Chapter 10
Numerical Solution Methods for Engineering Analysis

Chapter Learning Objectives

- Learn the alternative ways of using numerical methods to solve nonlinear equations, perform integration, and solve differential equations.
- Learn the principles of various numerical techniques for solving nonlinear equations, performing integrations, and solving differential equations by the Runge–Kutta methods.
- Learn the fact that numerical methods offer approximate but credible accurate solutions to the problems that are not readily or possibly solved by closed-form solution methods.
- Learn the fact that numerical solutions are available to the users only at the preset solution points, and the accuracy of the solutions is largely dependent on the size of the increments of the variable selected for the solutions.
- Become familiar with the value of commercially available numerical solution software packages such as Mathematica and MatLAB.

10.1 Introduction

Numerical methods are techniques for solving the mathematical problems involved in engineering analysis that cannot be solved by closed-form solution methods such as those presented in the preceding chapters. In this chapter, we will learn the use of some of the available numerical methods that will not only enable engineers to solve many mathematical problems, but also allow engineers to minimize the need for the many hypotheses and idealization of the conditions, such as those stipulated in Section 1.4.

Numerical techniques have greatly expanded the types of problems that engineers can solve, as illustrated in a number of publications in the open literature (Chapra, 2012; Ferziger, 1998; Hoffman, 1992). The goal of this chapter is to present to readers the basic principles of some of these techniques that are frequently used in engineering analysis. The author's own experience indicates that engineers who understand the principles of the numerical methods are usually more effective and intelligent users of these methods than most technical personnel who are trained to carry out the same computational assignments using “turn-key” software packages such as the finite-difference method and the MatLAB software package in Appendix 4. This chapter will cover the principles of commonly used numerical techniques for (1) the solution of nonlinear polynomial and transcendental equations; (2) integration involving complex forms of functions; and (3) the solution of differential equations by the basic finite-difference schemes and the Runge–Kutta methods. The chapter will also cover the overviews of two popular commercial software packages called Mathematica and MatLAB.

The principal task in numerical methods for engineering analysis is to develop algorithms that involve arithmetic and logical operations so that such operations can be performed at incredible speed by digital computers with enormous data storage capacities. Because readers of this book are expected to be users of numerical methods, we will present only the principles that are relevant to the development of these algorithms, not the theories and the proof of these methods.

10.2 Engineering Analysis with Numerical Solutions

Most engineering problems require enormous computational effort when numerical methods are used. Digital computers are essential tools for obtaining numerical solutions. Digital computers have incredible power in computational speed and enormous memory capacity. Unfortunately, these machines have no intelligence of their own, and they are not capable of making independent judgment on their own. Additionally, engineers need to realize the fact that digital computers can only perform simple arithmetic operations with (+, −, ×, ÷) and handle the logical flow of data. It cannot perform higher mathematical operations even in such simple cases as evaluating exponential and trigonometric functions without proper algorithms that convert the evaluation of these functions into simple arithmetic operations; thus, all complicated mathematical operations have to be converted into simple arithmetical operations. Numerical methods that enable engineers to develop algorithms for various mathematical functions and operations using digital computers have thus become essential knowledge and skills for solving many advanced engineering problems using mathematical tools.

Despite the fact that numerical techniques have greatly expanded the types of problems that engineers can handle as mentioned in Section 10.1, users need to be aware of many unique characteristics of these methods, as outlined in the following.

1. Numerical solutions are available only at selected (discrete) solution points of the domain that is being investigated, not at all points in the entire domain covered by the functions as is the case with analytical solution methods described in Chapters 7, 8, and 9.

2. Numerical methods are essentially "trial-and-error" processes. Typically, users need to know the initial and boundary conditions that the intended solution will cover. The selection of increments of the variable at which the solution points are positioned is critical in the solution of the problem. Unstable numerical solutions may result from improper selection of such increments, called the step sizes with solutions.

3. Most numerical solution methods result in some error in the solutions. Two types of error are inherent with numerical solutions:

a. *Truncation errors* – because of the approximate nature of numerical solutions of many engineering problems, these solutions often consist of both lower-order terms and higher-order terms. The latter terms are often dropped in the computations in order to achieve computational efficiency, resulting in error in the solution.

b. *Round-off errors* –Most digital computers handle numbers either with 7 places or 14 decimal places in numerical solutions. In the case of a 32-bit computer with double precision (i.e., numbers of 14 decimal places), any number after the 14th decimal point will be dropped. This may not sound like a big deal, but if a huge number of operations are involved in the computation, such error can accumulate and result in significant error in the end results.

Both of these types of error are of cumulative nature. Consequently, errors in numerical solution may grow to be significant in solutions obtained after many step increments.

10.3 Solution of Nonlinear Equations

Often, engineers need to solve nonlinear equations in their analyses. These equations can be as simple as quadratic equations such as in Equation (8.3) with the two roots expressed in Equation 8.4. There is also the need to find roots of equations in higher-order polynomial functions such as shown in Equation 10.1, relating to the problem of locating the marking on a measuring cup, to be described in Example 10.3 and in Figure 10.4:

$$L^3 + 70.3L^2 + 1647.39L - 18656.72 = 0 \tag{10.1}$$

or solution of the time t_f required for the mass to rupture from its spring attachment in the case of resonant vibration analysis that was illustrated in Example 8.9:

$$\left(0.005 - \frac{t_f}{20}\right)\cos 10t_f + \frac{1}{200}\sin 10t_f - 0.03 = 0 \tag{10.2}$$

Solutions of nonlinear equations such as Equations 10.1 and 10.2 may be obtained by setting expressions $f(x) = 0$, and finding their roots (i.e., L in Equation 10.1 and t_f in Equation 10.2) located at the points of cross-over of the function $f(x)$ and the x-coordinate axis, as illustrated in Figure 10.1.

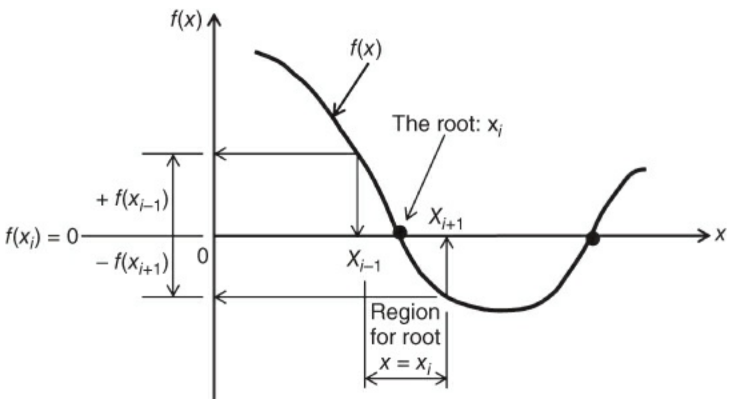

Figure 10.1 Root of a nonlinear equation $f(x) = 0$.

We will use the following two methods for the solution of the nonlinear equation given in Equation 10.2

10.3.1 Solution Using Microsoft Excel Software

Because the roots of an equation represented by $f(x) = 0$ are located at the intersections of the function $f(x)$ and the x-coordinate axis as illustrated in Figure 10.1, one may use the widely available tool of spreadsheets (such as Microsoft Office Excel) to locate the roots of the equation by evaluating the function $f(x)$ with for various values of the variable x. The range in which the roots of the equation $f(x) = 0$ lie can be identified as the values where $f(x)$ change sign from positive to negative or vice versa, as illustrated for the range (x_{i-1}–x_{i+1}) in Figure 10.1. More accurate values of the roots may be found by repeating the computation of the function values with smaller increments of the variable x within that range. This method, though sounds tedious, involves straightforward evaluation of the function $f(x)$ with an initially estimated value of x_i by using the Excel software that has enormous computational speed to identify the range in which the roots are located, as will be illustrated by Example 10.1.

Example 10.1

Solve the nonlinear polynomial Equation 10.3:

$$x^4 - 2x^3 + x^2 - 3x + 3 = 0 \qquad (10.3)$$

using a Microsoft Excel spreadsheet.

Solution:

As in many other numerical solution methods, we begin with an estimate of the root of the equation at $x = 0$ and assign an increment of 0.5 in variable x in our evaluation of the function $f(x) = x^4 - 2x^3 + x^2 - 3x + 3$. The spreadsheet indicates the values of $f(x)$ tabulated in the right-hand portion of Figure 10.2 with an assumed starting point at $x = 0$:

x	$f(x)$
0	3.00
0.5	1.56
1.0	0
1.5	−0.94
2.0	1.00
2.5	9.56
3.0	30.00
3.5	69.06
4.0	135.00

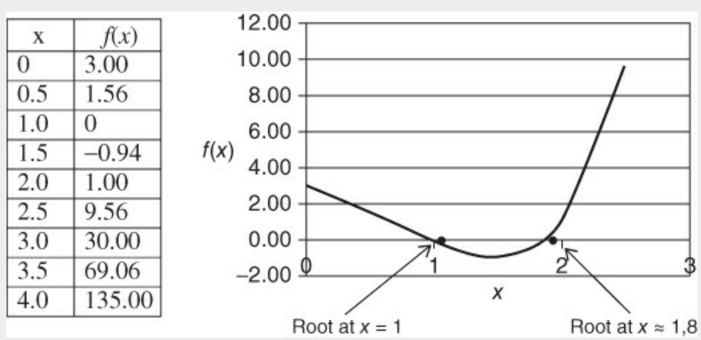

Figure 10.2 Roots of a nonlinear equation.

We note from the computed values of $f(x)$ with variable x in Figure 10.2 that there are two roots of the equation, one in the range of ($x = 1.0$ and 1.5) and the other in the range of ($x = 1.5$ and 2.0), because the sign of the function $f(x)$ changes across each these ranges of variable x. The first root of $x = 1$ is obvious because it results in $f(x) = 0$. The search for the second root by computation of the function $f(x)$ with smaller increment of x between $x = 1.5$ and $x = 2.0$ indicates a root at approximately $x = 1.8$, as illustrated in the graphic representation of the results in Figure 10.2.

10.3.2 The Newton–Raphson Method

Perhaps the most widely used method for finding the roots of nonlinear equations is the Newton–Raphson method. This method offers rapid convergence to the roots of many nonlinear equations from the initially estimated roots. The fast convergence to true roots from the estimated roots is achieved by means of computation of both the function $f(x_i)$ and the corresponding slope $f'(x_i)$ of the function at x_i, as illustrated in Figure 10.3.

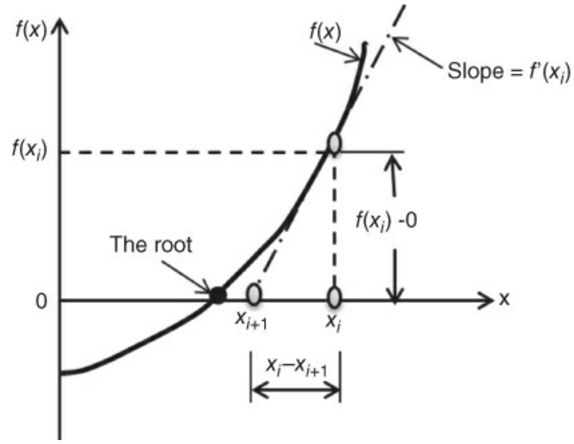

Figure 10.3 Newton–Raphson method for solving nonlinear equations.

