Richardson Extrapolation (Extrapolation to the Limit)
Romberg Integration and Adaptive Quadrature

I. Richardson Extrapolation

If we have a composite rule with degree of accuracy $p-1$, we can write

$$I(f) = I_n(f) + Ch^p + \mathcal{O}(h^{p+1}),$$

where h is proportional to $1/n$ and C is a constant which depends on f. Taking two different values for n (leading to two different values for h), and combining them in the following way, we have

$$h_2^p I(f) - h_1^p I(f) = h_2^p I_{n_1}(f) - h_1^p I_{n_2}(f) + Ch_2^p h_1^p - Ch_1^p h_2^p + \mathcal{O}(h_1^p h_2^{p+1}, h_2^p h_1^{p+1})$$

which simplifies to

$$I(f) = \frac{h_2^p I_{n_1}(f) - h_1^p I_{n_2}(f)}{h_2^p - h_1^p} + \mathcal{O}(h_i^{p+1}).$$

We have just derived a new approximation of $I(f)$ which is more accurate than the previous approximations. If $h_1 = mh_2$ for some integer m, then we have a new rule:

$$I(f) = \frac{m^p I_{mn}(f) - I_n(f)}{m^p - 1} + \mathcal{O}(h^{p+1}).$$

We can also use this technique to estimate the error by taking a different combination of the formulas with I_{n_1} and I_{n_2} to eliminate $I(f)$ and solve for C. We get the following estimate:

$$C \approx \frac{I_{n_1}(f) - I_{n_2}(f)}{h_2^p - h_1^p}.$$

II. Romberg Integration

When written out carefully, the error for the Composite Trapezoid Rule, looks like:

$$E_n(f) = D_1 h^2 + D_2 h^4 + D_3 h^6 + \sum_{k=4}^{\infty} D_k h^{2k}.$$

Applying Richardson Extrapolation to two Trapezoid approximations 'knocks-out' the first term of the error expansion. Let

$$R_{k,1} = I_{2^k}(f), \quad \text{i.e. Trapezoid rule on } 2^k \text{ intervals.}$$

Let $h_k = (b-a)/2^k$, then one can show

$$\begin{aligned} R_{0,1} &= \frac{b-a}{2}(f(a) + f(b)) \\ R_{k+1,1} &= \frac{1}{2}\left(R_{k,1} + h_k \sum_{i=1}^{2^k} f(a + (2i-1)h_{k+1})\right). \end{aligned}$$

Next, apply Richardson Extrapolation to the estimates $R_{k,1}$ and $R_{k+1,1}$ (note that $h_k = 2h_{k+1}$). So we get a new approximation (call it $R_{k+1,2}$):

$$R_{k+1,2} = \frac{4R_{k+1,1} - R_{k,1}}{3}.$$

Now all these $R_{k,2}$ have degree of accuracy 3 and we can again apply Richardson Extrapolation to $R_{k,2}$ and $R_{k+1,2}$, etc.. So, in general, we get:

$$R_{k+1,j+1} = \frac{4^j R_{k+1,j} - R_{k,j}}{4^j - 1}.$$

Note $R_{k,j}$ has degree of accuracy $2j - 1$. Also, the number of function evaluations to just compute $R_{N,1}$ is the same as the cost of computing $R_{N,N+1}$ using Romberg Integration. Also, the computations can be ordered in such a way that this becomes an adaptive-type scheme. The usual stopping criterion is when $R_{N,N+1} - R_{N-1,N}$ and $R_{N-1,N} - R_{N-2,N-1}$ are both within tolerance (each row is computed as needed).

III. Adaptive Quadrature

This version of adaptive quadrature is based on Simpson's rule and uses the error estimate from Richardson Extrapolation to estimate the error. We write

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

$$S(a, b) = \frac{b - a}{6}(f(a) + 4f(m) + f(b)), \qquad m = (a + b)/2,$$

and

$$S_2(a, b) = S(a, m) + S(m, b).$$

So we have for all a and b, with $h = (b - a)/2$:

$$I(a, b) = S(a, b) - \frac{h^5}{90} f^{(4)}(\zeta),$$

and

$$I(a, b) = S_2(a, b) - \frac{1}{16}\frac{h^5}{90} f^{(4)}(\tilde{\zeta}).$$

So we have the estimate

$$I(a, b) - S_2(a, b) \approx \frac{1}{15}(S(a, b) - S_2(a, b)).$$

Let $E(a, b) = S(a, b) - S_2(a, b)$.

