\sqrt{2-y^2}}, \quad (y^2 = 1-x). \quad (4.6)$$

Likewise, if your integrand has a very slow variation in some region, you can speed up the integration by changing to a variable that compresses that region and places few points there. Conversely, if your integrand has a very rapid variation in some region, you may want to change to variables that expand that region to ensure that no oscillations are missed.

4.3 METHOD: TRAPEZOID RULE

The trapezoid and Simpson integration rules use values of $f(x)$ at evenly spaced values of x . They use N points x_i , ($i = 1, N$), evenly spaced at a distance h apart throughout the integration region $[a, b]$ and include the endpoints. This means that there are $N - 1$ intervals of length h :

$$h = \frac{b-a}{N-1}, \quad (4.7)$$

$$x_i = a + (i-1)h, \quad i = 1, N. \quad (4.8)$$

Notice that we start our counting at $i = 1$, and that Simpson's rule requires an odd number of points N .

In Fig. 4.2 we see that in the trapezoid rule we take the integration interval i and construct a trapezoid of width h in it. This approximates $f(x)$ by a straight line in that interval i , and uses the average height $(f_i + f_{i+1})/2$ as the value for f . The area of a single trapezoid is in this way

$$\int_{x_i}^{x_i+h} f(x) dx \approx \frac{h(f_i + f_{i+1})}{2} = \frac{1}{2}hf_i + \frac{1}{2}hf_{i+1}. \quad (4.9)$$

In terms of our standard integration formula (4.3), the "rule" in (4.9) is for $N = 2$ points with weight $w_i \equiv \frac{1}{2}$.

In order to apply the trapezoid rule to the entire region $[a, b]$, we add the contributions from each subinterval:

$$\int_a^b f(x) dx \approx \frac{h}{2}f_1 + hf_2 + hf_3 + \cdots + hf_{N-1} + \frac{h}{2}f_N. \quad (4.10)$$

You will notice that because each internal point gets counted twice, it has a weight of h , whereas the endpoints get counted just once and on that account have weights of only $h/2$. In terms of our standard integration rule (4.39), we

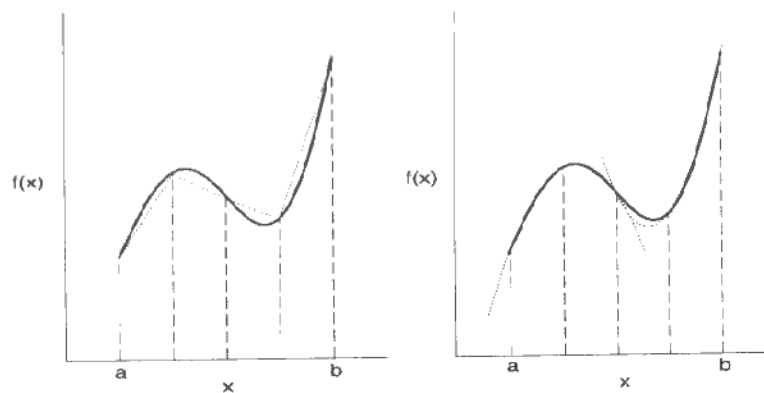


Fig. 4.2 (Left) straight-line sections used for the trapezoid rule; (right) the parabola used in Simpson's rule.

have

$$w_i = \left\{ \frac{h}{2}, h, \dots, h, \frac{h}{2} \right\}. \quad (4.11)$$

4.4 METHOD: SIMPSON'S RULE

In Simpson's rule we approximate the function $f(x)$ by a parabola

$$f(x) \approx \alpha x^2 + \beta x + \gamma. \quad (4.12)$$

for each interval, still keeping the intervals equally spaced. The area of each section is then the integral of this parabola

$$\int_{x_i}^{x_i+h} (\alpha x^2 + \beta x + \gamma) dx = \frac{\alpha x^3}{3} + \frac{\beta x^2}{2} + \gamma x \Big|_{x_i}^{x_i+h}. \quad (4.13)$$

This is equivalent to integrating the Taylor series up to the quadratic term. In order to relate the parameters α , β , and γ to the function, we consider an interval from -1 to $+1$, in which case

$$\int_{-1}^1 (\alpha x^2 + \beta x + \gamma) dx = \frac{2\alpha}{3} + 2\gamma. \quad (4.14)$$