Figure 10.3 illustrates the principle of the Newton–Raphson method of solving nonlinear equations. As in many other numerical solution methods, the user has to estimate a root at $x = x_i$ for the equation $f(x) = 0$, from which they may compute the function $f(x_i)$ and at the same time the slope of the curve generated by the function $f(x)$. This slope may be expressed $f'(x_i)$. The graphical representation of this situation indicates that the slope $f'(x_i)$ may be expressed by Equation 10.4:

$$f'(x_i) = \frac{f(x_i) - 0}{x_i - x_{i+1}} \tag{10.4}$$

which leads to the following expression for the next estimated root at $x = x_{i+1}$:

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)} \tag{10.5}$$

One readily sees from Figure 10.3 that the computed approximate next root x_{i+1} is much closer to the real root (shown as a solid circle) than the previous estimated value at x_i.

Example 10.2

Solve the nonlinear polynomial Equation 10.3 in Example 10.1 using the Newton–Raphson method.

Solution:

In this example, we are required to find the roots of Equation 10.3, with the function

$$f(x) = x^4 - 2x^3 + x^2 - 3x + 3 \tag{a}$$

From this we may express the first-order derivative that represent the slope of the curve generated by the function $f(x)$ as

$$f'(x) = 4x^3 - 6x^2 + 2x - 3 \tag{b}$$

Substituting $f(x_i)$ and $f'(x_i)$ into Equation 10.5 for the Newton–Raphson method, we will have the following expression for finding the estimated roots beginning with $x = x_i$:

$$\begin{aligned} x_{i+1} &= x_i - \frac{f(x_i)}{f'(x_i)} \\ &= x_i - \frac{x_i^4 - 2x_i^3 + x_i^2 - 3x_i + 3}{4x_i^3 - 6x_i^2 + 2x_i - 3} \end{aligned} \tag{c}$$

Thus, estimating the first root at $x = x_1 = 0.5$ for $i = 1$, we will have the next estimated x value at x_2 with $i = i + 1$ from Equation (c) as

$$\begin{aligned} x_2 &= 0.5 - \frac{(0.5)^4 - 2(0.5)^3 + (0.5)^2 - 3(0.5) + 3}{4(0.5)^3 - 6(0.5)^2 + 2(0.5) - 3} \\ &= 1.02083 \end{aligned}$$

Iterating the same procedure, we will obtain convergence of the x-values to the root of the equation. The following tabulation shows the attempts made to find the first two roots of Equation (a).

Roots of a polynomial equation of order 4

Attempt number (i)	x_i	Computed x_{i+1}	Note
1	0.5	1.0208	Estimate of first root
2	1.0208	0.9998	
3	0.9998	1.0010	Converges to first root
4	4.0	3.1818	Estimate of second root
5	3.1818	2.5990	
6	2.4655	2.1247	
7	2.1247	1.9382	
8	1.9382	1.8723	Begins to converge to root
9	1.8723	1.8638	
10	1.8638	1.8637	Converges to second root

We note from the above tabulation that it took only three attempts to find the convergence to the first root at $x = 1.0$. However, it took six attempts to reach convergence to the second root at $x = 1.8637$ with an initial estimate of the root at $x = 4$. These two roots are indicated in Figure 10.2

with the plot of the polynomial function.

Example 10.3

Locate the line mark (L) for a contents volume of 500 milliliters (ml) in a measuring cup with the dimensions shown in Figure 10.4.

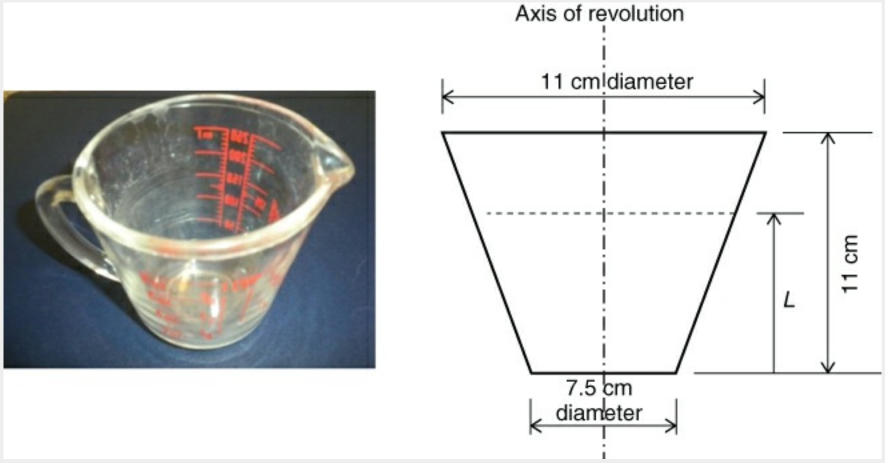

Figure 10.4 Dimensions of a measuring cup.

Solution:

We let the mark on the measuring cup for a contents volume of 500 ml be situated at L as shown in Figure 10.4. We let the volume of the measuring cup with the content at the height L to be 500 ml.

The volume of a solid of revolution of a given height may be determined from Equations (2.16) or 2.17 in Chapter 2. The profile of the measuring cup in Figure 10.4 is represented by a function $x(y) = 0.16y + 3.75$ in an x–y coordinate system in Figure 10.5.

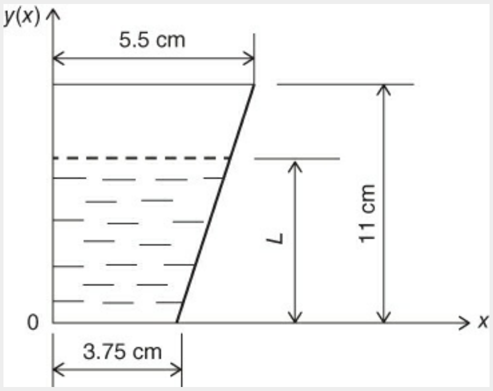

Figure 10.5 Profile of a measuring cup in the x–y coordinate system.

The volume of the contents of the measuring cup at height L may be determined by the following integral following Equation (2.17):

$$V = \pi \int_0^L [x(y)]^2 dy = \pi \int_0^L (0.16y + 3.75)^2 dy$$
$$= 0.0268L^3 + 1.884L^2 + 44.15L$$

Since the volume of the measuring cup at height L is 500 ml or 500 cm^3, we have the following equation for the unknown quantity L:

$$500 = 0.0268L^3 + 1.884L^2 + 44.15L$$

or in alternative form:

$$L^3 + 70.3L^2 + 1647.39L - 18656.72 = 0 \quad \textbf{a}$$

We recognize that Equation (a) is a cubic equation, and one of the roots of this equation will be the length L.

We will use the Newton–Raphson method to solve the cubic equation in Equation (a) with the formula given in Equation 10.5 by letting $L = x$ in Equation (a). We thus have the following expressions for the Newton–Raphson method:

$$f(x) = x^3 + 70.3x^2 + 1647.39x - 18656.72 \quad \textbf{b}$$

and its derivative:

$$f'(x) = 3x^2 + 140.6x + 1647.39 \quad \textbf{c}$$

We estimate the root of Equation (a) to be $L = x_1 = 4.0$ with $i = 1$, and using Equation 10.5 make the next estimate of the root as

$$\begin{aligned} x_2 = x_1 - \frac{f(x_1)}{f'(x_1)} &= 4 - \frac{f(4)}{f'(4)} \\ &= 4 - \frac{(-10878.36)}{2257.79} \\ &= 8.818 \end{aligned}$$

We find the subsequent estimations of the roots of Equation (c) to be rapidly converging:

$$x_3 = 8.818 - \frac{2022}{3120.47} = 8.17$$

$$x_4 = 8.17 - \frac{56.5825}{2996.3287} = 8.15$$

The last estimated root is $x_4 = 8.15$ which is close to the solution $x = 8.1566$ obtained from the online software Wolfram/Alpha Widgets (http://www.wolframalpha.com/widgets/view.jsp?id).

We may thus conclude that the mark line for 500 ml for the measuring cup in Figure 10.4 is located at the height $L = 8.15$ cm from the bottom of the cup.

Example 10.4

In Example 8.9 in Chapter 8, we derived an equation, Equation (a), to describe a mass that is attached to a spring that would break when its elongation reached 0.03 m during resonant vibration of the spring–mass system. We wish to determine the time t_f at which the spring breaks.
The equation derived in Example 8.9 for the breaking time t_f was

$$0.03 = 0.05\cos 10t_f + \frac{1}{200}\sin 10t_f - \frac{t_f}{20}\cos 10t_f$$

or

$$\left(0.05 - \frac{t_f}{20}\right)\cos 10t_f + \frac{1}{200}\sin 10t_f - 0.03 = 0 \qquad \textbf{a}$$

Solution:

We will use the Newton–Raphson method to solve for the unknown quantity t_f in Equation (a) by first assuming a solution of t_f = 0.75. We made this assumption of the solution based on a crudely approximated value of t_f = 0.7 in Example 8.9.

In order to illustrate how the Newton–Raphson method in Equation 10.5 is used to arrive at a convergent solution, we will replace the unknown quantity t_f in Equation (a) by x in the following form:

$$\left(0.05 - \frac{x}{20}\right)\cos 10x + \frac{1}{200}\sin 10x - 0.03 = 0 \qquad \textbf{b}$$

with

$$f(x) = \left(0.05 - \frac{x}{20}\right)\cos 10x + \frac{1}{200}\sin 10x - 0.03 \qquad \textbf{c}$$

and the derivative

$$f'(x) = -(0.05 - 0.5x)\sin 10x \qquad \textbf{d}$$

Thus, the estimated root x_{i+1} obtained after the previously estimated root x_i may be computed by using the expression in Equation 10.5.

We will begin the solution process with our initial estimate of the root of Equation (b) as x_1 = 0.75, which leads to the following subsequent approximate root x_2:

$$\begin{aligned} x_2 &= x_1 - \frac{f(x_1)}{f'(x_1)} = 0.75 - \frac{f(0.75)}{f'(0.75)} \\ &= 0.75 - \frac{[(0.05 - 0.75/20)\cos(10\times 0.75) + \sin(10\times 0.75/200 - 0.03]}{[-(0.05 - 0.5\times 0.75)\sin(10\times 0.75)]} \\ &= 0.869979 \end{aligned}$$

The result of this computation, x_2 = 0.869 979 is presented as trial no. 1 in the following table. We make similar trials with assumed solution x_i and with a coarse increment of 0.5 as shown in the same table using the Microsoft Excel software.

We note that the value of the function $f(x)$ in the table change from −0.003 432 with x_i = 0.85 in trial no. 3 to +0.008 5058 with x_i = 0.9 in trial no. 4. This change of the sign of the value of the function indicates that the first root of Equation (b) is between x = 0.85 and 0.90. Indeed, our subsequent trials with x_i values assigned within this range of solution using the Newton–Raphson method does indicate convergence to a more precise solution of x = 0.861 933 as indicated in the last trial, no. 8.

Trial no.	Assigned x_i	$f(x)$	$f'(x)$	x_{i+1}	% Difference
1	0.75	−0.036 576	0.304 86	0.869 979	16
2	0.80	−0.019 961	0.346 275	0.857 644	7.21
3	0.85	−0.003 432	0.299 433	0.861 46	1.35
4	0.90	0.008 5058	0.164 847	0.848 402	−5.733 12
5	0.855	−0.001 959	0.289 694	0.861 762	0.790 885
6	0.857	−0.001 384	0.285 55	0.861 846	0.565 415
7	0.86	0.005 367	0.279 071	0.861 923	0.223 604
8	0.863	0.000 2905	0.272 28	0.861 933	−0.123 61

10.4 Numerical Integration Methods

Integration of functions over specific intervals of the variables that define the functions is a frequent requirement in engineering analysis. Some of the practical applications of integration are presented in Section 2.3 in Chapter 2. Exact evaluation of many definite integrals can be found in handbooks (Zwillinger, 2003) but many others with functions to be integrated are so complicated that analytical solutions for these integrals are not possible. Numerical methods are the only viable ways for such evaluations.

