$$E = \frac{f^{(2n)}(\xi)}{(2n)!} \int_a^b \left(\prod_{i=0}^{n-1} (x - x_i) \right)^2 w(x)\, dx ,$$

where $\xi \in (a,b)$.

§4.5 Romberg Integration

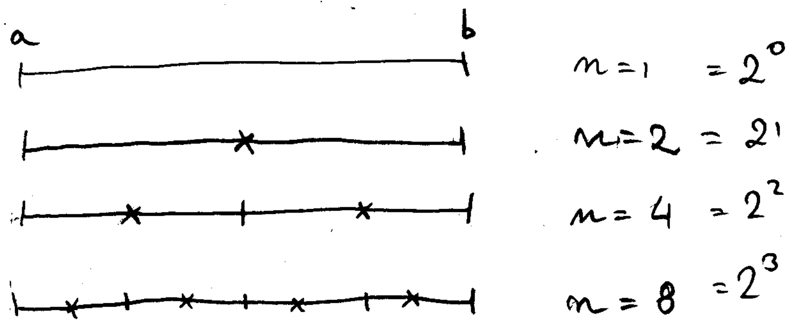
Idea: use Richardson's extrapolation to increase accuracy of composite trapezoidal rule.
→ get high accuracy method from multiple applications of a low accuracy method.

Recall composite trapezoidal rule:

$$\int_a^b f(x)\,dx = \frac{h}{2}\left(f(x_0) + f(x_n) + 2 \sum_{j=1}^{n-1} f(x_j) \right) - \frac{(b-a)}{12} h^2 f''(\xi)$$

where $\xi \in (a,b)$, $h = \frac{b-a}{n}$ and $x_j = a + jh$.

Romberg integration needs to evaluate composite trapezoidal rule for $n = 2^k$, $k = 0, 1, \dots, M$ without superfluous function evaluations



⋮

× = new function evaluation

| = function evaluation available from previous "level".

Thus composite trapz rule becomes with $h_k = 2^{-k}(b-a)$:

$$\int_a^b f(x)\,dx = \frac{h_k}{2}\left[f(a)+f(b)+2\sum_{i=1}^{2^k-1} f(a+ih_k)\right] - \frac{(b-a)}{12}h_k^2 f''(\xi_k)$$

$$= R_{k,0} + \text{error}$$

where $\xi_k \in (a,b)$.

So:

$$R_{00} = \frac{h_0}{2}(f(a)+f(b)) = \frac{b-a}{2}(f(a)+f(b))$$

$$R_{10} = \frac{h_1}{2}(f(a)+f(b)+2f(a+h_1))$$

$$= \frac{1}{2}R_{00} + h_1 f(a+h_1)$$

$$R_{20} = \frac{1}{2}R_{10} + h_2(f(a+h_2)+f(a+3h_2))$$

$$\vdots$$

$$R_{k0} = \frac{1}{2}R_{k-1,0} + h_k \sum_{j=1}^{2^{k-1}} f(a+(2j-1)h_k)$$

evaluation only at odd numbered nodes.

It can be shown that:

① $$\int_a^b f(x)\,dx = R_{k0} + K_1 h_k^2 + K_2 h_k^4 + K_3 h_k^6 + \cdots$$

② $$\int_a^b f(x)\,dx = R_{k+1,0} + K_1 h_{k+1}^2 + K_2 h_{k+1}^4 + K_3 h_{k+1}^6 + \cdots$$

$$= R_{k+1,0} + K_1 \frac{h_k^2}{4} + K_2 \frac{h_k^4}{16} + K_3 \frac{h_k^6}{64} + \cdots$$

Do Richardson's extrapolation trick to cancel out leading term in error:

$4 \times ② - ①$:

$$3\int_a^b f(x)\,dx = 4R_{k+1,0} - R_{k,0} - \frac{3}{4}h_k^4 - \frac{15}{16}h_k^6 - \cdots$$

$$\int_a^b f(x)\,dx = \underbrace{\frac{4R_{k+1,0} - R_{k,0}}{4-1}}_{= R_{k+1,1} \,=\, \mathcal{O}(h_k^4) \text{ approx}} - \frac{1}{4}h_k^4 - \frac{5}{16}h_k^6 - \cdots$$

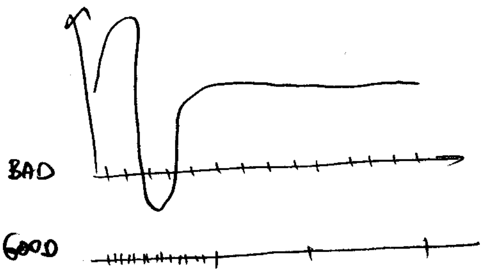
If we keep doing same thing:

$$R_{k+1,j} = \frac{4^j R_{k+1,j-1} - R_{k,j-1}}{4^j - 1} = \mathcal{O}(h_k^{2(j+1)})$$