But we notice that

$$\begin{aligned} f(-1) &= \alpha - \beta + \gamma, & f(0) &= \gamma, & f(1) &= \alpha + \beta + \gamma, \\ \Rightarrow \alpha &= \frac{f(1)+f(-1)}{2} - f(0), & \beta &= \frac{f(1)-f(-1)}{2}, & \gamma &= f(0). \end{aligned} \quad (4.15)$$

In this way we can express the integral as the weighted sum over the values of the function at 3 points:

$$\int_{-1}^1 (\alpha x^2 + \beta x + \gamma) dx = \frac{f(-1)}{3} + \frac{4f(0)}{3} + \frac{f(1)}{3}. \quad (4.16)$$

Because 3 values of the function are needed, we generalize this result to our problem by evaluating the integral over two adjacent intervals, in which case we evaluate the function at the two endpoints and the middle:

$$\int_{x_i-h}^{x_i+h} f(x) dx = \int_{x_i}^{x_i+h} f(x) dx + \int_{x_i-h}^{x_i} f(x) dx \approx \frac{h}{3} f_{i-1} + \frac{4h}{3} f_i + \frac{h}{3} f_{i+1}. \quad (4.17)$$

Simpson's rule requires the elementary integration to be over *pairs* of intervals, and this requires that the total number of intervals be even or the number of points N be odd. In order to apply Simpson's rule to the entire interval, we add up the contributions from each pair of subintervals, counting all but the first and last endpoints twice:

$$\int_a^b f(x) dx \approx \frac{h}{3} f_1 + \frac{4h}{3} f_2 + \frac{2h}{3} f_3 + \frac{4h}{3} f_4 + \dots + \frac{4h}{3} f_{N-1} + \frac{h}{3} f_N. \quad (4.18)$$

In terms of our standard integration rule (4.3), we have

$$w_i = \left\{ \frac{h}{3}, \frac{4h}{3}, \frac{2h}{3}, \frac{4h}{3}, \dots, \frac{4h}{3}, \frac{h}{3} \right\}. \quad (4.19)$$

The sum of these weights provides a useful check on your integration:

$$\sum_{i=1}^N w_i = (N-1)h. \quad (4.20)$$

Remember, N must be odd.

4.5 ASSESSMENT: INTEGRATION ERROR, ANALYTIC

In general, you want to choose an integration rule that gives an accurate answer using the least number of integration points. We obtain a feel for the absolute *approximation* or *algorithmic error* E and the relative error ϵ , by expanding $f(x)$ in a Taylor series around the midpoint of the integration interval. We then multiply that error by the number of intervals N to estimate the error for the entire region $[a, b]$. For the trapezoid and Simpson rules this

yields

$$E_t = O\left(\frac{|b-a|^3}{N^2}\right) f^{(3)}, \quad (4.21)$$

$$E_s = O\left(\frac{|b-a|^5}{N^4}\right) f^{(4)}, \quad (4.22)$$

$$\epsilon_{t,s} = \frac{E_{t,s}}{f}. \quad (4.23)$$

We see that the third derivative term in Simpson's rule cancels (much like the central difference method in differentiation). The equations (4.21)–(4.22) are illuminating by showing how increasing the sophistication of an integration rule leads to an error that falls off with a higher inverse power of N , yet that is also proportional to higher derivatives of f . Consequently, for small intervals and $f(x)$ functions with well-behaved high derivatives, Simpson's rule should converge more rapidly than the trapezoid rule.

To be more specific, we assume that after N steps the *relative* roundoff error is random and of the form

$$\epsilon_{ro} \approx \sqrt{N} \epsilon_m, \quad (4.24)$$

where ϵ_m is the machine precision ($\sim 10^{-7}$ for single precision and $\sim 10^{-15}$ for double precision). We want to determine an N that minimizes the total error, that is, the sum of the approximation and roundoff error

$$\epsilon_{tot} = \epsilon_{ro} + \epsilon_{approx}. \quad (4.25)$$

This occurs, approximately, when the two relative errors are of equal magnitude, which we approximate even further by assuming that the two errors are equal

$$\epsilon_{ro} = \epsilon_{approx} = \frac{E_{trap,simp}}{f}. \quad (4.26)$$

To continue the search for optimum N for a general function f , we set the scale of function size by assuming