In this section, we will present three numerical integration methods: (1) the trapezoidal rule; (2) Simpson's one-third rule; and (3) Gaussian quadrature. The first two methods are commonly used for integration of nonlinear functions, and the third method is extensively used in numerical analysis of complex engineering analyses, such as the finite-element analysis.

We will focus our effort on refreshing the principles that are relevant to the development of algorithms of these particular numerical integration methods, but will not rehearse the underlying theories and their proofs. The reader will find derivation of the formulae for these numerical integration methods in reference books such as those by Chapra (2012), Ferziger (1998), and Hoffman (1992).

10.4.1 The Trapezoidal Rule for Numerical Integration

We learned in Section 2.2.6 that the value of a definite integral of a function $y(x)$ is equal to the area under the curve produced by this function between the upper and lower limits of the integration as illustrated in Figure 10.6. Mathematically, the integral of function $y(x)$ can be expressed as

$$I = \int_{x_a}^{x_b} y(x)\,dx = \text{Area } A \tag{10.6}$$

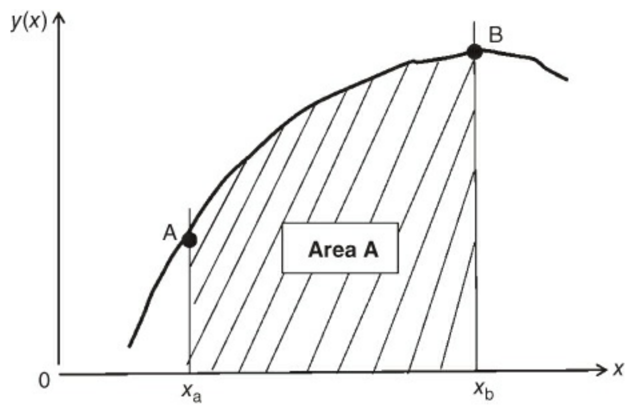

Figure 10.6 Graphical representation of integration of a continuous function.

The value of the integral of a function may thus be determined by computing the area covered by the function between the two specified limits. For example, the value of the function $y(x)$ in Figure 10.7 may be approximated by the sum of the three trapezoidal plane areas A_1, A_2, and A_3.

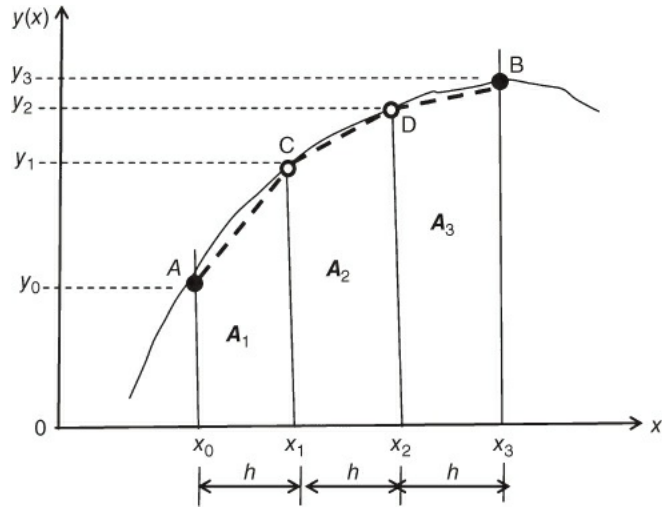

Figure 10.7 Approximation of the integral of a continuous function $y(x)$.

The area of a trapezoidal plane may be evaluated by the formula of half of the sum of the length of two parallel sides multiplied by the distance between these two sides. Mathematically, the plane areas A_1, A_2, and A_3 with equal distance h between two parallel sides in Figure 10.7 may be computed by the following formulae:

$$A_1 = \left(\frac{y_0 + y_1}{2}\right) h \qquad A_2 = \left(\frac{y_1 + y_2}{2}\right) h \qquad A_3 = \left(\frac{y_2 + y_3}{2}\right) h$$

The sum of A_1, A_2,and A_3 is given by

$$\begin{aligned} A_1 + A_2 + A_3 \approx \int_{x_0}^{x_3} y(x)\,dx &\approx \left(\frac{y_0 + y_1}{2}\right) h + \left(\frac{y_1 + y_2}{2}\right) h + \left(\frac{y_2 + y_3}{2}\right) h \\ &= \frac{h}{2}(y_0 + 2y_1 + 2y_2 + y_3) \end{aligned} \tag{10.7}$$

in which h is the assigned increment of variable x, and y_0, y_1, y_2, and y_3 are the values of the function evaluated at x_0, x_1, x_2, and x_3, respectively.

Example 10.5

Use the trapezoidal rule to evaluate the integral $\int_{x_a}^{x_b} y(x)\,dx$ in which the function $y(x) = x\sqrt{(16 - x^2)^3}$ with x_a = 0.5 and x_b = 3.5 and with assigned increment h = 1.0

Solution:

We will demonstrate the use of trapezoidal rule for numerical integration by plotting the function $y(x)$ versus x as shown in Figure 10.8. The value of the integral that we need to determine is

$$I = \int_{0.5}^{3.5} x\sqrt{(16 - x^2)^3}\,dx$$

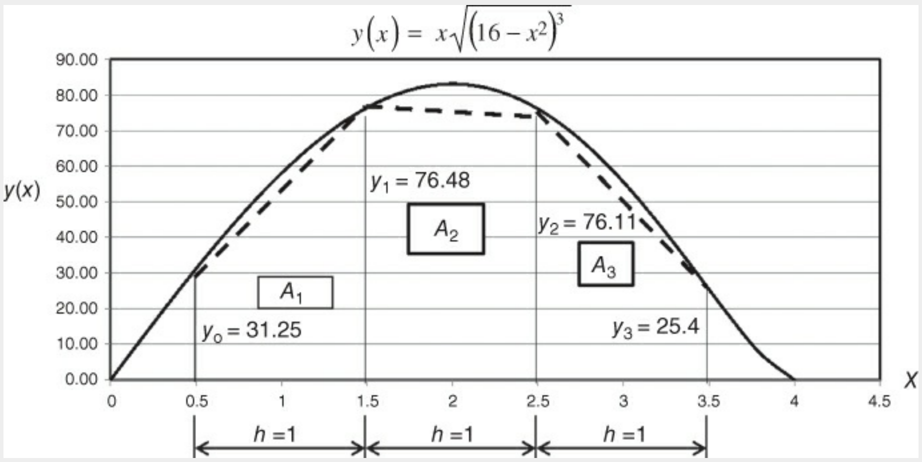

Figure 10.8 Numerical integration of a function $y(x)$ by three trapezoids.

By using the trapezoidal rule with the three trapezoids illustrated in Figure 10.8, we may evaluate the integral I by using the expression in Equation 10.7 as

$$\begin{aligned} I = \int_{0.5}^{3.5} x\sqrt{(16 - x^2)^3}\,dx &\approx \frac{h}{2}(y_0 + 2y_1 + 2y_2 + y_3) \\ &= \frac{1}{2}(31.25 + 2\times 76.48 + 2\times 76.11 + 25.4) \\ &= 180.92 \end{aligned}$$

This approximate value of the integral I obtained by the above formula is less than the analytical solution of 191.45 obtained from a mathematical handbook (Zwillinger, 2003). The difference between the value of the integral of 180.92 obtained using the trapezoidal rule and that of 191.45 from the analytical solution method is to be expected. This difference in results represents the errors inherent in any numerical method. In the trapezoidal rule method in this example, the discrepancy is introduced by the approximation of the curve representing the given function by straight line segments (shown dashed) that were the edges of the trapezoids in computing the approximate area under the curve in Figure 10.8. One may readily observe that this discrepancy between the curved edges and the straight edges shown as dashed lines can be reduced by the reduction of the size of the increment h, as can be observed from the graphic illustration in Figure 10.8. Consequently, closer approximation, or more accurate results from numerical integration, may be achieved by reducing the size of increment h in this particular numerical analysis.

Figure 10.9 shows the area under function $y(x)$, which can be approximated as the sum of the areas of (n –

1) trapezoids. Many more trapezoids under the function curve such as shown in Figure 10.9 entails many more increments of h along the x-coordinate between the upper and lower integration limits, with concomitant more accurate results.

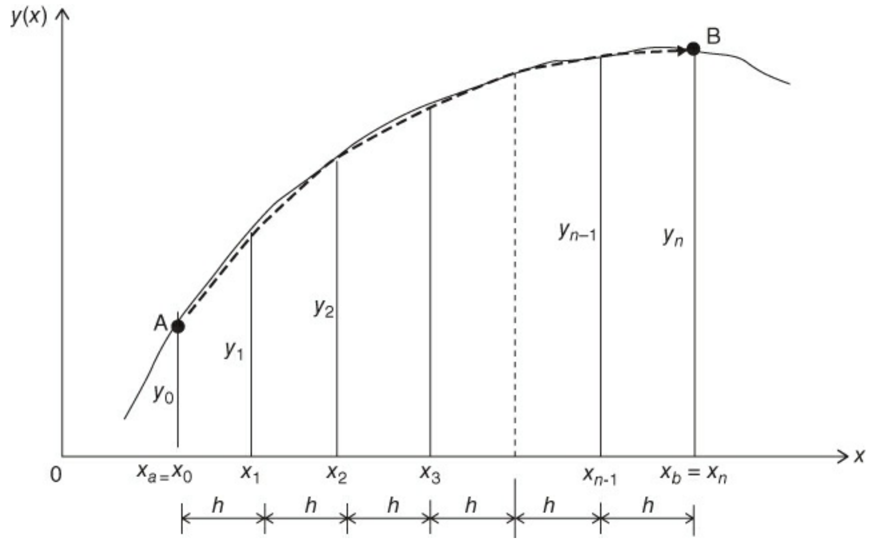

Figure 10.9 Integration of function $y(x)$ with multiple trapezoids.

The approximate value of the integral of function $y(x)$ in Figure 10.9 is equal to the sum of all the trapezoids in the figure according to the following equation derived from the same principle as in the earlier case with three trapezoids:

$$I = \int_{x_a}^{x_b} y(x)\,dx \approx \sum (A_1 + A_2 + A_3 + \cdots + A_{n-1})$$
$$= \frac{h}{2}(y_0 + 2y_1 + 2y_2 + 2y_3 + \cdots + 2y_{n-1} + y_n) \quad \textbf{10.8}$$

where h is the increment along the x-coordinate axis in the numerical integration.

Example 10.6

Evaluate the same integral as in Example 10.5 but with a reduced size of the increment h = 0.5.

Solution:

The value of the integral that we need to determine is

$$I = \int_{0.5}^{3.5} y(x)\,dx = \int_{0.5}^{3.5} x\sqrt{(16 - x^2)^3}\,dx$$

with the function $y(x) = x\sqrt{(16 - x^2)^3}$. The area covered by the function between x = 0.5 and x = 3.5 is approximated by what is shown in Figure 10.10.

