CHAPTER 5

NUMERICAL DIFFERENTIATION

SOLVED EXAMPLE PROBLEMS

for

NUMERICAL METHODS FOR SCIENTISTS AND ENGINEERS With Pseudocodes

By Zekeriya ALTAÇ

November 2024

EXAMPLE 5.1: Determining the Truncation Error of a Difference Formula

Consider a uniformly spaced discrete function $\{f(x_i)\}$. Using the method of Taylor series, (a) derive the truncation error term for the forward difference formula given below and determine the *order of error*. (b) Obtain an expression for optimum spacing (h^*) for which the total (truncation and round-off) error is minimum.

$$f_i' \approx \frac{2f_{i+3} - 9f_{i+2} + 18f_{i+1} - 11f_i}{6h}$$

where $f_{i+k} = f(x_i + kh)$ for $k = 0, ..., 3$.

SOLUTION:

(a) The Taylor series of $f(x_i + kh)$ about x_i can be generalized as

$$f_{i+k} = f_i \pm \frac{kh}{1!}f_i' + \frac{(kh)^2}{2!}f_i'' + \frac{(kh)^3}{3!}f_i''' + \frac{(kh)^4}{4!}f_i^{(4)} + \frac{(kh)^5}{5!}f_i^{(5)} + \frac{(kh)^6}{6!}f_i^{(6)} + \cdots \tag{1}$$

The given difference formula uses three data points to the right of $f(x_i)$, i.e., $f(x_{i+1})$, $f(x_{i+2})$, and $f(x_{i+3})$. The Taylor series approximations of $f(x)$ at these data points are obtained by setting $k = h$, $2h$, and $3h$ in the Eq. (1) generalized formula given above. We start by construction the numerator of Eq. (1), and as shown below all derivatives except f_i' cancel each other out.

$$\begin{aligned}
18f_{i+1} &= \cancel{18f_i} + \cancel{18hf_i'} + \cancel{9h^2f_i''} + \cancel{3h^3f_i'''} + \frac{3}{4}h^4f_i^{(4)} + \frac{3h^5}{10}f_i^{(5)} + \ldots \\
-9f_{i+2} &= -\cancel{9f_i} - \cancel{18hf_i'} - \cancel{18h^2f_i''} - \cancel{12h^3f_i'''} - 6h^4f_i^{(4)} - \frac{12}{5}h^5f_i^{(5)} - \ldots \\
2f_{i+3} &= \cancel{2f_i} + 6hf_i' + \cancel{9h^2f_i''} + \cancel{9h^3f_i'''} + \frac{27}{4}h^4f_i^{(4)} + \frac{81}{20}h^5f_i^{(5)} + \ldots \\
-11f_i &= -\cancel{11f_i}
\end{aligned}$$

which yields

$$2f_{i+3} - 9f_{i+2} + 18f_{i+1} - 11f_i = 6h\boxed{f_i'} + \frac{3f_i^{(4)}}{2}h^4 + \frac{9f_i^{(5)}}{5}h^5 + \ldots$$

or solving for f_i' leads to

$$f_i' = \frac{2f_{i+3} - 9f_{i+2} + 18f_{i+1} - 11f_i}{6h} - \boxed{\frac{h^3}{4}f_i^{(4)}} - \frac{3h^4}{10}f_i^{(4)} - \ldots \tag{2}$$

which involves the finite difference approximation formula and some extra terms on the right-hand side.

The error term that accounts for the *truncation error* is the difference between the exact derivative and the computed approximate value, i.e., $f'_{\text{true}}(x) - f'_{\text{comp}}(x)$. The truncation error of a finite difference formula (also referred to as *discretization error*) can be expressed as $E(h) = C \cdot h^n + \mathcal{O}(h^{n+1})$, where C is a constant that depends on the function being approximated and the finite difference approximation used. Here, the leading error term (i.e., *dominant term* in the truncation error) is $C \cdot h^n$, and by making use of Eq. (2), the truncation error for this example can be expressed as $E(h) = f_i^{(4)}(\xi)h^3/4 + \mathcal{O}(h^4)$. On the other hand, the *order of the error* is determined by the smallest power of h that appears in truncation error. In this example, the leading error is proportional to h^3, where the proportionality constant is $C = f_i^{(4)}(\xi)/4$, so the order of the error is 3. The proposed finite difference formula is said to be a *third-order approximation* and denoted by $\mathcal{O}(h^3)$, i.e., $E(h) = \mathcal{O}(h^3)$.