$$\frac{f^{(n)}}{f} \approx 1, \quad (4.27)$$

and the scale of length by assuming

$$b-a=1 \Rightarrow h = \frac{1}{N}. \quad (4.28)$$

The estimate (4.26), when applied to the **trapezoid rule**, yields

$$\sqrt{N}\epsilon_m \approx \frac{f^{(2)}(b-a)^3}{fN^2} = \frac{1}{N^2}, \quad (4.29)$$

$$\Rightarrow N \approx \frac{1}{(\epsilon_m)^{2/5}}. \quad (4.30)$$

Because the machine precision ϵ_m differs for single- and double-precision calculations, the optimum number of steps N for the **trapezoid rule** has the following two values:

$$N = \frac{1}{h} = \begin{cases} (1/10^{-7})^{2/5} = 631, & \text{for single precision,} \\ (1/10^{-15})^{2/5} = 10^6, & \text{for double precision.} \end{cases} \quad (4.31)$$

The corresponding errors are

$$\epsilon_{ro} \approx \sqrt{N}\epsilon_m = \begin{cases} 3 \times 10^{-6}, & \text{for single precision,} \\ 10^{-12}, & \text{for double precision.} \end{cases} \quad (4.32)$$

The estimate (4.26) when applied to **Simpson's rule** yields

$$\sqrt{N}\epsilon_m = \frac{f^{(4)}(b-a)^5}{fN^4} = \frac{1}{N^4}, \quad (4.33)$$

$$\Rightarrow N = \frac{1}{(\epsilon_m)^{2/9}}. \quad (4.34)$$

For single and double precision, this now yields

$$N = \frac{1}{h} = \begin{cases} (1/10^{-7})^{2/9} = 36, & \text{for single precision,} \\ (1/10^{-15})^{2/9} = 2154, & \text{for double precision.} \end{cases} \quad (4.35)$$

The corresponding errors are

$$\epsilon_{ro} \approx \sqrt{N}\epsilon_m = \begin{cases} 6 \times 10^{-7}, & \text{for single precision,} \\ 5 \times 10^{-14}, & \text{for double precision.} \end{cases} \quad (4.36)$$

These results are illuminating because they show that

1. Simpson's rule is an improvement over the trapezoid rule.
2. It is possible to obtain an error close to machine precision with Simpson's rule (and with other higher-order integration algorithms).
3. Obtaining the best numerical approximation to an integral is not obtained by letting $N \rightarrow \infty$, but with a relatively small $N \leq 1000$.

4.6 METHOD: GAUSSIAN QUADRATURE

It is often useful to rewrite the basic integration formula (4.3) such that we separate a weighting function $W(x)$ from the integrand:

$$\int_{\alpha}^{\beta} f(x)dx \equiv \int_{\alpha}^{\beta} W(x)F(x)dx \approx \sum_{i=1}^N w_i F(x_i). \quad (4.37)$$

In the Gaussian quadrature approach to integration, the N weights w_i are chosen to make the approximation error actually vanish if $f(x)$ was a $2N - 1$ degree polynomial. To obtain this incredible optimization, the points x_i end up having a very specific distribution over $[a, b]$. (If f is only given by an equally spaced table, then a *Simpson* or *trapezoid rule* is simpler, although any integration algorithm can be used if the table is interpolated.²)

In general, if $f(x)$ is smooth, or can be made smooth by factoring out some $W(x)$, Gaussian algorithms produce higher accuracy than lower-order ones, or conversely, the same accuracy with a fewer number of points. If the function being integrated is not smooth (for example, if it has some noise in it), then using a higher-order method such as Gaussian quadrature may well lead to lower accuracy. Sometimes the function may not be smooth because it has different behaviors in different regions. In these cases it makes sense to integrate each region separately with a low-order quadrature rule and then add the answers together. In fact, some of the "smart" integration subroutines will decide for themselves how many intervals to use and what rule to use in each interval.

All the rules indicated in Table 4.2 are Gaussian with the general form (4.37). We can see that in one case the weighting function is an exponential, in another a Gaussian, and in several an integrable singularity. In contrast to the equally-spaced rules, there is never an integration point at the extremes of the intervals, yet all of the points and weights change as the number of points N changes.

The values of α and β in (4.37) help define each integration scheme. If the integral you need to evaluate is over the range (a, b) and this differs from (α, β) , you must *map* your (a, b) onto (α, β) , possibly using one of the methods discussed below.

Although we will leave it to the references on numerical methods for the derivation of the Gauss points and weights, we note here that for ordinary Gaussian (Gauss-Legendre) integration, the points x_i turn out to be the N

²Interpolation schemes will be described soon.