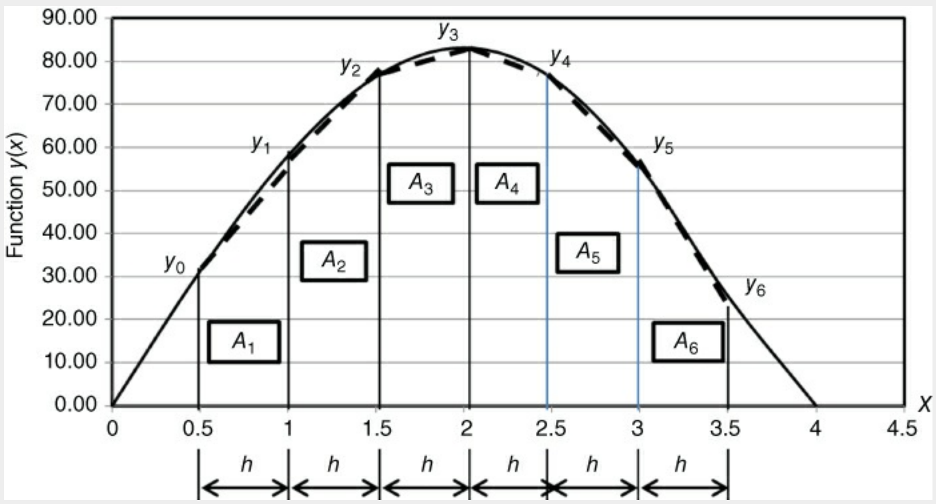

Figure 10.10 Integration of a function $y(x)$ with six trapezoids.

We will first determine the function values $y(x)$ at x = 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, and 3.5. Table 10.1 shows the function values, y_1, y_2, y_3, y_4, y_5, and y_6 required in the computation of the areas of the five trapezoids in Figure 10.10.

Table 10.1 Function values at designated points.

n	x_n	y_{n-1}
1	0.5	31.25
2	1.0	58.09
3	1.5	76.48
4	2.0	83.18
5	2.5	76.11
6	3.0	55.56
7	3.5	25.42

We may thus use Equation 10.8 to compute the sum of the areas of the five trapezoids in Figure 10.10 as

$$I \approx \frac{0.5}{2}(31.25 + 2 \times 58.09 + 2 \times 76.48 + 2 \times 83.18 + 2 \times 76.11 + 2 \times 55.56 + 25.42)$$
$$= 188.88$$

This value of 188.88 for the same integral now with h = 0.5 is much closer to the analytical value of 191.45 than that of 180 obtained with h = 1.0 in Example 10.5. It has thus been demonstrated the fact that the smaller the increment one uses in numerical integration, the more accurate a result will be obtained.

10.4.2 Numerical Integration by Simpson's One-third Rule

In Section 10.4.1, we refreshed the principle of evaluating an integral as the plane area under the curve representing the function (the integrand) in the integral between two limits of the variable in the integration. This area is graphically expressed in Figure 10.11a.

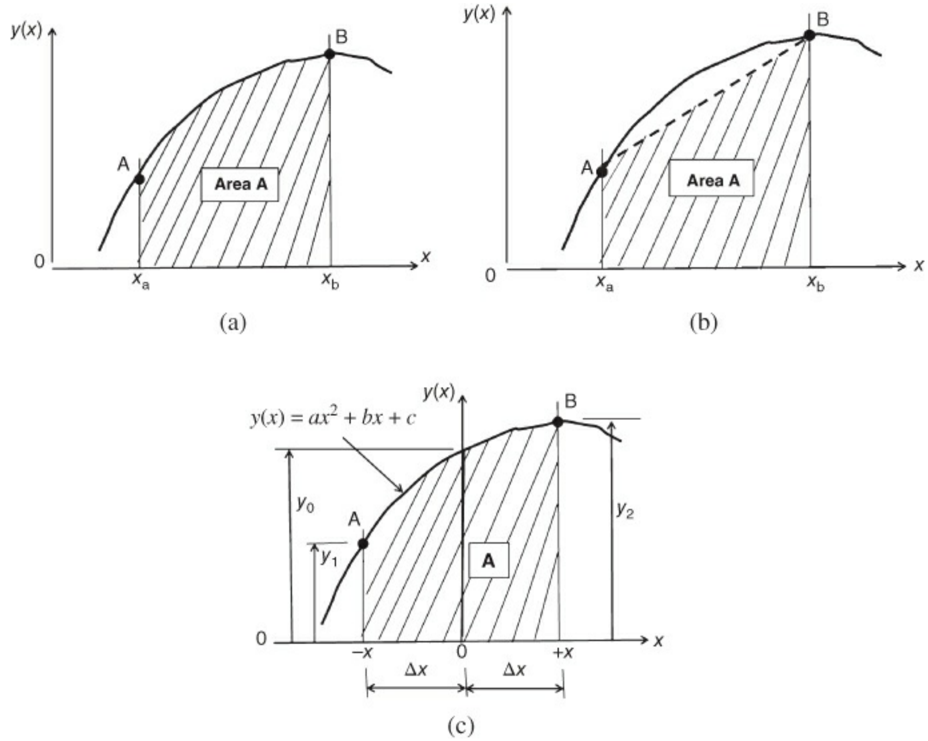

Figure 10.11 Graphical representation of integration of a continuous function. (a) Area defined by a function. (b) Approximation of the area by a trapezoid. (c) Approximation of the area by a parabolic function.

The value of an integral may be obtained by numerical methods such as the trapezoidal rule described in Section 10.4.1. The trapezoidal method is a simple but a relatively "crude" method that will result in an approximate value of the integral with minimal computational effort. Graphical representation of the trapezoidal method is illustrated in Figure 10.11b for the simplest possible numerical approximation of the area under the curve of $y(x)$ as the area of one trapezoid (shown as the cross-hatched area) under the curve. This method is popular and easy to use because the formula used to compute the plane area of trapezoids is well known to engineers.

Another popular numerical method for integration is by Simpson's rule, in particular, "Simpson's one-third rule." This rule differs from the trapezoidal method by assuming that the function $y(x)$ can be approximated in the range of interest by a parabolic function as shown in Figure 10.11c. The function $y(x) = ax^2 + bx + c$ that describes the sector of the function in Figure 10.11c contains the unknown constant coefficients a, b, and c which can be determined with the following simultaneous equations relating to the function values at the discrete variable values $x = -x$, $x = 0$, and $x = +x$ as follows:

$$y_0 = a(-x)^2 + b(-x) + c$$
$$y_1 = c$$
$$y_2 = a(x)^2 + b(x) + c$$

from which we may solve for

$$a = \frac{1}{2x^2}(y_0 - 2y_1 + y_2) \tag{10.9a}$$

$$c = y_1 \tag{10.9b}$$

The value of the integral of the function $y(x)$ is equal to the plane area A in Figure 10.11c, or

$$\begin{aligned} I = \text{AREA}(-xABx) &= \int_{-\Delta x}^{+\Delta x} y(x)\,dx = \int_{-\Delta x}^{+\Delta x} (ax^2 + bx + c)\,dx \\ &= \left(\frac{ax^3}{3} + \frac{bx^2}{2} + cx\right)\Bigg|_{-\Delta x}^{+\Delta x} \\ &= \frac{\Delta x}{3}[2a(\Delta x)^2 + 6c] \end{aligned}$$

By substituting the constant coefficients a and c in Equations 10.9a and 10.9b, together with $\Delta x = x$ illustrated in Figure 10.11c, into the above expression, we get the following relation for Simpson's one-third rule for the integral I:

$$I = \int_{-\Delta x}^{+\Delta x} y(x)\,dx = \int_{-x}^{+x} (ax^2 + bx + c)\,dx = \frac{\Delta x}{3}(y_0 + 4y_1 + y_2) \tag{10.10}$$

Example 10.7

Use Simpson's one-third rule to find the numerical value of the integral in Example 10.5.

Solution:

We will use the three function values at $x = 0.5$, 2.0, and 3.5 to compute the value of the integral of the function in Example 10.5. In this case, the increment of the integration variable is $\Delta x = 1.5$. The integral is determined using Equation 10.10 for the graphic representation of the function $y(x)$ in Figure 10.12.

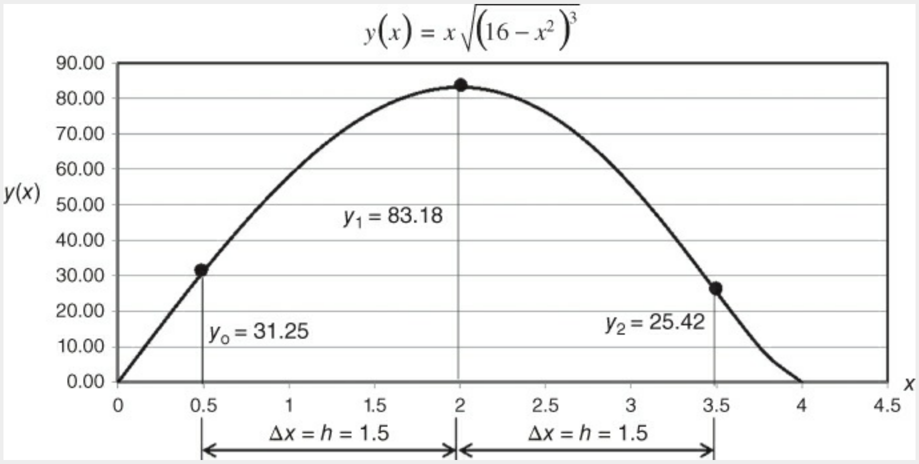

Figure 10.12 Numerical integration by Simpson's one-third rule.

We may obtain the function values y_0, y_1,and y_2 at $x = 0.5$, 2, and 3.5 from Table 10.1 as $y_0 = 31.25$, $y_1 = 83.18$, and $y_2 = 25.42$. Integration of the function $y(x)$ in Example 10.5 can thus be carried out by substituting the values of y_0, y_1 and y_2 and the increment of x, $\Delta x = 1.5$, into Equation 10.10 to give

$$
\begin{aligned}
I &= \int_{-\Delta x}^{+\Delta x} y(x)\,dx = \int_{-x}^{+x} (ax^2 + bx + c)\,dx \\
&= \frac{\Delta x}{3}(y_0 + 4y_1 + y_2) \\
&= \int_{0.5}^{3.5} y(x)\,dx \\
&= \int_{0.5}^{3.5} x\sqrt{(16 - x^2)^3}\,dx \\
&= \frac{1.5}{3}(31.25 + 4 \times 83.18 + 25.42) = 194.70
\end{aligned}
$$

The analytical or "exact" result of the above integral is 191.45 from a table of definite integrals (Zwillinger 2003), which we may compare with the results $I = 188.88$ using three trapezoids in Example 10.5 and I = 194.70 with three function values using Simpson's one-third rule.

We will use the same illustration in Figure 10.13 and Equation 10.10 to derive the general expression for Simpson's one-third rule for numerical integration.

