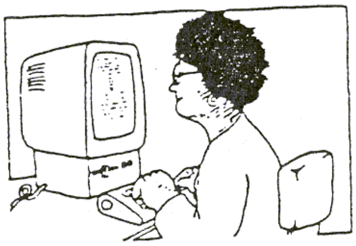
COMPUTER CORNER



EDITOR

Eugene A. Herman
Department of Mathematics
Grinnell College
Grinnell, IA 50112

In this column, readers are encouraged to share their expertise and experiences with computers as they relate to college-level mathematics. Articles that illustrate how computers can be used to enhance pedagogy, solve problems, and model real-life situations are especially welcome.

Classroom Computer Capsules features new examples of using the computer to enhance teaching. These short articles demonstrate the use of readily available computing resources to present or elucidate familiar topics in ways that can have an immediate and beneficial effect in the classroom.

Send submissions for both columns to Eugene A. Herman.

Some Examples Illustrating Richardson's Improvement

Steven Schonefeld

Steven Schonefeld received his Ph.D. in functional analysis from Purdue University in 1969. He is currently Associate Professor of Mathematics at Tri-State University, where he has been on the faculty since 1978. When not at his computer keyboard, he may be found shooting a game of pocket billiards, enjoying a stage play, or snapping 35 mm photos at family gatherings.

In 250 B.C., the Greek mathematician Archimedes approximated the number π by calculating the perimeters of several regular polygons inscribing and circumscribing a circle of unit diameter. In a similar fashion, many of the methods of numerical analysis involve generating sequences of numbers that converge to the solution to a given problem. Like Archimedes, a modern computer can calculate only a finite number of terms for each sequence. If too many terms are used, however, roundoff error may prevent the computer from achieving the desired accuracy. Lewis Fry Richardson [5], [6], [7], and [8] is credited with developing an important numerical technique for accelerating the convergence of certain sequences to the limit, thus giving the desired accuracy from fewer terms. This method is called the deferred approach to the limit, Richardson extrapolation, or Richardson's improvement.

The followin
improvemen

Introductor
the concept
tion of the c

and the cen

In class we
calculator, l
differentiab
nontrivial f
is $f(x) = \ln$
written in
decimal dig
Values of

Example 1.

k
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28

UTER
RNER

omputers as
ed to enhance

nce teaching.
present or
e classroom.

dson's

nal analysis
ociate Profes-
e has been on
keyboard, he
s, enjoying a
gatherings.

number π by
ircumscribing
of numerical
e solution to a
only a finite
ever, roundoff
cy. Lewis Fry
ant numerical
the limit, thus
l the deferred
improvement.

The following examples help to illustrate some of the pros and cons of Richardson's improvement.

Introductory example. Students in my numerical analysis class get introduced to the concept of Richardson's improvement via examples of numerical approximation of the derivative $f'(x)$ by the forward difference quotient

$$\Delta(f,x,h) \equiv \frac{f(x+h)-f(x)}{h}$$

and the central difference quotient

$$\delta(f,x,h) \equiv \frac{f(x+h)-f(x-h)}{2h}.$$

In class we start by calculating a few of these approximations with a hand-held calculator, but eventually I pass around the output from a computer program. Any differentiable function may be used for these calculations, but I like to use a nontrivial function whose derivative at x has a repeating decimal. A good example is $f(x) = \ln(x)$ with $x = 3$. The following table was generated by a simple program written in Turbo Pascal, where calculations are performed to approximately ten decimal digits of accuracy—about the same as given by my hand-held calculator. Values of $\Delta(f,x,h)$ and $\delta(f,x,h)$ are calculated for $h = 2^{-k}$, $k = 1, 2, \ldots$.