Table 4.2 Types of Gaussian integration rules

Integral	Name
$\int_{-1}^1 f(x) dx$	Gauss
$\int_{-1}^1 \frac{F(x)}{\sqrt{1-x^2}} dx$	Gauss-Chebyshev (rational)
$\int_{-\infty}^{\infty} e^{-x^2} F(x) dx$	Gauss-Hermite
$\int_0^{\infty} e^{-x} F(x) dx$	Gauss-Laguerre
$\int_0^{\infty} \frac{e^{-x}}{\sqrt{x}} F(x) dx$	Associated Gauss-Laguerre

Table 4.3 Points and weights for 4-point Gaussian quadrature (points repeated for negative x)

$\pm x_i$	w_i
0.33998 10435 84856	0.65214 51548 62546
0.86113 63115 94053	0.34785 48451 37454

zeros of the Legendre polynomials, with the weights related to the derivatives:

$$P_N(x_i) = 0, \quad w_i = \frac{2}{(1-x_i^2)[P_N'(x_i)]^2}. \quad (4.38)$$

Subroutines to generate these points and weights are standard in mathematical function libraries, are found in tables such as those in [A&S 64], or can be computed. The *gauss* subroutines we provide on the diskette and the Web also scales the points to a specified region. As a check that your points are correct, you may want to compare them to the four-point set we give in Table 4.3.

4.6.1 Scaling with Integration Rules

Our standard convention (4.3) for the general interval $[a, b]$ is

$$\int_a^b f(x) dx \approx \sum_{i=1}^N f(x_i) w_i. \quad (4.39)$$

But when the points and weights, (y_i, w_i') , are for a fixed integration range (such as Gaussian), the programmer must *scale* the Gaussian interval to $[a, b]$. Here are some mappings we have found useful in our work (they are in the subroutine *gauss* given on the diskette and Web).

In the scalings below, (y_i, w_i') are the elementary Gaussian points and weights for the interval $[-1, 1]$, and we want to scale to x .

$[-1, 1] \rightarrow [A, B]$ uniformly, $A \frac{1}{2} B = \text{midpoint}$:

$$x_i = \frac{B+A}{2} + \frac{B-A}{2} y_i \quad (4.40)$$

$$\Rightarrow \int_A^B f(x) dx = \frac{B-A}{2} \int_{-1}^1 f[x(y)] dy \quad (4.41)$$

$$\Rightarrow w_i = \frac{B-A}{2} w_i' \quad (4.42)$$

$[0 \rightarrow \infty]$, $A = \text{midpoint}$:

$$x_i = A \frac{1+y_i}{1-y_i}, \quad (4.43)$$

$$w_i = \frac{2A}{(1-y_i)^2} w_i' \quad (4.44)$$

$[-\infty \rightarrow \infty]$, scale set by A :

$$x_i = A \frac{y_i}{1-y_i^2}, \quad (4.45)$$

$$w_i = \frac{A(1+y_i^2)}{(1-y_i^2)^2} w_i' \quad (4.46)$$

$[B \rightarrow \infty]$, $A+2B = \text{midpoint}$:

$$x_i = \frac{A+2B+Ay_i}{1-y_i}, \quad (4.47)$$

$$w_i = \frac{2(B+A)}{(1-y_i)^2} w_i' \quad (4.48)$$

$[0 \rightarrow B]$, $AB/(B+A) = \text{midpoint}$:

$$x_i = \frac{AB(1+y_i)}{B+A-(B-A)y_i}, \quad (4.49)$$

$$w_i = \frac{2AB^2}{(B+A-(B-A)y_i)^2} w_i' \quad (4.50)$$

You can see, that if your integration range extends out to infinity, there will be points at large but not infinite x . As you keep increasing the number of grid points N , the largest x_i moves farther and farther out.

4.7 IMPLEMENTATION: INTEGRATION, INTEG.F (C)

Write a program to integrate an arbitrary function numerically using the trapezoid rule, the Simpson rule, and Gaussian quadrature. Use single precision in order to show more quickly the effects of error. (This may not be possible if your quadrature routines, such as the ones we supply on the diskette and the Web, are written in only double precision.)