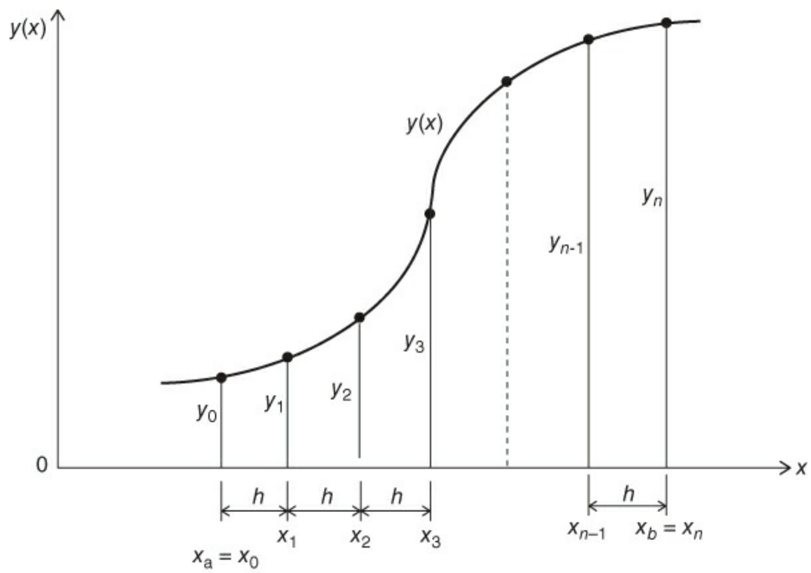

Figure 10.13 Integration of a nonlinear function $y(x)$ by Simpson's one-third rule.

We derived Equation 10.10 with the first three function values y_0, y_1 and y_2 at $x = x_0$, $x = x_1$, and $x = x_2$, respectively, with an assumed parabolic function connecting these three points. The next parabolic function for the next adjoining three-point segment requires that the function value y_2 be evaluated twice. The same happens to the second last point at x_{n-1} in the region for the integration at which the function value y_{n-1} in Figure 10.13 being evaluated twice. The coefficients associated with the y_i for n points in Figure 10.13 are given in the following tabulation:

x_i	0	1	2	3	4	5	6	…	$n-2$	$n-1$	n
Coefficient of y_i	1	4	2	4	2	4	2	…	2	4	1

We may thus formulate the general expression of Simpson's one-third rule as follows:

$$
\begin{aligned}
I &= \int_{x_a}^{x_b} y(x)\,dx \\
&= \frac{h}{3}(y_0 + 4y_1 + 2y_2 + 4y_3 + 2y_4 + \cdots + 2y_{n-2} + 4y_{n-1} + y_n)
\end{aligned}
\qquad \textbf{10.11}
$$

We note from Equation 10.11 that the use of this relationship for Simpson's one-third rule requires even number of function values with odd number increments in the integration variable.

Example 10.8

Use Simpson's one-third rule in Equation 10.10 to evaluate the integral in Example 10.6:

$$I = \int_{0.5}^{3.5} y(x)\,dx = \int_{0.5}^{3.5} x\sqrt{(16-x^2)^3}\,dx$$

Solution:

The trapezoidal method was used in numerical evaluation of the integral with seven function values as indicated in Table 10.1 in Example 10.6. Here we will require an even number of function values for Simpson's one-third rule as indicated in Equation 10.11. Consequently, we need to reduce the increment of x and increase the number of increments from seven in Example 10.6 to eight for the present example, with a slight reduction of increment from Δx = 0.5 in Example 10.6 to Δx = 0.43 in the present case. The eight function values are presented in Table 10.2 for the data points shown in Figure 10.14.

Table 10.2 Eight values of a function for integration using Simpson one-third rule.

n	1	2	3	4	5	6	7	8
x_n	0.5	0.93	1.36	1.79	2.21	2.64	3.07	3.5
$y = y(x_n)$	31.25	54.69	72.3	81.89	81.85	71.54	51.68	25.41

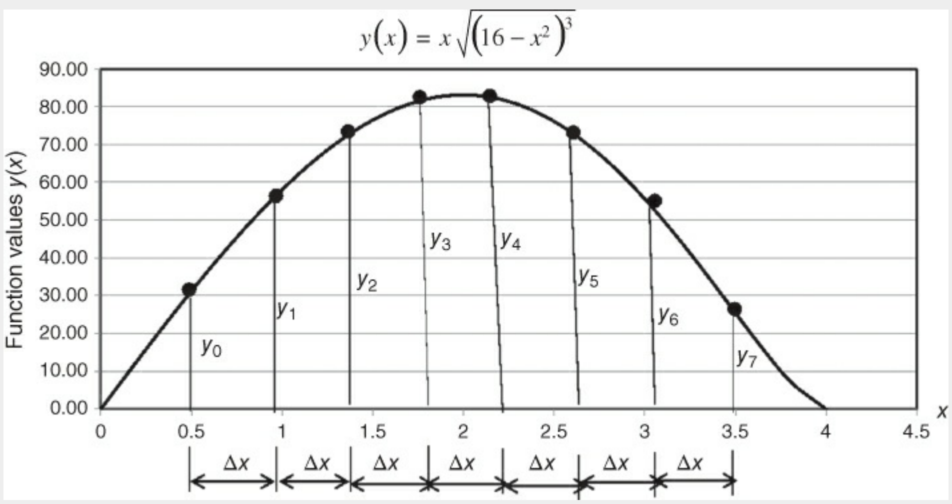

Figure 10.14 Numerical integration using Simpson's one-third rule with eight function values.

We may thus insert the function values presented in Table 10.2 into Equation 10.11 and evaluate the integral of the function:

$$\begin{aligned}
I &= \int_{0.5}^{3.5} y(x)\,dx = \int_{0.5}^{3.5} x\sqrt{(16-x^2)^3}\,dx \\
&\approx \frac{\Delta x}{3}(y_0 + 4y_1 + 2y_2 + 4y_3 + 2y_4 + 4y_5 + 2y_6 + y_7) \\
&= \frac{0.43}{3}(31.25 + 4\times 54.69 + 2\times 72.30 + 4\times 81.89 + 2\times 81.85 + 4\times 71.54 \\
&\quad + 2\times 51.68 + 25.41) \\
&= 186.45
\end{aligned}$$

This approximate solution of 186.45 of the integral has about 2.6% error from the exact solution of 191.45.

10.4.3 Numerical Integration by Gaussian Quadrature

Numerical integration of functions using the trapezoidal rule (Section 10.4.1) and Simpson's one-third rule (Section 10.4.2) enables us to determine approximate values of integrals of continuous functions $f(x)$ over a range of $(x_b - x_a)$ into equal parts with increment of the variable Δx (or h) as shown in Figures 10.7 and 10.9. This process allows us to select the sampling points and evaluate the integral in terms of the discrete values of the function at these points. These methods usually work well with well-behaved functions in integrals such as the ones used in Examples 10.5 to 10.8. However, neither the trapezoidal rule nor Simpson's one-third rule offers any guidance on the selection of the size of the increment of the variable in these numerical integration methods. There are times when engineers are expected to find numerical values of integrals involving functions that have drastic changes of shape over the range of the required integration. The two methods already discussed do not yield good approximations of the numerical values of these integrals because of improper selection of sampling points. Thus, it is desirable to have a numerical integration method that offers criteria for optimal sampling points in numerical integration.

The Gaussian integration method was established on the basis of strategically selected sampling points. The normal form of a Gaussian integral can be expressed as

$$I = \int_{-1}^{1} F(\xi)\,d\xi = \sum_{i=1}^{n} H_i F(a_i) \tag{10.12}$$

in which n is the total number of sampling points, and H_i are the weighting coefficients corresponding to sampling points located at $\xi = \pm a_i$ as given in Table 10.3.

Table 10.3 Weight coefficients of the Gaussian quadrature formula in Equation 10.12 (Kreyszig, 2011; Zwillinger, 2003)

n	$\pm a_i$	H_i
2	$a_1 = 0.577\,35$	$H_1 = 1.000\,00$
	$a_2 = -0.577\,35$	$H_2 = 1.000\,00$
3	$a_1 = 0.0$	$H_1 = 0.888\,88$
	$a_2 = 0.774\,59$	$H_2 = 0.555\,55$
	$a_3 = -0.774\,59$	$H_3 = 0.555\,55$
4	$a_1 = 0.339\,98$	$H_1 = 0.652\,14$
	$a_2 = -0.339\,98$	$H_2 = 0.652\,14$
	$a_3 = 0.861\,13$	$H_3 = 0.347\,85$
	$a_4 = -0.861\,13$	$H_4 = 0.347\,85$
5	$a_1 = 0.0$	$H_1 = 0.568\,88$
	$a_2 = 0.538\,46$	$H_2 = 0.478\,62$
	$a_3 = -0.538\,46$	$H_3 = 0.478\,62$
	$a_4 = 0.906\,17$	$H_4 = 0.236\,92$
	$a_5 = -0.906\,17$	$H_5 = 0.236\,92$
6	$a_1 = 0.238\,61$	$H_1 = 0.467\,91$
	$a_2 = -0.238\,61$	$H_2 = 0.467\,91$
	$a_3 = 0.661\,20$	$H_3 = 0.360\,76$

	$a_4 = -0.661\ 20$	$H_4 = 0.360\ 76$
	$a_5 = 0.932\ 46$	$H_5 = 0.171\ 32$
	$a_6 = -0.932\ 46$	$H_6 = 0.171\ 32$
7	$a_1 = 0.0$	$H_1 = 0.417\ 95$
	$a_2 = 0.405\ 84$	$H_2 = 0.381\ 83$
	$a_3 = -0.405\ 84$	$H_3 = 0.381\ 83$
	$a_4 = 0.741\ 53$	$H_4 = 0.279\ 70$
	$a_5 = -0.741\ 53$	$H_5 = 0.279\ 70$
	$a_6 = 0.949\ 10$	$H_6 = 0.129\ 48$
	$a_7 = -0.949\ 10$	$H_7 = 0.129\ 48$
8	$a_1 = 0.183\ 43$	$H_1 = 0.362\ 68$
	$a_2 = -0.183\ 43$	$H_2 = 0.362\ 68$
	$a_3 = 0.525\ 53$	$H_3 = 0.313\ 70$
	$a_4 = -0.525\ 53$	$H_4 = 0.313\ 70$
	$a_5 = 0.796\ 66$	$H_5 = 0.222\ 38$
	$a_6 = -0.796\ 66$	$H_6 = 0.222\ 38$
	$a_7 = 0.960\ 28$	$H_7 = 0.101\ 22$
	$a_8 = -\ 0.960\ 28$	$H_8 = 0.101\ 22$

The form of Gaussian integral shown in Equation 10.12 is rarely seen in practice. A transformation of coordinate is required to convert the general form of integration such as shown in Equation 10.6 to the form shown in Equation 10.12, as illustrated in Figure 10.15.

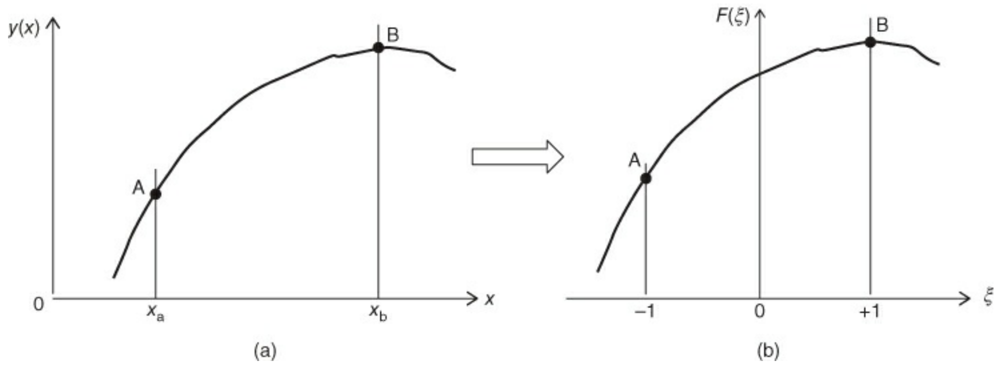

Figure 10.15 Transformation of coordinates for Gaussian integration. (a) With the original coordinates. (b) After transformation of coordinates.