(b) In order to assess the loss of accuracy during the computation of $\hat{f}'_{\text{comp}}(x)$, we refer to the floating point representation of the function $\hat{f}(x)$. The difference between the true and machine-computed values is

$$f'_{\text{true}}(x) - \hat{f}'_{\text{comp}}(x) = f'_{\text{true}}(x) - \frac{2\hat{f}(x+3h) - 9\hat{f}(x+2h) + 18\hat{f}(x+h) - 11\hat{f}(x)}{6h}$$

Assuming that $\hat{f}(x)$ differs from $f_{\text{true}}(x)$ by a small amount on the order of machine epsilon, i.e., $\hat{f}(x) = f(x)(1+\epsilon)$, where $\epsilon \approx \mathcal{O}(\epsilon_{\text{mach}})$. Substituting the floating point values of the function into the approximation gives

$$\begin{aligned} & f'_{\text{true}}(x) - \frac{2(1+\epsilon_1)f(x+3h) - 9(1+\epsilon_2)f(x+2h) + 18(1+\epsilon_3)f(x+h) - 11(1+\epsilon_4)f(x)}{6h} \\ &= \Big(f'_{\text{true}}(x) - f'_{\text{approx}}(x)\Big) + \frac{2\epsilon_1 f(x+3h) - 9\epsilon_2 f(x+2h) + 18\epsilon_3 f(x+h) - 11\epsilon_4 f(x)}{6h} \end{aligned} \tag{3}$$

where $f'_{\text{approx}}(x)$ has been defined as

$$f'_{\text{approx}}(x) = \frac{2f(x+3h) - 9f(x+2h) + 18f(x+h) - 11f(x)}{6h}$$

Note that the first and second terms in Eq. (3) account for the truncation and round-off errors, respectively. The total error can then be expressed as

$$E_{\text{total}}(h) = E_{\text{trunc}}(h) + E_{\text{rndoff}}(h)$$

where $E_{\text{trunc}}(h) = f'_{\text{true}}(x) - f'_{\text{approx}}(x)$ and

$$E_{\text{rndoff}}(h) = \frac{2\epsilon_1 f(x+3h) - 9\epsilon_2 f(x+2h) + 18\epsilon_3 f(x+h) - 11\epsilon_4 f(x)}{6h}$$

An upper bound for the truncation error, using Eq. (2), is expressed as

$$E_{\text{trunc}}(h) = \frac{h^3}{4}M \tag{4}$$

where $M = \max|f^{(4)}(\xi)|$ on $x \leqslant \xi \leqslant x + 3h$. On the other hand, for very small h, the upper bound for the round-off error is reduced to

$$\begin{aligned} E_{\text{rndoff}}(h) &= \frac{|2\epsilon_1 f(x+3h) - 9\epsilon_2 f(x+2h) + 18\epsilon_3 f(x+h) - 11\epsilon_4 f(x)|}{6h} \\ &\leqslant \frac{|2\epsilon_1 - 9\epsilon_2 + 18\epsilon_3 - 11\epsilon_4|}{6h}N \leqslant \frac{20\varepsilon^*}{3h}N \end{aligned}$$

where ε^* is the upper bound for the relative error in which no underflow or overflow occurs and N is determined as $N = \max|f(x)|$ on $x \leqslant \xi \leqslant x + 3h$, assuming $|f(x+3h)| \approx |f(x+2h)| \approx |f(x+h)| \approx |f(x)|$ for sufficiently small h. Nevertheless, in the above expression, we do not know the signs of ϵ's; thus, we cannot accurately estimate the numerator. But if ϵ_2 and ϵ_4 have opposite signs to ϵ_1 and ϵ_3, the magnitude of the difference could be largest, so we could replace the numerator with $|2\epsilon_1 - 9\epsilon_2 + 18\epsilon_3 - 11\epsilon_4| \cong 40\,\varepsilon^*$.

The total error can now be written as follows:

$$E(h) = \frac{h^3}{4}M + \frac{20\varepsilon^*}{3h}N$$

To determine the condition where the total error is a local minimum, we set $dE/dh = 0$:

$$\frac{dE}{dh} = \frac{3h^2}{4}M - \frac{20\varepsilon^*}{3h^2}N = 0$$

Solving for h, we find

$$h^* = \left(\frac{80\,\varepsilon^* N}{9M}\right)^{1/4} \tag{5}$$

Discussion: Finding the optimum spacing (also referred to as the step size, h) for finite difference formulas is essential for achieving a good balance between *accuracy* and computational *efficiency*. The key goal is to minimize the error in approximating the derivative or other quantities while avoiding excessive computational cost due to extremely small h.

Minimizing Error of Approximations: The optimal value of h typically minimizes the total error, which involves both truncation error (from the finite difference approximation) and round-off error (from finite precision arithmetic). Truncation error typically decreases as h gets smaller, but too small a value of h increases the round-off error due to the limitations of floating-point precision. The optimal step size typically balances the truncation error and round-off error for the machine precision you are using.

Factors Affecting Optimum Step Size: The optimal step size is influenced by the order of the method (i.e., $\mathcal{O}(h^n)$), approximation-specific constants (i.e., M and N), and the working precision in the computation. Higher-order methods (larger n) allow for larger h while maintaining accuracy but may amplify round-off errors if h is too small. The total error depends on the magnitude of constants M and N, which are influenced by the function being approximated and the finite difference approximation adopted. Systems with higher precision (e.g., double vs. single precision) reduce ϵ's, allowing for smaller h without significant round-off error.