Example 1. Approximation of the derivative of $f(x) = \ln(x)$ at $x = 3$.

k	h	$\Delta(f,x,h)$	$\delta(f,x,h)$
1	0.50000000000	0.30830135966	0.33647223662
2	0.25000000000	0.32017083070	0.33410816933
3	0.12500000000	0.32657595616	0.33352643575
4	0.06250000000	0.32990859525	0.33338157120
5	0.03125000000	0.33160918514	0.33334539045
6	0.01562500000	0.33246828022	0.33333634748
7	0.00781250000	0.33290005778	0.33333408693
8	0.00390625000	0.33311650762	0.33333352162
9	0.00195312500	0.33322487306	0.33333338052
10	0.00097656250	0.33327909186	0.33333334513
11	0.00048828125	0.33330620825	0.33333333582
12	0.00024414063	0.33331977576	0.33333333582
13	0.00012207031	0.33332654834	0.33333333582
14	0.00006103516	0.33332994580	0.33333332837
15	0.00003051758	0.33333164454	0.33333334327
16	0.00001525879	0.33333253860	0.33333337307
17	0.00000762939	0.33333277702	0.33333325386
18	0.00000381470	0.33333301544	0.33333325386
19	0.00000190735	0.33333301544	0.33333349228
20	0.00000095367	0.33333396912	0.33333396912
21	0.00000047684	0.33333587646	0.33333396912
22	0.00000023842	0.33333587646	0.33333206177
23	0.00000011921	0.33332824707	0.33333587646
24	0.00000005960	0.33334350586	0.33332824707
25	0.00000002980	0.33331298828	0.33331298828
26	0.00000001490	0.33337402344	0.33337402344
27	0.00000000745	0.33325195313	0.33325195313
28	0.00000000373	0.33349609375	0.33325195313

From Example 1, it is easy for the students to see the approximations getting closer to $f'(x) = 0.33\bar{3}$ for several successive iterations. This example also illustrates the typical way that roundoff error may creep into these calculations. As h gets small, there is a loss of accuracy (roundoff error) in the calculations of $\Delta(f, x, h)$ and $\delta(f, x, h)$ due to the subtraction of nearly equal quantities. In fact, had the iterations continued long enough, the values of h would become so small that $x + h$ and $x - h$ would both round to x, giving computed values of zero for $\delta(f, x, h)$ and $\Delta(f, x, h)$.

With some encouragement, an observant student may notice that $\delta(f, x, h)$ does a better job of approximating $f'(x)$ than $\Delta(f, x, h)$. The best that $\Delta(f, x, h)$ can do in this example is to give six-digit accuracy in iterations 16 through 19, whereas $\delta(f, x, h)$ gives seven digit accuracy in iterations 10 through 15 and is never less accurate than $\Delta(f, x, h)$.

To see why $\delta(f, x, h)$ and $\Delta(f, x, h)$ behave this way, we look at a general function $f(x)$ that is analytic at x. A standard argument [1] shows:

$$\Delta(f, x, h) = f'(x) + \frac{f''(x)}{2!}h^1 + \frac{f^{(3)}(x)}{3!}h^2 + \frac{f^{(4)}(x)}{4!}h^3 + \cdots \qquad (1)$$

and

$$\delta(f, x, h) = f'(x) + \frac{f^{(3)}(x)}{2 \cdot 3!}h^2 + \frac{f^{(5)}(x)}{2 \cdot 5!}h^4 + \frac{f^{(7)}(x)}{2 \cdot 7!}h^6 + \cdots \qquad (2)$$

where $f^{(3)}(x), f^{(4)}(x), \ldots$ are the 3rd, 4th, ... derivatives at x. So when $f''(x) \neq 0$, $f^{(3)}(x) \neq 0$, and h is close to zero, $\Delta(f, x, h) - f'(x)$ will behave like a constant times h and $\delta(f, x, h) - f'(x)$ will behave like a constant times h^2. Thus $\delta(f, x, h)$ can be expected to give a more accurate approximation to the derivative than $\Delta(f, x, h)$.

The derivation of Richardson's improvement. Suppose we wish to approximate a limit $L = \lim_{h \to 0} T(h)$ by calculating $T(h)$ for several different values of h that are close to zero. That is, suppose there exists an algorithm or formula that will permit us to calculate $T(h)$ for a sequence $\{h_k\}$ with $h_1 > 0$ and $h_{k+1} = h_k/2$; (students are reminded of Example 1, in which $h_k = 2^{-k}$). The calculation of each $T(h)$ may be quite expensive, and we expect significant roundoff error as h approaches zero. Even though we might not know the Maclaurin series for $T(h)$, we assume one exists of the form:

$$T(h) = L + a_1 h^{n_1} + a_2 h^{n_2} + a_3 h^{n_3} + \text{higher powers of } h \qquad (3)$$

with the natural numbers $n_1 < n_2 < n_3 < \cdots$ known. For example, when $T(h) = \Delta(f, x, h)$ we have $n_k = k$, and when $T(h) = \delta(f, x, h)$ we have $n_k = 2k$ for $k = 1, 2, \ldots$. Suppose we have calculated $T(2h)$ and $T(h)$. Adding $-T(2h)$ and $2^{n_1}T(h)$, we see that the h^{n_1} terms drop out.