The transformation of coordinates from $y(x)$ in the x-coordinate to the function $F(\xi)$ in the coordinate ξ may be accomplished using the relationship

$$x = \frac{1}{2}(x_b - x_a)\xi + \frac{1}{2}(x_b + x_a) \tag{10.13}$$

which leads to the following expression:

$$\int_{x_a}^{x_b} y(x)\,dx = \frac{x_b - x_a}{2}\int_{-1}^{1} F(\xi)\,d\xi \tag{10.14}$$

with

$$F(\xi) = y(x)|_{x=\frac{1}{2}(x_b - x_a)\xi + \frac{1}{2}(x_b - x_a)}$$

and

$$d\xi = \left[\frac{1}{2}(x_b - x_a)\right]^{-1} dx$$

We will obtain the expression for the required evaluation of the integral in Equation 10.6 using Gaussian quadrature by substituting the relationship in Equation 10.12 into Equation 10.14.

$$I = \int_{x_a}^{x_b} y(x)\, dx = \frac{x_b - x_a}{2} \sum_{i=1}^{n} H_i F(a_i) \qquad \textbf{10.15}$$

Numerical values of the weight coefficients H_i with sampling points a_i are given in Table 10.3.

One needs to realize that the term $F(a_i)$ on the right-hand-side of Equation 10.15 denotes the function $y(x)$ evaluated at the sampling points a_i after it has been transformed to the function $F(\xi)$ with integration limits of (−1 to +1) as in Figure 10.15b.

Example 10.9

Evaluate the following integral using Gaussian quadrature as shown in Equation 10.14:

$$I = \int_0^{\pi} \cos x \, dx \tag{a}$$

Solution:

We have the function $y(x) = \cos x$ over the integration limits $x_a = 0$ and $x_b = \pi$. The transformation of coordinates makes use of the relationship $x = (\pi/2)\xi + \pi/2$ from Equation 10.13, from which we obtain $dx = (\pi/2)\, d\xi$, and the functions are related as

$$\begin{aligned} y(x) &= \cos x = F(\xi) \\ &= \cos\left(\frac{\pi}{2}\xi + \frac{\pi}{2}\right) \\ &= \sin\left(\frac{\pi}{2}\xi\right) \end{aligned}$$

from Equation 10.14 with the use of the trigonometric relationships

$$\sin\left(\frac{\pi}{2} + \theta\right) = \cos\theta \quad \text{and} \quad \cos\left(\frac{\pi}{2} + \theta\right) = \sin\theta$$

We thus arrive at the following expression for integral I in Equation (a) using Gaussian quadrature:

$$\begin{aligned} I = \int_0^{\pi} \cos x \, dx &= \int_{-1}^{1} \left[\sin\left(\frac{\pi}{2}\xi\right)\left(\frac{\pi}{2} d\xi\right)\right] \\ &= \frac{\pi}{2}\int_{-1}^{1} \sin\frac{\pi}{2}\xi \, d\xi \\ &= \frac{\pi}{2}\sum_{i=1}^{n} H_i \sin\left(\frac{\pi}{2}a_i\right) \end{aligned} \tag{b}$$

Let us take, for example, three sampling points (i.e., $n = 3$ from Table 10.3):

$$\begin{array}{lll} a_1 = 0 & a_2 = +0.77459 & a_3 = -0.77459 \\ H_1 = 0.88888 & H_2 = 0.55555 & H_3 = 0.55555 \end{array}$$

Substituting the above values into Equation (b) gives

$$\begin{aligned} I &= \frac{\pi}{2}\left[0.88888 \sin(0) + 0.55555 \sin\left(\frac{\pi}{2} \times 0.77459\right) + 0.55555 \sin\left(-\frac{\pi}{2} \times 0.77459\right)\right] \\ &= \frac{\pi}{2}[0.55555 \sin(1.2167) - 0.55555 \sin(1.2167)] = 0 \end{aligned} \tag{c}$$

Example 10.10

Evaluate the following integral using the Gaussian quadrature as in Equation 10.15:

$$I = 2\int_0^{\pi} \sin x dx \quad \mathbf{a}$$

Solution:

The integral in Equation (a) is similar to that in Example 10.9 but with $y(x) = \sin x$. We may thus use a similar derivation to obtain the value of the integral in Equation (a) as

$$\begin{aligned} I &= 2\int_0^{\pi} \sin x\, dx \\ &= 2\int_{-1}^{1} \left[\cos\left(\frac{\pi}{2}\xi\right)\left(\frac{\pi}{2}d\xi\right)\right] \\ &= 2\times\frac{\pi}{2}\int_{-1}^{1} \cos\frac{\pi}{2}\xi\, d\xi \\ &= \pi\sum_{i=1}^{n} H_i \cos\left(\frac{\pi}{2}a_i\right) \end{aligned} \quad \mathbf{b}$$

and the result of using the same three sampling points with $n = 3$ leads to a similar expression to that in Equation (c) in Example 10.9:

$$\begin{aligned} I = 2\times\frac{\pi}{2}&\left[0.88888\cos(0) + 0.55555\cos\left(\frac{\pi}{2}\times 0.77459\right)\right. \\ &\left. +0.55555\cos\left(-\frac{\pi}{2}\times 0.77459\right)\right] \\ &= \pi[0.88888 + 0.55555\cos(1.2167) + 0.55555\cos(-1.2167)] = 4.002497 \end{aligned}$$

The exact solution is 4.0 for the integral in Equation (a).

Example 10.11

Evaluate the following integral from Example 10.8 using the Gaussian quadrature method:

$$I = \int_{0.5}^{3.5} y(x)\,dx = \int_{0.5}^{3.5} x\sqrt{(16 - x^2)^3}\,dx \quad \textbf{a}$$

Solution:

The function $y(x)$ in the integral in Equation (a) is

$$y(x) = x\sqrt{(16 - x^2)^3} \quad \textbf{b}$$

with integration limits $x_a = 0.5$ and $x_b = 3.5$.

We will first derive the expression of the function $F(\xi)$ for the function $y(x)$ in Equation (a) from the integration limits of (0.5, 3.5) to the limit (−1, +1) as required in Gaussian quadrature.

The transformation of coordinate systems as illustrated in Figure 10.15 begins with the transformation of the variable from x to ξ using the relationship in Equation 10.13, leading to the following relationship between the variables x and ξ:

$$x = \frac{1}{2}(x_b - x_a)\xi + \frac{1}{2}(x_b + x_a) = 1.5\xi + 2$$

from which we obtain

$$dx = 1.5d\xi \quad \textbf{c}$$

The integral in Equation (a) in the $y(x)$ vs. x coordinates can thus be transformed to the $F(\xi)$ vs. ξ coordinates by using Equation 10.14, yielding the following expression:

$$\begin{aligned} I = \int_{0.5}^{3.5} y(x)\,dx &= \int_{0.5}^{3.5} x\sqrt{(16 - x^2)^3}\,dx \\ &= \frac{x_b - x_a}{2}\int_{-1}^{1} F(\xi)\,d\xi \\ &= \frac{3.5 - 0.5}{2}\int_{-1}^{1} y(x)\Big|_{x=1.5\xi+2}\,d\xi \end{aligned} \quad \textbf{d}$$

We may thus evaluate the integral from Equation (d) as

$$\begin{aligned} I &= 1.5\int_{-1}^{1}(1.5\xi + 2)\sqrt{[16 - (1.5\xi + 2)^2]^3}\,d\xi \\ &= 1.5\int_{-1}^{1}(1.5\xi + 2)\sqrt{(-2.25\xi^2 - 6\xi + 12)^3}\,d\xi \end{aligned}$$

from which we have the function

$$F(\xi) = (1.5\xi + 2)\sqrt{(-2.25\xi^2 - 6\xi + 12)^3} \quad \textbf{e}$$

We may then use Equation 10.12 for the value of the integral I in Equation (a):

$$I = 1.5\sum_{i=1}^{n} H_i F(a_i) \tag{f}$$

Let us choose three sampling points for the integral in Equation (f), i.e., $n = 3$ in Equation (e) with a_i and H_i ($i = 1, 2, 3$) from Table 10.3, as tabulated below:

i	a_i	H_i
1	0	0.888 88
2	0.774 59	0.555 55
3	−0.774 59	0.555 55

By substituting the above numbers into Equation (f), we will obtain the value of the integral I as

$$I = 1.5[H_1F(0) + H_2F(0.7746) + H_3F(-0.7746)] \tag{g}$$

We may evaluate the function values at a_i using Equation (e) as

$$F(0) = 83.1384, \quad F(0.774\,59) = 46.4981, \ F(-0.774\,59) = 50.1454 \tag{10.16}$$

Substituting these function values into Equation (f) will give the integral I in Equation (a) as

$$\begin{aligned} I = \int_{0.5}^{3.5} y(x)\,dx &= \int_{0.5}^{3.5} x\sqrt{(16-x^2)^3}\,dx \\ &= 1.5(0.8888 \times 83.1384 + 0.5555 \times 46.4981 + 0.5555 \times 50.1454) \\ &= 191.3684 \end{aligned}$$

This value of the integral in Equation (a) obtained by Gaussian quadrature with three sampling points is remarkably close to the exact value of 191.45 obtained from a table of integrals (Zwillinger, 2003).

10.5 Numerical Methods for Solving Differential Equations

Differential equations frequently appear in engineering analysis, as described in Section 2.5. Many differential equations that engineers use for their analyses are "linear equations," such as those presented in Chapters 7, 8, and 9, which can be solved by classical solution methods. There are, however, occasions in which engineers need to solve either highly complicated linear differential equations or nonlinear differential equations; in such cases numerical solution methods become viable alternative methods for finding solutions.

Numerical solution methods for differential equations relating to two types of engineering analysis problems: (1) "initial value" problems, and (2) "boundary value" problems. Solution of initial value problems involves a starting point with the variable of the function, say at x_0, that is a specific value of variable x for solution $y(x)$. With the solution given at this starting point, one may find the solutions at $x = x_0+h, x_0+2h, x_0+3h, \ldots, x_0+nh$, where h is the selected "step size" in the numerical computations and n is an integer number of steps used in the analysis. The number of steps n in the computation can be as large as is required to cover the entire range of the variable in the analysis. Numerical solution to boundary value problems is more complicated; function values are often specified at a number of variable values, and the selected steps for solution values may be restricted by the specified values at these variable points.

There are numerous numerical solution methods available for the solutions of differential equations. It is not possible to cover all these solution methods in this section. What we will learn from this section is the principle of converting "differential equations" to "difference equations," followed by numerical computation of solutions at discrete times or locations of the domain in the analysis. We will also confine our coverage to selected numerical solution methods involving only initial value problems. Readers are referred to reference books that have extensive coverage of numerical solutions for differential equations of both ordinary and partial differential equations, such as Ferziger (1998) and Chapra (2012).

10.5.1 The Principle of Finite Difference

We have learned in Chapter 2 that differential equations are equations that involve derivatives. Physically, a derivative represents the rate of change of a physical quantity represented by a function with respect to the change of its variable(s).