Practical Considerations: Estimating the exact values of constants such as C, M, or N may not always be known, but the general relationship guides the choice of h. In practical implementations, adaptive methods dynamically adjust h to minimize total error based on local error estimates. As a guideline when working in double precision, it is to use $h^* \approx 10^{-5}$ for $\mathcal{O}(h)$ and $h^* \approx 10^{-3}$ for $\mathcal{O}(h^2)$ approximations. Selecting h near these values provides a good balance between truncation and round-off errors for many standard problems.

EXAMPLE 5.2: Finding the Optimum Spacing

Using the finite difference approximation given in Example 5.1, calculate $f'(1)$, the derivative of the function $f(x) = x^2 \sinh x$, the true and leading truncation error, and round-off error estimates for $h = 10^{-m}$, where $m = 1, 2, \cdots, 10$. Theoretically and numerically estimate the value of h where the total error becomes minimum.

SOLUTION:

We will estimate the $f'(1)$ derivative using the following difference formula:

$$f'_{\text{approx}}(1) = \frac{2f(1+3h) - 9f(1+2h) + 18f(1+h) - 11f(1)}{6h}$$

The truncation error, given by Eq. (4) in Example 5.1, requires an analytical expression for $f^{(4)}(x)$ and determining $M = \max|f^{(4)}(\xi)|$ on $1 \leqslant \xi \leqslant 1+3h$, where $f^{(4)}(x) = \left(x^2+12\right)\sinh x + 8x\cosh x$. On the other hand, the round-off error term requires $N = \max|f(\xi)|$ on the same interval. It should be noted that both $f(x)$ and $f^{(4)}(x)$ are positive for all $x > 0$ and their absolute maximums appear at the right endpoint.

Table 5.1

h	$f'_{\text{approx}}(1)$	$E_{\text{true}}(h)$	$E_{\text{trunc}}(h)$	$E_{\text{rndoff}}(h)$
10^{-1}	3.901837069	8.3540×10^{-3}	-0.01093791	5.3923×10^{-15}
10^{-2}	3.893490061	7.0392×10^{-6}	-7.2430×10^{-6}	2.4356×10^{-14}
10^{-3}	3.893483029	6.9178×10^{-9}	-6.9387×10^{-9}	2.2298×10^{-13}
10^{-4}	3.893483022	9.8290×10^{-12}	-6.9089×10^{-12}	2.2100×10^{-12}
6×10^{-5}	3.893483022	3.9080×10^{-12}	-1.4920×10^{-12}	3.6819×10^{-12}
10^{-5}	3.893483022	1.9635×10^{-10}	-6.9059×10^{-15}	2.2080×10^{-11}
10^{-6}	3.893483022	4.8459×10^{-10}	-6.9056×10^{-18}	2.2078×10^{-10}
10^{-7}	3.893483038	1.6095×10^{-8}	-6.9056×10^{-21}	2.2078×10^{-9}
10^{-8}	3.893482899	1.2305×10^{-7}	-6.9056×10^{-24}	2.2078×10^{-8}
10^{-9}	3.893483758	7.3552×10^{-7}	-6.9056×10^{-27}	2.2078×10^{-7}
10^{-10}	3.893481093	1.9290×10^{-6}	-6.9056×10^{-30}	2.2078×10^{-6}

The f'_{approx} value, true error, truncation, and round-off errors are computed and tabulated in Table 5.1 for exponentially decreasing h values. The true, truncation, and round-off errors are calculated by the following expressions:

$$E_{\text{true}}(h) = 2\sinh 1 + \cosh 1 - f'_{\text{approx}}(1), \quad E_{\text{trunc}}(h) = \frac{h^3}{4}M, \quad E_{\text{rndoff}}(h) = \frac{20\varepsilon^*}{3h}N$$

where $N = f(1+3h)$ and $M = f^{(4)}(1+3h)$, and the machine epsilon for double precision evaluation is determined to be $\varepsilon^* = 2.818\times10^{-17}$.

In inspecting the tabulated values, we notice that when h is reduced by a factor of 10, the truncation error is reduced by a factor of 10^3. On the other hand, the true error is also reduced by a factor of 10^3, at least as long as h is not too small. However, when h becomes smaller than 6×10^{-6}, the true error increases. The effect of the round-off error increases as h is reduced due to arithmetic with very small numbers, which in turn amplifies the accumulation of round-off error. Round-off error typically scales as $\mathcal{O}(1/h)$, as smaller h increases the sensitivity of the numerical computation to finite precision. The truncation error dominates when h is large, and reducing h decreases the total error. When h is small, the round-off errors dominate the total error.

Using Eq. (), the optimum spacing is obtained as

$$h^* = \sqrt[4]{\frac{80(2.818 \times 10^{-17})(1.175)}{9(27.64)}} = 5.71 \times 10^{-5} \approx 6 \times 10^{-5}$$

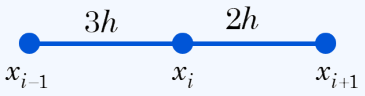
which is compatible with the tabulated results.