$$\begin{array}{ll} -T(2h) & = -L - 2^{n_1}a_1h^{n_1} - 2^{n_2}a_2h^{n_2} - \cdots \\ +2^{n_1}T(h) & = 2^{n_1}L + 2^{n_1}a_1h^{n_1} + 2^{n_1}a_2h^{n_2} + \cdots \\ \hline 2^{n_1}T(h) - T(2h) & = (2^{n_1} - 1)L \quad + \quad (2^{n_1} - 2^{n_2})a_2h^{n_2} + \cdots \end{array}$$

Dividing this last equation by $(2^{n_1} - 1)$, we get the first Richardson improvement,

ting closer
strates the
gets small,
, x, h) and
, had the
small that
f zero for

, x, h) does
, h) can do
9, whereas
never less

a general

(1)

(2)

n $f''(x) \neq 0$,
a constant
us $\delta(f, x, h)$
ivative than

to approxi-
values of h
formula that
$_{k+1} = h_k/2$;
ation of each
error as h
ies for $T(h)$,

(3)

vhen $T(h) =$
$= 2k$ for $k =$
$- T(2h)$ and

mprovement,

T_1, defined by:

$$T_1(h) \equiv \frac{2^{n_1}T(h) - T(2h)}{2^{n_1} - 1} = L + \frac{2^{n_1} - 2^{n_2}}{2^{n_1} - 1} a_2 h^{n_2} + \cdots .$$

With the h^{n_1} term eliminated, we expect this improved value to give a better approximation to the limit L. This same process may be applied to $T_1(h)$ to give a Richardson improvement $T_2(h)$ with the h^{n_2} term eliminated, then applied to $T_2(h)$ to give a Richardson improvement $T_3(h)$ with the h^{n_3} term eliminated, etc. That is, if we have calculated values for $T_k(2h)$ and $T_k(h)$, we may define the $(k+1)$th Richardson improvement by:

$$T_{k+1}(h) \equiv \frac{2^{n_{k+1}}T_k(h) - T_k(2h)}{2^{n_{k+1}} - 1}, \qquad k = 0, 1, 2, \ldots,$$

where we are using the convention $T_0(h) = T(h)$. The power series for $T_k(h)$ will be of the form

$$T_k(h) = L + b_{k+1}h^{n_{k+1}} + b_{k+2}h^{n_{k+2}} + b_{k+3}h^{n_{k+3}} + \text{higher powers of } h.$$

With the lower powers of h eliminated, we expect $T_k(h)$ to give a better approximation to the limit L than $T_0(h), T_1(h), \ldots,$ or $T_{k-1}(h)$.

It is traditional to display the Richardson improvements in a triangular table as follows:

h	$T_0(h) = T(h)$	$T_1(h)$	$T_2(h)$	$T_3(h)$	$T_4(h)$
1/2	$T(1/2)$				
1/4	$T(1/4) \rightarrow$	$T_1(1/4)$			
1/8	$T(1/8) \rightarrow$	$T_1(1/8) \rightarrow$	$T_2(1/8)$		
1/16	$T(1/16) \rightarrow$	$T_1(1/16) \rightarrow$	$T_2(1/16) \rightarrow$	$T_3(1/16)$	
1/32	$T(1/32) \rightarrow$	$T_1(1/32) \rightarrow$	$T_2(1/32) \rightarrow$	$T_3(1/32) \rightarrow$	$T_4(1/32)$

where the arrows pointing to an improvement indicate the numbers that were used to calculate the improvement. For example, $T_2(1/16)$ has arrows pointing to it from $T_1(1/8)$ and $T_1(1/16)$, indicating that these two numbers were used to calculate $T_2(1/16)$.

How to determine n_k

It may happen that we have an algorithm for calculating $T(h)$ and are reasonably sure that we know the exponents n_k in the Maclaurin series (3) for $T(h)$, but we would like another check to reassure us that things are going as planned. The ratios $R_k(h)$ defined below provide such a check.