Organizing work in a table we get: (for example)

$$\begin{array}{lllll}
R_{00} & & & & \\
R_{10} & \hookrightarrow R_{11} & & & \\
R_{20} & \hookrightarrow R_{21} & \hookrightarrow R_{22} & & \\
R_{30} & \hookrightarrow R_{31} & \hookrightarrow R_{32} & \hookrightarrow R_{3,3} & \\
R_{40} & \hookrightarrow R_{41} & \hookrightarrow R_{42} & \hookrightarrow R_{43} & R_{4,4} \\
\mathcal{O}(h_4^2) & \mathcal{O}(h_4^4) & \mathcal{O}(h_4^6) & \mathcal{O}(h_4^8) & \mathcal{O}(h_4^{10})
\end{array}$$

§ 4.6 Adaptive quadrature methods



Using a uniform partition of an interval for a quadrature rule for approximating the integral of a function is not the most efficient way of doing things!

Obviously we can do better if we adapt the # of points to put more points where they are needed the most $\Rightarrow$ principle behind adaptive methods.

Key ingredient: we need a way that tells us how good our quadrature is working. We could do a number of things for example using more nodes to estimate error term or using a more accurate method. Here we choose to refine # of points to estimate error.

On some interval $[a,b]$:

$$(*) \quad \int_a^b f(x)\,dx = \underbrace{\frac{b-a}{6}\left(f(a)+f\left(\tfrac{a+b}{2}\right)+f(b)\right)}_{\equiv\, S(a,b)} - \frac{h^5}{90} f^{(4)}(\xi)$$

for some $\xi \in (a,b)$ where $h = \frac{b-a}{2}$

Of course we could estimate error if we knew $f^{(4)}$ but this is asking too much from end user!

Instead we apply Simpson's rule again: on two subintervals of $[a,b]$:

$$(**) \quad \int_a^b f(x)\,dx = S\left(a,\tfrac{a+b}{2}\right) + S\left(\tfrac{a+b}{2},b\right) - \frac{h^5}{2^5 90} f^{(4)}(\xi_1) - \frac{h^5}{2^5 90} f^{(4)}(\xi_2)$$

$$= \quad '' \qquad '' \quad - \frac{h^5}{90}\left(\frac{1}{16}\right) f^{(4)}(\tilde{\xi})$$

where $\xi_1 \in \left(a,\tfrac{a+b}{2}\right)$, $\xi_2 \in \left(\tfrac{a+b}{2}, b\right)$ and $\tilde{\xi} \in (a,b)$.

Now assume $f^{(4)}$ does not vary too much on (a,b) so:

$$f^{(4)}(\xi) \approx f^{(4)}(\tilde{\xi})$$

So to estimate the error we simply do (**) − (*) to get:

$$\frac{h^5}{90} f^{(4)}(\xi) \approx -\frac{16}{15}\left[S(a,\tfrac{a+b}{2}) + S(\tfrac{a+b}{2},b) - S(a,b)\right]$$

Plugging into (**) and bounding integration error:

$$\left|\int_a^b f(x)\,dx - S(a,\tfrac{a+b}{2}) - S(\tfrac{a+b}{2},b)\right| \le \frac{1}{15}\left|S(a,b) - S(a,\tfrac{a+b}{2}) - S(\tfrac{a+b}{2},b)\right|$$

So if we want (**) to be accurate to within ε we need:

$$\left|S(a,b) - S(a,\tfrac{a+b}{2}) - S(\tfrac{a+b}{2},b)\right| < 15\varepsilon$$

To be conservative (and account for some changes in $f^{(4)}$) we can require 10ε bound.

<u>Algorithm</u>: the book implements a recursive procedure using a for loop. This is like writing a tree traversal algorithm with for loops: very complicated! ⇝ use <u>recursion</u> i.e. a function that calls itself.

⚠ Using recursion is probably less efficient than running code in the book but it's much easier to understand!

⚠ Also some programming languages are limited in # of recursion levels that are allowed e.g. if you run a C program you may get "stack overflow" or other messages of the kind.

⚠ It may be good to impose a limit on # of recursions in "industrial" code as infinite recursions can be nastier than infinite loops!

function $r =$ asimpson(a, b, f, ε)

$$S = \frac{b-a}{6}\left(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b)\right)$$

$$S1 = \frac{b-a}{12}\left(f(a) + 4f\left(a + \frac{b-a}{4}\right) + f\left(\frac{a+b}{2}\right)\right)$$

$$S2 = \frac{b-a}{12}\left(f\left(\frac{a+b}{2}\right) + 4f\left(a + \frac{3(b-a)}{4}\right) + f(b)\right)$$

If $|S - S_1 - S_2| < \varepsilon$

$r = S1 + S2$; % we are happy with approx

else

$r =$ asimpson$(a, \frac{a+b}{2}, f, \varepsilon/2)$ + asimpson$(\frac{a+b}{a}, b, f, \varepsilon/2)$

- recursive calls of adaptive Simpson code for left and right halfs of intervals.

Requiring that error be $\varepsilon/2$ ensures the total error from left + right subint will not exceed ε when summed up.

- instead of accumulating only integrals we can also accumulate e.g. points of discretization.