Discussion: Calculating the optimum spacing (step size) in numerical differentiation is theoretically possible and can be practical in certain situations, where $f(x)$ is known. However, it also has limitations in real-world applications.

The truncation error depends on the smoothness of $f(x)$ and its higher-order derivatives, which may not be readily known, especially if $f(x)$ is a discrete function. The round-off error depends on the machine precision and floating-point representation, which are usually known, but its impact varies based on implementation details. On the other hand, if $f(x)$ is highly nonlinear or has regions with rapid changes, the optimum value may not be easily determined or implemented. Overall, the decision on the use of optimum value depends on the problem, required precision, and computational resources.

EXAMPLE 5.3: Deriving Finite-Difference Formulas using Polynomials

As depicted below, a discrete function $f(x)$ is defined at nodal points (x_{i-1}, f_{i-1}), (x_i, f_i), and (x_{i+1}, f_{i+1}), where $x_i - x_{i-1} = 3h$ and $x_{i+1} - x_i = 2h$. Develop second-order accurate central difference formulas for $f'(x_i)$ and $f''(x_i)$.



SOLUTION:

To approximate the distribution passing through these nodal points, we use a 2nd degree polynomial $f(x) \cong P_2(x) + \mathcal{O}(h^2) = ax^2 + bx + c$. To determine ($a$, b, and c) coefficients, we need to make use of three nodal points: $f(x-h)$, $f(x)$, and $f(x+2h)$.

Using the shifted coordinates, we can write $x_i = 0$, $x_{i-1} = -3h$, and $x_{i+1} = 2h$. Then, we obtain the following three equations and three unknowns:

$$\begin{aligned} f_{i-1} &= f(-3h) = a(-3h)^2 + b(-3h) + c \\ f_{i+1} &= f(2h) = a(2h)^2 + b(2h) + c \\ f_i &= f(0) = a(0^2) + b(0) + c \end{aligned}$$

The solution of this linear system for a, b, and c yields

$$a = \frac{2f_{i-1} - 5f_i + 3f_{i+1}}{30h^2}, \qquad b = \frac{-4f_{i-1} - 5f_i + 9f_{i+1}}{30h}, \qquad c = f_i$$

To obtain approximations for the derivatives in question, the $f'(x_i)$ and $f''(x_i) = 2a$ derivatives are evaluated at $x_i = 0$:

$$f_i' = f'(0) = P_2'(0) = 2a(0) + b = b \qquad \text{and} \qquad f_i'' = f''(0) = P_2''(0) = 2a$$

yields

$$f_i' = \frac{-4f_{i-1} - 5f_i + 9f_{i+1}}{30h} + \mathcal{O}(h^2), \qquad f_i'' = \frac{2f_{i-1} - 5f_i + 3f_{i+1}}{15h^2} + \mathcal{O}(h^2)$$

Both difference formulas are second-order.

Discussion: Polynomials play a critical role in deriving finite difference formulas due to their simplicity, flexibility, and effectiveness in approximating smooth functions. Furthermore, the truncation error in finite difference formulas can be precisely determined and controlled using polynomials. Higher-degree polynomials enable higher-order accuracy, allowing finite difference schemes to be tailored for specific problems.

Polynomials can handle non-uniform grids (unequal spacing between points). Lagrange polynomials, in particular, are useful for deriving finite difference formulas on irregular grids.

EXAMPLE 5.4: Estimating the Diffusion Coefficient for a Cementation Process

The diffusion of carbon into iron is governed by the following partial differential equation:

$$\frac{\partial C}{\partial t} = D\frac{\partial^2 C}{\partial x^2}$$

where $C(x,t)$ is the concentration (wt%) and D is the diffusion coefficient (m^2/s). Two identical sheets (S1 and S2) with the same size and content were subjected to the cementation process under the same conditions, S1 for one hour and S2 for 2 hours. The carbon concentration as a function of x (depth) has been determined and tabulated in the table below:

Distance (mm)	Sample S1 (1 h) Concentration, (wt%)	Sample S2 (2 h) Concentration (wt%)
0	0.1	0.1
0.5	0.0565	0.0677
1.0	0.0239	0.0405
1.5	0.0073	0.0208
2.0	0.0035	0.0096

Use time and space derivatives to roughly estimate the diffusion coefficient.

SOLUTION:

Here we will use the derivative to get an approximate answer. The diffusion process is an unsteady process, and the governing differential equation for diffusion atoms into a specimen that can be considered planar is given in the question statement.