Now, assume we want to estimate $I(a, b)$ to an accuracy of δ. The procedure is as follows:

1. Compute $S(a, b)$ and $S_2(a, b)$ (this can be done with 5 function evaluations)
2. If $|E(a, b)| \leq 15\delta$ then we are done, take the value as either $S_2(a, b)$ or $S_2(a, b) + \frac{1}{15}E(a, b)$.
3. Otherwise, repeat this procedure on each of $I(a, m)$ and $I(m, b)$ to estimate them to an accuracy of $\delta/2$.

The only trick to programming this is to make sure you don't evaluate the function any more than you have to.

An example: suppose we want to approximation $I = \int_0^1 e^x\,dx = 1.7182818$ with a value V so that $|I - V| \le \delta$, where $\delta = 2 \times 10^{-6}$. Start with $V = 0$.
Level 1, Estimate $I(0,1)$

$$\begin{aligned}
S(0,1) &= \frac{1}{6}(e^0 + 4e^{0.5} + e^1) = 1.718861... && (1)\\
S(0,\tfrac{1}{2}) &= \frac{1}{12}(e^0 + 4e^{0.25} + e^{0.5}) = 0.648735... && (2)\\
S(\tfrac{1}{2},1) &= \frac{1}{12}(e^{0.5} + 4e^{0.75} + e^1) = 1.069583... && (3)\\
S_2(0,1) &= 1.718318... && (4)
\end{aligned}$$

Now we estimate the error

$$|E(0,1)| \approx \frac{1}{15}|S_2(0,1) - S(0,1)| = 3.6E - 5,$$

This is not within tolerance (δ), so we split the interval.
Level 2 Left, Estimate $I(0, \frac{1}{2})$

$$\begin{aligned}
S(0,\tfrac{1}{2}) &= 0.648735... && (5)\\
S(0,\tfrac{1}{4}) &= 0.284025... && (6)\\
S(\tfrac{1}{4},\tfrac{1}{2}) &= 0.364696... && (7)\\
S_2(0,\tfrac{1}{2}) &= 0.648722... && (8)
\end{aligned}$$

Now we estimate the error

$$|E(0,\tfrac{1}{2})| \approx 8.73 \times 10^{-7}$$

This is less than $\delta/2$ we take

$$V = V + S_2(0,\tfrac{1}{2}) + \frac{1}{15}(S_2(0,\tfrac{1}{2}) - S(0,\tfrac{1}{2})) = 0.648721...$$

Note: to increase the accuracy we use the Richardson Extrapolation value for the estimate of $I(0, \frac{1}{2})$.
Level 2 Right, Estimate $I(\frac{1}{2}, 1)$

$$\begin{aligned}
S(\frac{1}{2},1) &= 1.069583... && (9)\\
S(\frac{1}{2},\frac{3}{4}) &= 0.468279... && (10)\\
S(\frac{3}{4},1) &= 0.601282... && (11)\\
S_2(\frac{1}{2},1) &= 1.069562... && (12)
\end{aligned}$$

with an error of

$$|E(\tfrac{1}{2},1)| \approx 1.43 \times 10^{-7}$$

This is not less than $\delta/2$ so we split again.

Level 3 Left, Estimate $I(\frac{1}{2}, \frac{3}{4})$

$$\begin{aligned} S(\frac{1}{2}, \frac{3}{4}) &= 0.468279... && (13) \\ S(\frac{1}{2}, \frac{5}{8}) &= 0.219524... && (14) \\ S(\frac{5}{8}, \frac{3}{4}) &= 0.248754... && (15) \\ S_2(\frac{1}{2}, \frac{3}{4}) &= 0.468278... && (16) \end{aligned}$$

with an error of

$$|E(\tfrac{1}{2}, \tfrac{3}{4})| \approx 3.96 \times 10^{-8}$$

which is within the tolerance of $\delta/4$ so we take

$$V = V + S_2(\tfrac{1}{2}, \tfrac{3}{4}) + \frac{1}{15}(S_2(\tfrac{1}{2}, \tfrac{3}{4}) - S(\tfrac{1}{2}, \tfrac{3}{4})) = 1.117000...$$