Suppose we have calculated values for $T_{k-1}(h)$, $T_{k-1}(2h)$, and $T_{k-1}(h/2)$ as above. Again using the notation $T_0(h) = T(h)$, we define the ratios $R_k(h)$ by:

$$R_k(h) \equiv \frac{T_{k-1}(2h) - T_{k-1}(h)}{T_{k-1}(h) - T_{k-1}(h/2)} \qquad k = 1, 2, \ldots. \tag{4}$$

Next, we replace the values of T_{k-1} in (4) by the following approximations:

$$T_{k-1}(h) \approx L + a_k h^{n_k},$$
$$T_{k-1}(2h) \approx L + 2^{n_k} a_k h^{n_k}, \text{ and}$$
$$T_{k-1}(h/2) \approx L + (1/2)^{n_k} a_k h^{n_k},$$

where we assume a_k is nonzero. Upon simplifying, we conclude that

$$R_k(h) \approx 2^{n_k}.$$

So for $T(h) = \Delta(f,x,h)$ we expect $R_k(h) \approx 2^k$, and for $T(h) = \delta(f,x,h)$ we expect $R_k(h) \approx 2^{2k} = 4^k$.

There are more general ways of treating Richardson's improvement [2], [4]. However, this is about as general as most numerical analysis students would like to get. In fact, they may need to see several examples before the significance of this method sinks in. I have students calculate Richardson improvements and ratios using a hand-held calculator for values of $\delta(f,x,h)$ or $\Delta(f,x,h)$ selected from Example 1; later I show them the output from a computer program that calculates the first three Richardson improvements on $\Delta(f,x,h)$ and the corresponding ratios.

Examples involving $\Delta(f,x,h)$. The program that generated the numbers in the following examples was written in Turbo Pascal. Pseudocode for the calculation of Richardson improvements for an arbitrary $T(h)$ is included at the end of this article. In the present examples, $\Delta(f,x,h)$ was calculated for $h = 2^{-k}$, $k = 1,2,3,\ldots$, but the values of h were not output due to space considerations. The calculation of $\Delta(f,x,h)$ was terminated when 20 iterations were completed or when the last two calculated Richardson improvements in a row of the table agreed to eight digits.

Example 2. $f(x) = \ln(x)$, with $x = 3$.

$T_0(h) = \Delta(f,x,h)$	$R_1(h)$	$T_1(h)$	$R_2(h)$	$T_2(h)$	$R_3(h)$	$T_3(h)$
0.3083013597						
0.3201708307	1.85	0.3320403017				
0.3265759562	1.92	0.3329810816	3.62	0.3332946749		
0.3299085952	1.96	0.3332412343	3.80	0.3333279519	7.13	0.3333327057
0.3316091851	1.98	0.3333097750	3.89	0.3333326219	7.53	0.3333332891
0.3324682801	1.99	0.3333273751	3.95	0.3333332417	7.76	0.3333333303
0.3329000575	1.99	0.3333318350	3.97	0.3333333216	7.76	0.3333333330
0.3331165076		0.3333329577		0.3333333319		0.3333333334

Example 2 is typical of what one expects for Richardson improvements on $\Delta(f,x,h)$:

(a) As one traces down the column of a particular Richardson improvement, the numbers (usually) get closer to $f'(x)$.

(b) As one traces across a row of the table, the values of $T_k(h)$ (usually) get significantly closer to $f'(x)$.

(c) The ratios $R_k(h)$ are approximately 2^k.

(d) The last calculated value of $T_3(h)$ is a better approximation to $f'(x)$ than any of the values calculated from $\Delta(f,x,h)$ in Example 1.

Example 3.

$$f(x) = \begin{cases} \dfrac{5x}{9} + x^2 \sin(1/x) & \text{for } x \neq 0 \\ 0 & \text{for } x = 0 \end{cases}, \quad \text{with } x = 0.$$