We approximate the partial derivatives at $t = 1$ h with respect to time using the forward difference formula: $\partial C_i/\partial t \approx (C_{2i} - C_{1i})/\Delta t$ and the partial derivative with respect to x using the central difference formula: $\partial^2 C_i/\partial x^2 \approx (C_{1,i+1} - 2C_{1i} + C_{1,i-1})/(\Delta x)^2$, where $\Delta t = 1\,\text{h} = 3600\,\text{s}$ and $\Delta x = 0.5\,\text{mm} = 0.0005\,\text{m}$. The diffusion coefficient is then calculated from $D_i = (\partial C_{1i}/\partial t)/(\partial^2 C_{1i}/\partial x^2)$, and all the results computed are tabulated in Table 5.2. We note that the diffusion coefficients computed at different depths are different, which is understandable since the data can involve experimental uncertainty and numerical error due to truncation errors.

Table 5.2

i	x_i (mm)	C_{1i}	C_{2i}	$\partial C_i/\partial t$	$\partial^2 C_i/\partial x^2$	D_i (m^2/s)
0	0	0.1	0.1	0		
1	0.5	0.0565	0.0677	3.1111×10^{-6}	4.36×10^{4}	7.14×10^{-11}
2	1.0	0.0239	0.0405	4.6111×10^{-6}	6.40×10^{4}	7.20×10^{-11}
3	1.5	0.0073	0.0208	3.7500×10^{-6}	5.12×10^{4}	7.32×10^{-11}
4	2.0	0.0035	0.0096	1.6944×10^{-6}		

Discussion: In experimental settings, the data are discrete due to the nature of measurements, sampling intervals, or finite observation points. The accuracy of discrete (experimental) data depends on how closely the measured values represent the true values of the physical quantities being investigated. Likewise, the accuracy of any numerical calculation depends on the accuracy of the experimental data. In this example problem, the use of finite differences was presented; however, there are more elaborate mathematical methods as well as numerical techniques for estimating parameters of engineering importance, such as the diffusion coefficient in this case.

EXAMPLE 5.5: Differentiating Nonuniformly Spaced Discrete Functions

The growth of the world population is one of the urgent questions to be answered today in order to be able to make future plans for almost everything. Will the population continue to grow? Or will it perhaps stabilize at some point, and if so, when?

A simple mathematical model (referred to as the *exponential growth model*) postulates that the rate of change of the population is proportional to the population, which seems to be pretty reasonable. Because for a small community of people, there will be fewer births and deaths, so the rate of change will naturally be small. However, for a large community, there will be more births and deaths; we can then expect a larger rate of change. Thus, if we denote $P(t)$ as the population t years after the year 1950, we may express this assumption as

$$\frac{dP}{dt} = kP$$

where k is a proportionality constant (i.e., the rate at which the population grows).

The population data* for planet Earth since 1950 is given below: (a) Use the given data to calculate k, the relative population growth rate for the years 1950 to 2020. (b) According to this model, when will the Earth's population reach 10 billion? (c) What does this model predict for the population in the year 2300?

1950	1955	1960	1970	1975	1985	1990	2000	2005	2009	2015	2019	2020
2.49	2.74	3.02	3.69	4.07	4.87	5.33	6.17	6.59	6.93	7.47	7.81	7.89

* https://ourworldindata.org/population-growth

SOLUTION:

(a) The proportionality constant k in the exponential model has an important meaning when it is expressed as $k = (dP/dt)/P$, which gives the ratio of the *rate of change* to the *population*. This parameter k is often referred to as *per capita growth rate*.

The exponential model assumes a constant per capita growth rate, meaning that the per capita growth rate is the same regardless of the population size, i.e., $k \neq f(t)$. To determine k in this example, we need to estimate the rate of change in population, dP/dt, as well as $(dP/dt)/P$ for every available data point. We begin by noting that the population data in question is a *non-uniformly spaced* discrete distribution. The first derivative for interior data points can be estimated using the central difference formula, Eq. (5.46), which applies to this problem as

$$\frac{dP_i}{dt} \cong \frac{h_1^2 P_{i+1} + \left(h_2^2 - h_1^2\right) P_i - h_2^2 P_{i-1}}{h_1 h_2 \left(h_1 + h_2\right)}$$

where $h_1 = t_i - t_{i-1}$, $h_2 = t_{i+1} - t_i$, and the order of truncation error is $\mathcal{O}(h_1 h_2)$.

For the left and right endpoints, we employ the following second-order forward and backward difference formulas, Eqs. (5.44) and (5.48), respectively:

$$\frac{dP_0}{dt} \cong \frac{-3P_0 + 4P_1 - f_2}{10} \qquad \text{and} \qquad \frac{dP_{12}}{dt} \cong \frac{24P_{12} - 25P_{11} + P_{10}}{20}$$

Note that the interval spacings are $h_1 = h_2 = 5$ in the forward difference and $h_1 = 1$ and $h_2 = 4$ in the backward difference formulas.

For $i = 0$ to 12, the computed values of (dP_i/dt) and $(dP_i/dt)/P_i$ are tabulated in Table 5.3. The derivatives at the endpoints can be computed using the forward and backward difference formulas.