For our **problem** we assume exponential decay so that there actually is an analytic answer:

$$\frac{dN(t)}{dt} = e^{-t} \quad (4.51)$$

$$\Rightarrow N(1) = \int_0^1 e^{-t} dt = 1 - e^{-1}. \quad (4.52)$$

4.8 ASSESSMENT: EMPIRICAL ERROR ESTIMATE

Compare the relative error

$$\epsilon = \left| \frac{\text{numeric-exact}}{\text{exact}} \right|, \quad (4.53)$$

for the trapezoid rule, Simpson's rule, and Gaussian quadrature.

1. Make a table of the form

N	h	T -Rule	S -Rule	G -Quad	ϵ_T	ϵ_S	ϵ_G
10	0.1111	⋮	⋮	⋮	⋮		

Try N values of 2, 10, 20, 40, 80, 160,

2. Make a plot like Fig. 4.3 of $\log_{10} \epsilon$ versus $\log_{10} N$. Note that the ordinate is effectively the number of decimal places of precision.
3. Use your plot to determine the power-law dependence of the error on the number of points N . (Notice that you may not be able to reach the roundoff error regime for the trapezoid rule because the approximation error is so large.)
4. (Optional) If possible, see how your answers change for double precision.

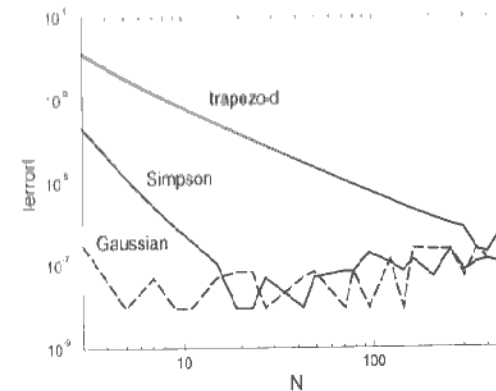


Fig. 4.3 Log-log plot of the error in integration of exponential decay using trapezoid rule, Simpson's rule, and Gaussian quadrature, versus number of integration points N . Approximately 7 decimal places of precision are attainable with single precision (shown here) and 15 places with double precision.

4.9 ASSESSMENT: EXPERIMENTATION

Try two integrals for which the answers are less obvious:

$$F_1 = \int_0^{2\pi} \sin(100x) dx, \quad F_2 = \int_0^{2\pi} \sin^2(100x) dx. \quad (4.54)$$

Explain why the computer may have trouble with these integrals.

4.10 METHOD: ROMBERG EXTRAPOLATION

As in the case of numerical differentiation, we can use the known functional dependence of the error on interval size h to reduce the error in integration. For simple rules like the trapezoid and Simpson's, we have the analytic estimates (4.26), while for others you may have to experiment to determine the h dependence. For example, if $A(h)$ and $A(h/2)$ are the values of the integral determined with the trapezoid rule using interval h and $h/2$, respectively, we know the integrals have expansions with a leading error term proportional to h^2 :

$$A(h) \approx \int_a^b f(x) dx + \alpha h^2 + \beta h^4 + \dots, \quad (4.55)$$

$$A(h/2) \approx \int_a^b f(x) dx + \frac{\alpha h^2}{4} + \frac{\beta h^4}{16} + \dots. \quad (4.56)$$

Consequently, we make the h^2 term vanish by computing the combination

$$\frac{4}{3}A\left(\frac{h}{2}\right) - \frac{1}{3}A(h) \approx \int_a^b f(x)dx - \frac{\beta h^4}{4} + \dots \quad (4.57)$$

Clearly this particular trick (Romberg's extrapolation) works only if the h^2 term dominates the error, and then only if the derivatives of the function are well behaved. An analogous extrapolation can also be made for other algorithms.

4.10.1 Other Closed Newton–Cotes Formulas

In Table 4.1 we gave the weights for several equal-interval rules. Whereas the Simpson rule used two intervals, the $\frac{3}{8}$ rule uses three, and the Milne³ rule four. (These are single-interval rules and must be strung together to obtain a rule *extended* over the entire integration range. This means that the points that end one interval and begin the next get weighted twice.) You can easily determine the number of elementary intervals integrated over, and check whether you and we have written the weights right, by summing the weights for any rule. The sum is the integral of $f(x) = 1$ and must equal h times the number of intervals (which, in turn, equals $b - a$):

$$\sum_{i=1}^N w_i = h \times N_{\text{intervals}} = b - a. \quad (4.58)$$

Part II

APPLICATIONS

³There is, not coincidentally, a Milne Computer Center at Oregon State University.