$T_0(h) = \Delta(f, x, h)$	$R_1(h)$	$T_1(h)$	$R_2(h)$	$T_2(h)$	$R_3(h)$	$T_3(h)$
1.0102042690						
0.3663549317	−2.06	−0.2774944055				
0.6792253364	−2.21	0.9920957410	−2.13	1.4152924566		
0.5375615983	−4.02	0.3958978601	−2.81	0.1971652332	−2.53	0.0231470584
0.5727876393	−12.33	0.6080136804	−5.18	0.6787189538	−3.84	0.7475123425
0.5699309624	0.33	0.5670742855	2.80	0.5534278205	21.39	0.5355290871
0.5611886627	0.92	0.5524463629	1.42	0.5475703888	0.66	0.5467336128
0.5516523992	−2.35	0.5421161357	−0.59	0.5386727266	−0.33	0.5374016320
0.5557108651	−13.09	0.5597693311	−3.77	0.5656537295	−2.23	0.5695081585
0.5554007378	−158.40	0.5550906105	−14.90	0.5535310370		0.5517992237
0.5554026957		0.5554046536				

In Example 3 we see the classic example of a function having a first derivative at $x = 0$ but no higher derivatives at $x = 0$. We should not expect the Richardson improvements to behave well for this function. The following things happen:

(a) The forward difference approximations continue to get closer to $f'(x) = 5/9$ for the entire 20 iterations.

(b) Even though the Richardson improvements do not get any closer to $f'(x)$ than T_0, they are not worse.

(c) The ratios R_k continue their erratic behavior for the entire 20 iterations, reinforcing our suspicion that no improvement is possible.

Examples involving $\delta(f, x, h)$. The program that generated the values for the following example is nearly identical to the program used for Examples 2 and 3.

Example 4. $f(x) = \ln(x)$, with $x = 3$.

$T_0(h) = \delta(f, x, h)$	$R_1(h)$	$T_1(h)$	$R_2(h)$	$T_2(h)$	$R_3(h)$	$T_3(h)$
0.3364722366						
0.3341081693	4.06	0.3333201469				
0.3335264358	4.02	0.3333325246	16.32	0.3333333497		
0.3333815712	4.00	0.3333332830	16.08	0.3333333336	64.02	0.3333333333
0.3333453904		0.3333333302		0.3333333333		0.3333333333

Example 4 is a companion to Example 2. It illustrates the typical behavior of Richardson improvements on $\delta(f, x, h)$:

(a) The ratios $R_k(h)$ are approximately 4^k.

(b) Fewer calculations are required to get a good approximation to $f'(x)$ than with improvements on $\Delta(f, x, h)$.

(c) The Richardson improvements on $\delta(f, x, h)$ are significantly better approximations to $f'(x)$ than the corresponding improvements on $\Delta(f, x, h)$.

The reader is encouraged to conduct similar experiments in calculating Richardson's improvements on $\Delta(f, x, h)$ and $\delta(f, x, h)$. For example, one may try performing the calculations using single precision (usually 7–8 decimal digits accuracy) and double precision (usually 15–16 decimal digits accuracy) and compare the results. Here are some other functions to use in the caloulation of Richardson's improvements on $\Delta(f, x, h)$ or $\delta(f, x, h)$.

$$f(x) = -1/x, \qquad \text{with } x = 3,\ f'(x) = 0.111111\ldots$$

$$f(x) = \frac{8\sqrt{x}}{9}, \qquad \text{with } x = 4,\ f'(x) = 0.222222\ldots$$

$$f(x) = \frac{4x}{9} + \frac{x^2}{7} - \frac{x^4}{4}, \qquad \text{with } x = 0,\ f'(x) = 0.444444\ldots$$

$$f(x) = \frac{7x}{9} + x|x|, \qquad \text{with } x = 0,\ f'(x) = 0.777777\ldots$$

$$f(x) = \frac{8x}{9} + 10|x|\sin(100x), \qquad \text{with } x = 0,\ f'(x) = 0.888888\ldots$$

Some things to look for are the following:

(a) For polynomials having degree 4 or less, the third Richardson improvement on $\Delta(f, x, h)$ equals $f'(x)$ and the first Richardson improvement on $\delta(f, x, h)$ equals $f'(x)$. (Why?)

(b) Looking at the ratios $R_k(h)$ can help us recognize when the Richardson improvement does not improve the accuracy of our approximations due to round-off error or other factors.

(c) In some of these examples, $\Delta(f, x, h)$ and its Richardson improvements approximate $f'(x)$ better than $\delta(f, x, h)$ and its Richardson improvements. (Explain this phenomenon.)