Table 5.3

i	t_i	P_i	dP_i/dt	$(dP_i/dt)/P_i$
0	1950	2.49	0.04700	0.01888
1	1955	2.74	0.05300	0.01934
2	1960	3.02	0.05967	0.01976
3	1970	3.69	0.07300	0.01978
4	1975	4.07	0.07733	0.01900
5	1985	4.87	0.08800	0.01807
6	1990	5.33	0.08933	0.01676
7	2000	6.17	0.08400	0.01361
8	2005	6.59	0.08456	0.01283
9	2009	6.93	0.08700	0.01255
10	2015	7.47	0.08700	0.01165
11	2019	7.81	0.08100	0.01037
12	2020	7.89	0.07900	0.01001

Our first observation is that the true growth rate of Earth's population is not constant. It increases until the year 1970 and then exhibits a decreasing trend. The average value of the growth rate for the 1970-2020 period is 0.014463. As any population of living things grows, it is expected that the per capita growth rate will decrease because there will not be enough resources to support that many lives. It is clear that a mathematical model that assumes that the per capita growth rate depends on population P would be more realistic.

(b) Assuming that the growth rate will remain constant at 0.01001 after the year 2020 and incorporating the first-order forward difference approximation for the derivative in the mathematical model, we may write

$$\frac{P(t) - P_0}{\Delta t} = \frac{10 - 7.89}{\Delta t} \cong 0.01001\, P(t) = 0.01001(10) = 0.1001$$

or

$$\Delta t = 2.11/0.1001 \cong 21 \text{ years}$$

In other words, this model estimates that the population will be 10 billion in the year 2041. However, notice that this model has an exact solution: $P(t) = P(t_0)\, e^{k(t-t_0)}$, where t_0 is the initial point. Using the exact solution, we can also make another (perhaps better) prediction. Setting $t_0 = 2020$ and noting $P(t) = 10$, $P(t_0) = 7.89$, and $k = 0.01001$, the exact solution of the mathematical model points to $\cong$2044 (or 23.68 years after 2020). Using the exact solution eliminates the truncation error (about 3 years) resulting from applying the forward difference formula in our first estimate.

(c) Setting $t = 2300$ and assuming that k remains constant, we find

$$\frac{P(2300) - P(2020)}{t - 2020} = \frac{P(2300) - 7.89}{2300 - 2020} \cong 0.01001(7.89) \quad \text{and} \quad P \cong 30 \text{ billion}$$

or using the exact solution of the model, we find

$$P(2300) = P(2020)\, e^{0.01037(2300-2020)} = 7.89 e^{0.01001(280)} \cong 130 \text{ billion}$$

There is a large difference between these two estimates. Note that since $\Delta t = 280$ is such a large value, the first-order forward difference approximation should be expected to yield a large truncation error.

Discussion: The exponential model gives estimates that are fairly accurate for the years relatively close to 2020, i.e., small Δt. But, if we go too far into the future (large Δt), the model with either numerical or exact solution will yield larger rates of change, causing the population to grow arbitrarily large. However, predictions made with this model cannot be realistic in that the Earth would not be able to support such a large population. Thus, we should consider improving the mathematical model.

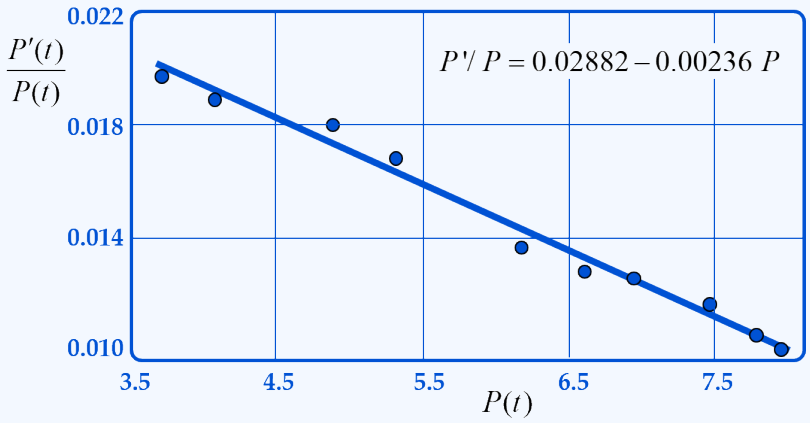


Figure 5.1

We begin by noting that per capita growth rate has been decreasing (almost linearly) since 1970 (*see* Figure 5.1). (This is due to declining fertility rates in many countries, especially in developed and developing countries.) Fitting the data between 1970-2020 (i.e., P_i vs. $(dP_i/dt)/P_i$) using the linear least squares (*see* Chapter 7), we obtain

$$\frac{dP/dt}{P} = (a - b\,P) \qquad \text{or} \qquad \frac{dP}{dt} = P\,(a - b\,P)$$

where $a = 0.02882$, $b = 0.00236$ with a correlation coefficient of $r^2 = 0.9844$.

The exact solution of this mathematical model, referred to as the *logistic growth model*, given as

$$P(t) = \frac{aP_0}{bP_0 + (a - b\,P_0)e^{-a(t-t_0)}}$$

where $P_0 = P(t_0)$.