Referring to Figure 10.16, we have a continuous function $f(x)$ that has values f_{i-1}, f_i, and f_{i+1} corresponding to the three values of its variable x at x_{i-1}, x_i, and x_{i+1}, respectively. We may also write the three function values at the three x-values as

$$f_i = f(x) \qquad \textbf{10.16a}$$

$$f_{i+1} = f(x + \Delta x) \qquad \textbf{10.16b}$$

$$f_{i-1} = f(x - \Delta x) \qquad \textbf{10.16c}$$

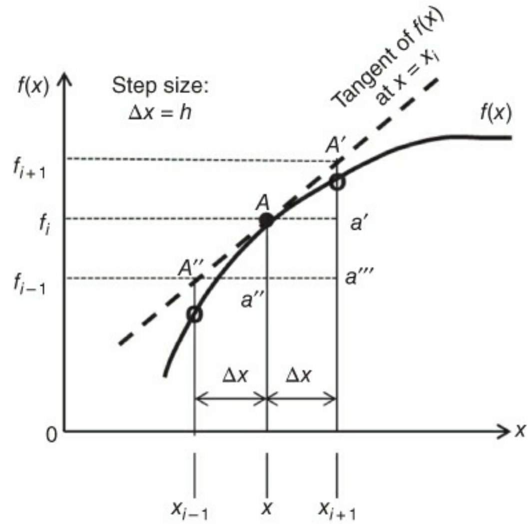

Figure 10.16 Function $f(x)$ evaluated at three positions.

The derivative of the function $f(x)$ at point A with $x = x_i$ in Figure 10.16 is graphically represented by the tangent line A″–A′ to the curve representing function $f(x)$ at point A. Mathematically, we may express the derivative as given in Equation (2.9), or in the form

$$\frac{df(x)}{dx} = \lim_{\Delta x \to 0} \frac{\Delta f}{\Delta x} \approx \frac{\Delta f}{\Delta x} \qquad \textbf{10.17}$$

where Δx is the increment used to change the values of the variable x, and dx is the increments of variable x with infinitesimally small sizes.

One may observe from Equation 10.17 the important relation that the derivative may be approximated by finite increments of Δf and Δx as indicated in Equation 10.18:

$$\frac{df(x)}{dx} \approx \frac{\Delta f}{\Delta x} \qquad \textbf{10.18}$$

if the condition on the increment $\Delta x \to 0$ is removed in the evaluation of the rate of change of the function $f(x)$ in Equation 10.17.

Thus from Equation 10.17 we see that derivatives of functions may be approximated by adopting finite—but not infinitesimally small—increments of the variable x. A formulation with such an approximation is called “finite difference”.

10.5.2 The Three Basic Finite-difference Schemes

There are three basic schemes that one may use to approximate a derivative: (1) the “forward difference” scheme; (2) the “backward difference” scheme; and (3) the “central difference” scheme. Mathematical expressions of these difference schemes are given below.

The forward difference scheme

In the “forward difference scheme,” the rate of change of the function $f(x)$ with respect to the variable x is accounted for between the function value at the current value $x = x_i$ and the value of the same function at the next step, that is, $x_{i+1} = x + \Delta x$ in the triangle ΔA′Aa′ in Figure 10.16. The mathematical expression of this scheme is given in Equation 10.19:

$$\nabla f_i = \left.\frac{df(x)}{dx}\right|_{x=x_i} \approx \left.\frac{\Delta f}{\Delta x}\right|_{x=x_i} = \frac{f_{i+1}-f_i}{x_{i+1}-x_i} = \frac{f_{i+1}-f_i}{\Delta x} = \frac{f_{i+1}-f_i}{h} \qquad \textbf{10.19}$$

in which $h = \Delta x$ is the "step size."

The derivative of the function $f(x)$ at other values of the variable x in the positive direction can be expressed following Equation 10.19 as

$$\begin{aligned} \nabla f_{i+1} &= \frac{f_{i+2}-f_{i+1}}{h} \\ \nabla f_{i+2} &= \frac{f_{i+3}-f_{i+2}}{h} \\ &\text{and so on} \end{aligned} \qquad \textbf{10.20}$$

The second-order derivative of the function $f(x)$ at x can be obtained according to the following procedure:

$$\nabla^2 f_i = \frac{d}{dx}\left.\left(\frac{df(x)}{dx}\right)\right|_{x=x_i} = \lim_{\Delta x\to 0}\frac{\nabla f_{i+1}-\nabla f_i}{\Delta x} \approx \frac{\nabla f_{i+1}-\nabla f_i}{\Delta x} = \frac{\nabla f_{i+1}-\nabla f_i}{h} = \frac{\frac{f_{i+2}-f_{i+1}}{h}-\frac{f_{i+1}-f_i}{h}}{h} = \frac{f_{i+2}-2f_{i+1}+f_i}{h^2} \qquad \textbf{10.21}$$

The backward difference scheme

In this difference scheme, the rate of the change of the function with respect to the variable x is accounted for between the current value at $x = x_i$ and the step backward, that is, $x_{i-1} = x - \Delta x$ in the triangle ΔAA″a″ in Figure 10.16. The mathematical expression of this scheme is given in Equation 10.22:

$$\nabla f_i = \lim_{\Delta x\to 0}\frac{f_i - f_{i-1}}{\Delta x} = \lim_{\Delta x\to 0}\frac{f(x)-f(x-\Delta x)}{\Delta x} \approx \frac{f_i - f_{i-1}}{\Delta x} = \frac{f_i - f_{i-1}}{h} \qquad \textbf{10.22}$$

Following a similar procedure as in the forward difference scheme, we may express the second-order derivative in the following form:

$$\nabla^2 f_i \approx \frac{f_i - 2f_{i-1} + f_{i-2}}{h^2} \qquad \textbf{10.23}$$

The central difference scheme:

The rate of change of function $f(x)$ in this finite-difference scheme includes the function values between the preceding step at $(x - \Delta x)$ and the step ahead, that is, $(x + \Delta x)$. The triangle involved in this difference scheme is ΔA′A″a. in Figure 10.16. We have the first-order derivative as in Equation 10.24:

$$\nabla f \approx \frac{f_{i+1} - f_{i-1}}{x_{i+1} - x_{i-1}} = \frac{f_{i+1} - f_{i-1}}{2h} \qquad \textbf{10.24}$$

Note that in this finite-difference scheme a much larger step of size 2*h* is used in the first-order derivative as given in Equation 10.24. These "coarse" steps will compromise the accuracy of the values of the derivatives. A better central difference scheme is to employ for "half" steps in both directions. In other words, if we define

$$f_{i+\frac{1}{2}} = f\left(x + \frac{\Delta x}{2}\right) \qquad \text{and} \qquad f_{i-\frac{1}{2}} = f\left(x - \frac{\Delta x}{2}\right) \qquad \textbf{10.25}$$

we will then have the derivative of the function $f(x)$ using this modified central difference scheme as

$$\nabla f_i \approx \frac{f_{i+\frac{1}{2}} - f_{i-\frac{1}{2}}}{h} \qquad \textbf{10.26}$$

Example 10.12

Solve the following differential equation using the finite-difference method.

$$\frac{d^2x(t)}{dt^2} + x(t) = 0 \tag{a}$$

with specified initial conditions

$$x(0) = 1 \tag{b}$$

$$\dot{x}(0) = 0 \tag{c}$$

Solution:

Let us use the *forward difference scheme* according to Equations 10.19 and 10.21 with

$$\frac{dx(t)}{dt} \approx \frac{x(t+\Delta t) - x(t)}{\Delta t} \tag{d}$$

and

$$\frac{d^2x(t)}{dt^2} \approx \frac{x(t+2\Delta t) - 2x(t+\Delta t) + x(t)}{(\Delta t)^2} \tag{e}$$

Substitution of Equation (e) into Equation (a) results in the following finite-difference form of the differential equation:

$$\frac{x(t+2\Delta t) - 2x(t+\Delta t) + x(t)}{(\Delta t)^2} + x(t) = 0$$

Upon rearranging the terms in the above equation, we get the following "recurrence relation" for the approximate solution of Equation (a):

$$x(t+2\Delta t) - 2x(t+\Delta t) + [1+(\Delta t)^2]x(t) = 0 \tag{f}$$

Applying the finite-difference operator to the initial conditions in Equations (b) and (c) gives

$$x(0) = 1 \tag{g}$$

$$x(\Delta t) = x(0) = 1 \tag{h}$$

We are now ready to solve for $x(t)$ in Equation (a) using the finite-difference operator by repeated use of the recurrence relation in Equation (f). The solution of $x(t)$ will be on the incremental steps of Δt chosen by the user. By referring to the initial conditions in Equations (g) and (h) we may obtain $x(t)$ at all subsequent steps.

Choice of solution steps, Δt:

Let us assume that a step size $\Delta t = 0.05$ is chosen for the solution. The corresponding solution becomes, from Equation (f) with $\Delta t = 0.05$:

$$x(t+0.1) - 2x(t+0.05) + 1.0025x(t) = 0 \tag{j}$$

and from Equation (h):

$$x(0.05) = 1$$

Now, at $t = 0$, we get from Equation (j):

$$x(0.1) - 2x(0.05) + 1.0025x(0) = 0$$

Because $x(0) = 1$ is the initial condition in Equation (g), the above relation yields

$$x(0 + 0.1) - 2x(0 + 0.05) = -1.0025$$

But since $x(0.05) = 1$ from Equation (h), we have

$$x(0.1) = -1.0025 + 2x(0.05) = 0.9975 \quad \textbf{k}$$

We may now move to the next time point by letting $t = t + \Delta t = 0 + 0.05 = 0.05$ and substituting $t = 0.05$ into Equation (f) to get

$$x(0.05 + 0.1) - 2x(0.05 + 0.05) + 1.0025x(0.05) = 0$$

or

$$x(0.15) - 2x(0.1) + 1.0025x(0.05) = 0 \quad \textbf{m}$$

Since we have already obtained $x(0.1) = 0.9975$ from Equation (k) and $x(0.05) = 1$ from Equation (h), we will thus have another solution point from Equation (m):

$$x(0.15) = 2 \times 0.9975 - 1.0025 \times 1 = 0.9925 \quad \textbf{p}$$

We move to the next time point, that is, $t = t + \Delta t = 0.05 + 0.05 = 0.1$. Substituting $t = 0.1$ into Equation (f) we get

$$x(0.1 + 0.1) - 2x(0.1 + 0.05) + 1.0025x(0.1) = 0$$

or

$$x(0.2) - 2x(0.15) + 1.0025x(0.1) = 0$$

But with $x(0.15) = 0.9925$ from the last step in Equation (p) and $x(0.1) = 0.9975$ from Equation (k), we have

$$x(0.2) = 2 \times 0.9925 - 1.0025 \times 0.9975 = 0.9850 \quad \textbf{q}$$

Thus, following the same procedure as illustrated above, we may obtain solution of Equation (a) at all time points with an increment $\Delta t = 0.05$.

The results obtained from the above exercise are summarized in the tabulation below, with comparison to the exact solution of $x(t) = \cos\ t$.

Variable, t	Solution by the finite-difference method	Exact solution	% Error
0	1	1	0
0.05	1	0.999 996	≈0
0.10	0.9975	0.995 0041	0.25
0.15	0.9925	0.988 77	0.38
0.20	0.9850	0.980 066	0.503

One will observe from the tabulated values that the percentage error of the results obtained from the finite-difference method increases with the increase of the variable t. *One may also show that the accuracy of the finite-difference method improves with smaller increments of the variables—the Δt in the present example*.

10.5.3 Finite-difference Formulation for Partial Derivatives

Partial derivatives along a single dimension are computed in the same fashion as for ordinary derivatives

(Chapra 2012) illustrated in Section 10.5.2. For example, the central difference scheme for function $f(x,y)$ can be shown to have the following expressions:

$$\frac{\partial f(x,y)}{\partial x} \approx \frac{f(x+\Delta x,y)-f(x-\Delta x,y)}{2\Delta x} \tag{10.27a}$$

$$\frac{\partial f(x,y)}{\partial y} \approx \frac{f(x,y+\Delta y)-f(x,y-\Delta y)}{2\Delta y} \tag{10.27b}$$

The error of the above approximation is of order $\Delta x = h$.