Archimedes' approximation of π. Returning to Archimedes' approximation of π, we note one formula (compare [3]) for the perimeter of the regular 2^n-gon inscribed in a circle of diameter one:

$$p_n = 2^{n-1}\sqrt{2 - x_{n-1}}, \quad n = 2, 3, \ldots \tag{5}$$

where x_n is defined recursively by $x_1 = 0$ and $x_{n+1} = \sqrt{2 + x_n}$. We may let $h = 2^{-n}$ and regard p_n as $T(h)$. By connecting vertices of this regular 2^n-gon with the center of the circle, we get 2^n central angles of size $2\pi/2^n$. Simple trigonometry gives the formula

$$p_n = 2^n \sin\left\{\frac{\pi}{2^n}\right\}.$$

In terms of $h = 2^{-n}$ this becomes

$$T(h) = \frac{1}{h}\sin(\pi h).$$

Using the fact that

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \frac{x^9}{9!} - \cdots \quad \text{(for all } x\text{)}$$

we conclude that

$$T(h) = \frac{1}{h}\left\{\pi h - \frac{(\pi h)^3}{3!} + \frac{(\pi h)^5}{5!} - \frac{(\pi h)^7}{7!} + \frac{(\pi h)^9}{9!} - \cdots\right\}$$

$$= \pi - \frac{\pi^3}{3!}h^2 + \frac{\pi^5}{5!}h^4 - \frac{\pi^7}{7!}h^6 + \frac{\pi^9}{9!}h^8 - \cdots$$

which is in the form of (3) with $n_k = 2k$.

Incorporati
$n_k = 2k$ yields

Example 5.

$T_0(h) = p_n$
2.8284271247
3.0614674589
3.1214451522
3.1365484905
3.1403311567

Formula (.
and thus the
to the subtra
develop a fo
does not hav
also interest
circumscribi
ments on th

Final remar
extrapolatio
is required r
For my nu
to Romberg
integral usir
performed v
to the ones
analysis tex
analysis tex
zoidal rule.
$(b - a)/2$ s

T

when comp
We touc
approxima
In all th
most three
page or di
mathemati
in the Ricl
value. Usir

Incorporating formula (5) for calculating $T(h) = p_n$ in a computer program using $n_k = 2k$ yields the following table.

Example 5. Richardson improvements approximating π.

$T_0(h) = p_n$	$R_1(h)$	$T_1(h)$	$R_2(h)$	$T_2(h)$	$R_3(h)$	$T_3(h)$
2.8284271247						
3.0614674589	3.89	3.1391475703				
3.1214451522	3.97	3.1414377167	15.77	3.1415903931		
3.1365484905	3.99	3.1415829365	15.94	3.1415926179	63.83	3.1415926532
3.1403311567		3.1415920455		3.1415926527		3.1415926533

Formula (5), however, has a fatal flaw: the numbers x_n approach 2 (prove it), and thus the subtraction $2 - x_{n-1}$ inside the radical will cause roundoff error due to the subtraction of nearly equal numbers. The interested reader may wish to develop a formula for p_n (involving only the operations $+$, $-$, $*$, $/$, and $\sqrt{\ }$) that does not have this flaw and see how it affects the Richardson improvements. It is also interesting to develop a formula for the perimeters of the regular 2^n-gons circumscribing a circle of unit diameter and generate the Richardson improvements on them.

Final remarks. This is just an introduction to Richardson improvement and extrapolation techniques. The extremely comprehensive survey article by Joyce [2] is required reading for anyone interested in learning more about this subject.

For my numerical analysis students, the above examples serve as an introduction to Romberg integration, where $T(h)$ is the trapezoidal rule approximation to an integral using equal subintervals of size h and the Richardson improvements are performed with $n_k = 2k$. Specific examples of Romberg integration tables similar to the ones given above (without the ratios) are included in most numerical analysis textbooks. An interesting fact that is glossed over by some numerical analysis texts is that Simpson's rule is a Richardson improvement on the trapezoidal rule. This should be clear from the following hint: use $n_1 = 2$ and $h = (b-a)/2$ so that the trapezoidal rule approximations to $\int_a^b f(x)\,dx$ will be

$$T(h) = \frac{h}{2}\left\{f(a) + f\left[\frac{a+b}{2}\right]\right\} + \frac{h}{2}\left\{f\left[\frac{a+b}{2}\right] + f(b)\right\} \quad \text{and}$$

$$T(2h) = \frac{2h}{2}\{f(a) + f(b)\}$$

when computing $T_1(h)$.