To find the year when the Earth's population would reach 10 billion, we set $t_0 = 2020$, $P_0 = 7.89$, and $P(t) = 10$ in the above model and solve it for t, which yields 2051.5 (i.e., about 8 years later than the exponential model predicted). On the other hand, the Earth's population in 2300 is estimated to be 12.21 billion, which is significantly lower than the exponential growth model. This is because the logistic growth model accounts for environmental limits, where population growth slows down as the population size approaches a maximum capacity. This exponential growth model, on the other hand, is often unrealistic because it assumes there are no limiting factors like food shortages or space constraints, and so on.

EXAMPLE 5.6: Using the Richardson Extrapolation to Evalaute Derivatives

The temperature-dependent specific Gibbs energy at $p_0 = 0.1$ MPa is given by Patek *et al.** as

$$g_0(T) = g(p_0, T) = RT_R \left(c_1 + c_2\tau + c_3\tau\log(\tau) + \sum_{i=1}^{3} a_i\alpha^{n_i} + \sum_{i=1}^{4} b_i\beta^{m_i} \right)$$

where $\tau = T/T_R$, $\alpha = T_R/(T_a - T)$, $\beta = T_R/(T - T_b)$, $T_R = 10$ K, $T_a = 593$ K, $T_b = 232$ K, $R = 461.51805$ J/kg·K, and the rest of the data are given in the table below.

The isobaric heat capacity c_{p0}, entropy s_0 and enthalpy h_0 are calculated by the relationships:

$$c_{p0} = -T\frac{d^2 g_0}{dT^2}, \qquad s_0 = -\frac{dg_0}{dT}, \qquad \text{and} \qquad h_0 = g_0 + Ts_0$$

To calculate c_{p0}, s_0, and h_0 at $T = 375$ K, start with $\Delta T = 2$ K and use Richardson extrapolation to estimate the first and second derivatives of $g_0(T)$ with respect to T within $\varepsilon = 10^{-4}$. Apply the central difference formulas for both derivatives.

i	n_i	m_i	a_i	b_i	c_i
1	4	2	-1.661470539×10^5	-0.823742626	-245.2093414
2	5	3	$+2.708781640\times10^6$	$+1.908956353$	$+38.69269598$
3	7	4	-1.557191544×10^8	-2.017597384	-8.983025854
4		5		$+0.8546361348$	

* Patek, J., Hruby, J., Klomfar, J., Souckova, M., Harvey, H. A., *Reference Correlations for Thermophysical Properties of Liquid Water at 0.1 MPa*, J PHYS CHEM REF DATA, vol. 38(3), pp. 21-29, 2009.

SOLUTION:

As you see, the specific Gibbs energy of liquid water at p_0 is given as a complex function with many parameters and coefficients. Analytical differentiations of $g_0(T)$ can become even more complicated, so numerical differentiation may be regarded as a more attractive alternative.

To start the extrapolation process, we require two estimates of dg_0/dT using the CDF, one with $h = \Delta T = 2$ K and another with $h = \Delta T = 1$ K.

$$D_{0,0} = \frac{g_0(375+2) - g_0(375-2)}{2(2)} = \frac{-73737.3935 - (-68425.2003))}{4} = -1328.0483$$

$$D_{1,0} = \frac{g_0(375+1) - g_0(375-1)}{2(1)} = \frac{-72392.4874 - (-69736.3640)}{2} = -1328.0617$$

Using these estimates, an improved estimate is obtained by applying the (first-step) Richardson extrapolation as

$$D_{1,1} = \frac{4D_{1,0} - D_{0,0}}{3} = \frac{4(-1328.0617) - (-1328.0483))}{3} = -1328.06616$$

The estimated error at the end of the first step is $|D_{1,1} - D_{0,0}| = 0.017854 > 10^{-4}$. To obtain a second step (improved) estimate $D_{2,0}$, dg_0/dT is calculated using the CDF with $h = \Delta T = 0.5$ K. Subsequently, $D_{2,1}$ and $D_{2,2}$ can be found using the Richardson extrapolation formula. The results are presented in Table 5.4. Notice that $D_{1,1}$ and $D_{2,2}$ estimates of $dg_0(375)/dT$ are correct to six decimal places. In fact, we have found a result that is more accurate than the desired level of accuracy ($|D_{2,2} - D_{1,1}| = 5.3\times10^{-8} < 10^{-4}$).