Level 3 Right, Estimate $I(\frac{3}{4}, 1)$

$$\begin{aligned} S(\frac{3}{4}, 1) &= 0.601282... && (17) \\ S(\frac{3}{4}, \frac{7}{8}) &= 0.281875... && (18) \\ S(\frac{7}{8}, 1) &= 0.319406... && (19) \\ S_2(\frac{3}{4}, 1) &= 0.601281... && (20) \end{aligned}$$

with an error of

$$|E(\tfrac{3}{4}, 1)| \approx 5.08 \times 10^{-8}$$

which is within the tolerance of $\delta/4$, so we take

$$V = V + S_2(\tfrac{3}{4}, 1) + \frac{1}{15}(S_2(\tfrac{3}{4}, 1) - S(\tfrac{3}{4}, 1)) = 1.718281...$$

For the actual error, we have $|I - V| = 5.3 \times 10^{-9}$. It is common for smooth integrands to get a much smaller error than needed.

This is the way a computer program would do this, except that it would be more careful to save and reuse the function evaluations. In pseudo-code, we would have a subfunction like this

```
V = adsimp(a,b,fa,fm,fb,V0,delta)

h  = b-a
f1 = f(a + h/4)                 % left-half midpoint
f2 = f(b - h/4)                 % right-half midpoint
sl = h*(fa + 4*f1 + fm)/12      % left-half estimate
sr = h*(fm + 4*f2 + fb)/12      % right-half estimate
s2 = sl+sr
err = (s2-V0)/15
if (abs(err)<delta)             % estimate is within tolerance, so accept it
   V = s2 + err
else                            % split interval into two pieces
   m = a + h/2
   V1 = adsimp(a,m,fa,f1,fm,sl,delta/2)
   V2 = adsimp(m,b,fm,f2,fb,sr,delta/2)
   V = V1 + V2
endif
```

To call this you would use

```
% main program

fa = f(a)
fm = f((a+b)/2)
fb = f(b)
V0 = (fa + 4*fm + fb)*(b-a)/6

V = adsimp(a,b,fa,fm,fb,V0,delta)
```

Errors for $I = \int_0^1 \sqrt{x}\, dx$.

Function Evals = Method	1	2	3	4	5
Closed N-C	-	-1.7e-1	-2.9e-2	-1.9e-2	-8.9e-3
Open N-C	4.0e-2	3.0e-2	8.3e-3	6.9e-3	
Comp. Trap	-	-1.7e-1	-6.3e-2	-3.5e-2	-2.3e-2
Comp. Simp	-	-	-2.9e-2	-	-1.0e-2
Romberg	-	-	-2.9e-2	-	-8.9e-3
Gauss-Leg	4.0e-2	7.2e-3	2.5e-3	1.2e-3	6e-4

Errors for $I = \int_0^1 \sin x\, dx$.

Function Evals = Method	1	2	3	4	5
Closed N-C	-	-3.9e-2	1.6e-4	7.3e-5	-2.5e-7
Open N-C	2.0e-2	1.3e-2	-1.4e-4	-1.0e-4	
Comp. Trap	-	-3.9e-2	-9.6e-3	-4.3e-3	-2.4e-3
Comp. Simp	-	-	1.6e-4	-	1.0e-5
Romberg	-	-	1.6e-4	-	-2.5e-7
Gauss-Leg	2.0e-2	-1.1e-4	2.4e-7	-2.7e-10	1.8e-13

Errors for $I = \int_0^{10} \sin x\, dx$.

Function Evals = Method	1	2	3	4	5
Closed N-C	-	-4.6	-9.1	-1.8	1.9
Open N-C	-11.4	-0.92	11.6	6.4	
Comp. Trap	-	-4.6	-8.0	-2.1	-1.1
Comp. Simp	-	-	-9.1	-	1.2
Romberg	-	-	-9.1	-	1.9
Gauss-Leg	-11.4	7.4	-2.1	0.29	-0.022