We touch on Richardson's improvement again when we discuss numerical approximations to solutions of differential equations.

In all the examples, the number of Richardson improvements, m, has been at most three. One reason for this restriction on m is to fit the output nicely on a page or display that is 80 characters wide. Another reason for restricting m is mathematical in nature. The termination test will stop the calculation of new rows in the Richardson improvement table when $T_{m-1}(h) - T_m(h)$ is small in absolute value. Using

$$T_m(h) = \frac{2^{n_m} T_{m-1}(h) - T_{m-1}(2h)}{2^{n_m} - 1}$$

and simplifying, we get

$$T_{m-1}(h) - T_m(h) = \frac{T_{m-1}(2h) - T_{m-1}(h)}{2^{n_m} - 1}. \qquad (6)$$

Thus, when there are no restrictions on m, $T_{m-1}(h) - T_m(h)$ will nearly always become small in absolute value simply because the denominator in (6) becomes large. This may terminate the calculations before the desired accuracy is achieved.

Pseudocode for calculation of Richardson improvements on $T(h)$.

```
MaxRows ← 20                                  {The reader may experiment with other values here.}
MaxColumns ← 3                                {With this tolerance we hope for 8 digit accuracy.}
Tol ← 10^-8                                   {Any convenient starting value for h may be used.}
h ← 1/2
D_0,0 ← T(h)                                  {Use i to store the current row number.}
i ← 0                                         {Calculate the Richardson improvements.}
┌ REPEAT
│  h ← h/2
│  i ← i + 1                                  {Store T(h) in column 0 of array D.}
│  D_i,0 ← T(h)
│  m ← MIN(i, MaxColumns)                     {Store T_k(h) in D_i,k.}
│    ┌ DO FOR k = 1 TO m
│    └ D_i,k ← [2^n_k * D_i,k-1 - D_i-1,k-1]/[2^n_k - 1]
└ UNTIL  (i ≥ MaxRows)  OR  (|D_i,m-1 - D_i,m| ≤ Tol * |D_i,m|)
OUTPUT(D_0,0, CrLf)                           {Output followed by carriage return and line feed.}
┌ DO FOR j = 1 TO i - 1                       {The bottom calculated row of array D is i.}
│ OUTPUT(D_j,0)                               {Output without CrLf.}
│    ┌ DO FOR k = 1 TO m                      {Calculate the Ratio = R_k(2^-j-1).}
│    │    ┌ IF  (k ≤ j) THEN
│    │    │ A ← D_j-1,k-1 - D_j,k-1           {Possible numerator for Ratio.}
│    │    │ B ← D_j,k-1 - D_j-1,k-1           {Possible denominator for Ratio.}
│    │    │ IF (B ≠ 0) THEN Ratio ← A/B ELSE Ratio ← 9999
│    │    └ OUTPUT(Ratio, D_j,k)
│    └ END DO
└ OUTPUT(CrLf)                                {Output the bottom row of D separately.}
OUTPUT(D_i,0) END DO                          {We still have m = MIN(i, MaxColumns).}
┌ DO FOR k = 1 TO m
└ OUTPUT(D_i,k)
```


References

1. R. W. Hornbeck, *Numerical Methods*, Quantum, New York, 1975, pp. 20–23.
2. D. Joyce, Survey of extrapolation processes in numerical analysis, *Society for Industrial and Applied Mathematics Reviews* 13 (1971) 435–490.
3. D. Kahner, C. Moler, and S. Nash, *Numerical Methods and Software*, Prentice Hall, Englewood Cliffs, NJ, 1989, p. 16.
4. M. J. Maron, *Numerical Analysis—a Practical Approach* (2nd ed.), Macmillan, New York, 1987, pp. 329–344.
5. L. F. Richardson, The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stress in a masonry dam, *Philosophical Transactions of the Royal Society of London, Series A* 210 (1910) 307–357.
6. ______, Theory of the measurement of wind by shooting spheres upward, *Philosophical Transactions of the Royal Society of London, Series A* 223 (1923) 345–382.
7. ______, How to solve differential equations approximately by arithmetic, *Mathematical Gazette* 12 (1925) 415–421.
8. ______, The deferred approach to the limit. I: Single lattice, *Philosophical Transactions of the Royal Society of London, Series A* 226 (1927) 299–349.