Table 5.4: Richardson extrapolation table for dg_0/dT at $T = 375$.

k	h	$\mathcal{O}(h^2)$	$\mathcal{O}(h^4)$	$\mathcal{O}(h^6)$
0	2	-1328.048308		
1	1	-1328.061699	-1328.066162	
2	0.5	-1328.065046	-1328.066162	-1328.066162

Next, in order to estimate $d^2g_0(375)/dT^2$, we similarly use the CDF to obtain two estimates of d^2g_0/dT^2 with $h = \Delta T = 2$ K and $h = \Delta T = 1$ K.

$$\begin{aligned} D_{0,0} &= \frac{g_0(375+2) - 2g_0(375) + g_0(375-2)}{2^2} \\ &= \frac{-73737.3935 - 2(-71058.802079) + (-68425.2003))}{4} = -11.247398530 \\ D_{1,0} &= \frac{g_0(375+1) - 2g_0(273) + g_0(375-1)}{1^2} \\ &= -72392.4874 - 2(-71058.802079) + (-69736.3640) = -11.247343579 \end{aligned}$$

Using the above estimates, the Richardson extrapolation yields

$$D_{1,1} = \frac{4D_{1,0} - D_{0,0}}{3} = \frac{4(-11.247343579) - (-11.247343579))}{3} = -11.24732525$$

The estimated error at the end of the first step is $|D_{1,1} - D_{0,0}| = 7.33 \times 10^{-5} < 10^{-4}$. Hence, at this point the extrapolation process is terminated, leading to $d^2g_0(375)/dT^2 \cong -11.2473252$. Now that we have obtained the required derivatives, we can go ahead and calculate the properties asked as follows:

$$\begin{aligned} s_0 &= -\frac{dg_0}{dT} = 1328.066162 \ \frac{\text{J}}{\text{kg} \cdot \text{K}} \\ c_{p0} &= -T\frac{d^2g_0}{dT^2} = (-375 \text{ K})\left(-11.24732525 \ \frac{\text{J}}{\text{kg} \cdot \text{K}^2}\right) = 4217.74696875 \ \frac{\text{J}}{\text{kg} \cdot \text{K}} \\ h_0 &= g_0 + Ts_0 = -71058.802079 \ \frac{\text{J}}{\text{kg}} + (375 \text{ K})\left(1328.066162 \ \frac{\text{J}}{\text{kg} \cdot \text{K}}\right) \\ &= 426966.008671 \ \frac{\text{J}}{\text{kg}} \end{aligned}$$

Discussion: In many real-world problems, such as in this example, functions can be complex. A numerical differentiation process is resorted to in cases where analytical differentiation is difficult or impossible to obtain. However, finite difference formulas have their own disadvantages, which affect the accuracy and reliability of the computed results.

Finite difference formulas are approximations that introduce truncation error, whose magnitude depends on the choice of the step size h, the type (FDF, BDF, or CDF), and the order of the finite difference formula, $\mathcal{O}(h^n)$, used. One way to improve accuracy is to reduce the step size. When the step size is reduced, round-off errors can lead to deterioration of the overall accuracy of the derivative due to floating-point precision limits or round-off errors. So one must balance step size to avoid both truncation and round-off errors, which in general requires preparatory work to find optimum spacing h^*. However, as a guideline for smooth functions, $h^* \cong 10^{-5}$ and $h^* \cong 10^{-3}$ can be used for finite difference approximations of order $\mathcal{O}(h)$ and $\mathcal{O}(h^2)$, respectively, provided that you are working in

double precision. Selecting h near the stated h^* values can provide a good balance between truncation and round-off errors for many standard problems.

Richardson extrapolation is a numerical technique to accelerate the convergence of the approximation by combining results at different step sizes to eliminate the leading error terms, leaving behind a more accurate estimate. To be specific, if $\mathcal{O}(h^n)$ is the order of a numerical approximation of the derivative in question (i.e., $D(h) + \mathcal{O}(h^n)$), then Richardson extrapolation gives an approximation with leading error $\mathcal{O}(h^{n+1})$:

$$D(h) = \frac{2^n D(h/2) - D(h)}{2^n - 1} + \mathcal{O}(h^{n+1}) \tag{A}$$

For instance, if a difference formula (like FDF or BDF) of order $\mathcal{O}(h)$ is used to start the differentiation process, the Richardson extrapolation procedure will reduce the error to $\mathcal{O}(h^2)$. Likewise, using difference formulas of order $\mathcal{O}(h^2)$ (like CDF or 2nd order FDF/BDFs) to start the extrapolation procedure will reduce the error to $\mathcal{O}(h^4)$. Notice that Eq. (A) is especially suitable for adaptive computations. Successive extrapolations of order $\mathcal{O}(h^6)$, $\mathcal{O}(h^8)$, and so on can be obtained very easily. Furthermore, computation of $D_{k,m}$ is faster and cheaper (due to fewer function evaluations) than using equivalent higher-order finite difference equations. Thus, adaptive algorithms can be devised that efficiently adjust the step size to meet the specified accuracy or tolerance. In this method, round-off errors are less of a concern since high accuracy results can generally be obtained with relatively large values of h.

Overall, Richardson extrapolation offers a more stable approach and can help mitigate the issues related to truncation and round-off errors by leveraging multiple approximations at different step sizes, reducing the impact of such errors. That is why the method is of vital importance in scientific/engineering applications, where *high precision* in derivative estimates is required. In other words, it is most useful when aiming for *high accuracy* in derivative approximations, especially when dealing with lower-order methods or small step sizes and you are willing to perform additional function evaluations to achieve higher accuracy.