For higher-order partial derivatives, such as $\partial^2 f(x,y)/\partial x \partial y$:

$$\frac{\partial^2 f(x,y)}{\partial x \partial y} = \frac{\partial}{\partial x}\left[\frac{\partial f(x,y)}{\partial y}\right] \approx \frac{\dfrac{\partial f(x+\Delta x,y)}{\partial y} - \dfrac{\partial f(x-\Delta x,y)}{\partial y}}{2\Delta x} \tag{10.28}$$

Evaluating each of the partial derivatives in Equation 10.28 will lead to the following final expression:

$$\frac{\partial^2 f(x,y)}{\partial x \partial y} \approx \frac{f(x+\Delta x,y+\Delta y)-f(x+\Delta x,y-\Delta y)}{4\Delta x\,\Delta y} + \frac{-f(x-\Delta x,y+\Delta y)+f(x-\Delta x,y-\Delta y)}{4\Delta x\,\Delta y} \tag{10.29}$$

The error of the approximation of second-order differentiation is of order $(\Delta x)^2$.

10.5.4 Numerical Solution of Differential Equations

There are a number of numerical solution methods available for solving differential equations relating to both types of initial value and boundary value problems (Hoffman, 1992; Bronson, 1994; Ferziger, 1998; Chapra, 2012). We will present only the expressions used in the classic fourth-order Runge–Kutta method to illustrate the power of numerical methods for solving differential equations for initial value problems.

The simplest method of solving differential equations is to convert the derivatives in differential equations to the forms of "finite differences" as presented in Section 10.5.2 and Example 10.12. This method is straightforward but usually has significant accumulation of errors in the solution, as indicated in the numerical illustration in Example 10.12. There are several alternative versions available for solutions with better accuracies in the references cited above. We will include the Runge–Kutta method in this section for numerical solution of differential equations using initial value processes.

The Runge–Kutta methods are integrative methods for approximation of solutions of differential equations. These method, with several versions of the technique, were developed around 1900s by German mathematicians C. Runge and M.W. Kutta. The essence of the Runge–Kutta methods involves numerical integration of the function in a differential equation by using a trial step at mid-point of an interval—within a step Δx or h—using numerical integration techniques such as the trapezoidal or Simpson's rules as presented in Section 10.4. The numerical integrations will allow the cancellation of low-order error terms for more accurate solutions. Several versions of Runge–Kutta methods with different orders for solving differential equations have been developed over the years.

10.5.4.1 The Second-order Runge–Kutta Method

This is the simplest form of the Runge–Kutta method, with the formulation for the solution of first-order differential equation in the following form:

$$y'(x) = f(x,y) \tag{10.30}$$

with a specified solution point corresponding to one specific condition for Equation 10.30. The solution points of this differential equation can be expressed as

$$y_{i+1} = y_i + k_2 + O(h^3) \quad \text{10.31}$$

where $O(h^3)$ is the order of error of the step h^3, and

$$k_2 = h\left(x_i + \frac{1}{2}h,\ \ y_i + \frac{1}{2}k_1\right) \quad \text{10.32a}$$

$$k_1 = hf(x_i, y_i) \quad \text{10.32b}$$

Example 10.13

Use the second-order Runge–Kutta method shown in Equations 10.31 and 10. 32a,b to solve the following first-order ordinary differential equation similar to that in Example 7.4:

$$\frac{dy(x)}{dx} + 2y(x) = 2 \qquad \textbf{a}$$

with a specified condition $y(0) = 2$.

We will solve Equation (a) with condition $y(0) = 2$ in Example 7.4 with an exact solution of $y(x) = 1 + e^{-2x}$.

Solution:

Let us first rearrange Equation (a) in the form

$$\frac{dy(x)}{dx} = y'(x) = 2 - 2y(x) = f(x, y)$$

from which we have

$$f(x, y) = 2 - 2y \qquad \textbf{b}$$

and the specified solution point $y(0) = y_0 = 2$.

We are now ready to determine the first solution point using Equations 10.30 to 10.32a,b.

Step 1: With $i = 0$ and selected increment $h = 0.1$

$$\begin{aligned} y_1 &\approx y_0 + k_2 \\ k_1 &= 0.1(2 - 2y_0) = 0.1(2 - 2 \times 2) = -0.2 \\ k_2 &= 0.1f\left(x_0 + \frac{h}{2}, y_0 + \frac{1}{2}k_1\right) \\ &= 0.1\left[2 - 2\left(2 + \frac{-0.2}{2}\right)\right] = -0.18 \end{aligned}$$

We thus have a solution point

$$y_1 = y_0 + k_2 = 2 - 0.18 = 1.82$$

(the exact solution is $y_1 = 1.8187$). We may move the solution point forward to $i = 1$, $h = 0.1$ with $y_1 = 1.82$.

We should have the solution point: $y_2 = y_1 + k_2$ as in Equation 10.31, with

$$\begin{aligned} k_1 = hf(x_1, y_1) &= 0.1(2 - 2y_1) \\ &= 0.1(2 - 2 \times 1.82) = 0.164 \end{aligned}$$

$$\begin{aligned} k_2 &= hf\left(x_1 + \frac{1}{2}h, y_1 + \frac{1}{2}k_1\right) \\ &= 0.1\left[2 - 2\left(1.82 + \frac{1}{2}0.164\right)\right] = -0.1804 \end{aligned}$$

Hence the second solution point y_2 is

$$y_2 = y_1 + k_2 = 1.82 - 0.1804 = 1.6396$$

(the exact solution is $y_2 = 1.67$). We observe that the error of numerical solutions accumulates from 0.07% for y_1 to 1.82% for y_2.

10.5.4.2 The Fourth-order Runge–Kutta Method

This is the most popular version of the Runge–Kutta method for solving differential equations in initial value problems. Formulation of this solution method is similar to that of the second-order method.

The differential equation is similar to that shown in Equation 10.30:

$$y'(x) = f(x, y)$$

with the solution point given by the following formula:

$$y_{i+1} = y_i + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \tag{10.33}$$

where

$$k_1 = f(x_i, y_i) \tag{10.34a}$$

$$k_2 = f\left(x_i + \frac{h}{2}, y_i + \frac{k_1 h}{2}\right) \tag{10.34b}$$

$$k_3 = f\left(x_i + \frac{h}{2}, y_i + \frac{k_2 h}{2}\right) \tag{10.34c}$$

$$k_4 = f(x_i + h, y_i + k_3 h) \tag{10.34d}$$

Example 10.14

Use the Runge–Kutta fourth-order method to solve the same differential equation as in Example 10.13 but only for the second solution point y_2.

Solution:

The differential equation we will solve is

$$\frac{dy(x)}{dx} + 2y(x) = 2$$

with the given condition $y(0) = 2$.

Translating this equation into the form of the differential equation using the Runge–Kutta solution method will give $y'(x) = f(x,y)$ in Equation 10.30 with

$$f(x, y) = 2 - 2y \tag{a}$$

and $y_0 = 2$.

Example 10.13 has already solved the first solution point with $y_1 = 1.82$ with a chosen step size of $h = 0.1$. We are required to find the next solution point at y_2 with $i = 1$ and h = 0.1.

Using the expression in Equation 10.33 for the fourth-order Runge–Kutta method, we may express the solution at solution point 2 for the differential equation in Example 10.13 as

$$y_2 = y_1 + \frac{0.1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \tag{b}$$

We may obtain the constants k_1, k_2, k_3, and k_4 in Equation (b) by using Equations 10.34a, b, c, and d, respectively:

$$k_1 = (x_1, y_1) = 2 - 2y_1 = 2 - 2 \times 1.82 = -1.64 \tag{c}$$

$$k_2 = f\left(x_1 + \frac{h}{2}, y_1 + \frac{k_1 h}{2}\right) = 2 - 2\left[1.82 + \frac{(-1.64)0.1}{2}\right] = -1.476 \tag{d}$$

$$k_3 = f\left(x_1 + \frac{h}{2}, \quad y_1 + \frac{k_2 h}{2}\right) = 2 - 2\left[1.82 + \frac{(-1.476)0.1}{2}\right] = -1.4924 \tag{e}$$

$$k_4 = f\left(x_1 + \frac{h}{2}, y_1 + k_3 h\right) = 2 - 2(1.82 - 1.4924 \times 0.1) = -1.34152 \tag{f}$$

We will evaluate the solution y_2 by substituting the constants k_1, k_2, k_3, and k_4 in Equations (c), (d), (e), and (f) into Equation (b), resulting in

$$y_2 = 1.82 + \frac{0.1}{6}[-1.64 + 2(-1.476) + 2(-1.4924) + (-1.34152)] = 1.6714$$

We notice that the solution y_2 using the fourth-order Runge–Kutta method is remarkably close to the exact solution of $y_2 = 1.67$. This is a much more accurate result than that obtained using the second-order Runge–Kutta method as illustrated in Example 10.13.

10.5.4.3 Runge–Kutta Method for Higher-order Differential Equations

We have seen that Runge–Kutta method can solve differential equations often with remarkable accuracy as demonstrated in Example 10.14. Unfortunately, most textbooks offer the application of this valuable method only for solving first-order differential equations. Its use for solving higher- order differential equations requires the conversion of higher-order differential equations to the first-order-equivalent forms such as that shown in Equation 10.30. The solution of the converted higher-order differential equations can be obtained using expressions such as that given in Equation 10.33 for the fourth-order Runge–Kutta formulation. We will present the following formulation to illustrate how the fourth-order Runge–Kutta method can be used to solve second-order ordinary differential equations (the treatment is derived from an online tutorial http://www.eng.colostate.edu/~thompson/Page/CourseMat/Tutorials/CompMethods/Rungekutta.pdf).

We will use the Runge–Kutta method to solve a second-order ordinary differential equation of the form

$$\frac{d^2y(x)}{dx^2} = f\left(x, y, \frac{dy(x)}{dx}\right) = f(x, y, y') \qquad \textbf{10.35}$$

The left-hand side of Equation 10.35 may be expressed as $dy'(x)/dx$, which thus converts the second-order differential equation in (10.35) into a first-order differential equation in the form

$$\frac{dy'(x)}{dx} = f[x, y, y'(x)] \qquad \textbf{10.36a}$$

with

$$\frac{dy(x)}{dx} = y'(x) = F[x, y, y'(x)] \qquad \textbf{10.36b}$$

Solution of the second order differential equation in Equation 10.35 may be obtained by the solutions of Equations 10.36a and 10.36b using the fourth-order Runge–Kutta formulations given in Equations 10.37 and Equation 10.38:

$$y_{i+1} = y_i + \frac{1}{6}(F_1 + 2F_2 + 2F_3 + F_4)h \qquad \textbf{10.37}$$

and

$$y'_{i+1} = y_i + \frac{1}{6}(f_1 + 2f_2 + 2f_3 + f_4)h \qquad \textbf{10.38}$$

We note that the expression in Equation 10.38 is similar to that in Equation 10.33 for the converted first-order differential equation in Equation 10.36a.

The coefficients f_1, f_2, f_3, and f_4 in Equations 10.37 and 10.38 can be obtained from the expressions given in Table 10.4

Table 10.4 Coefficients in the fourth-order Runge–Kutta method for solving second-order differential equations

$f_1 = f(x_i, y_i, y'_i)$	$F_1 = y'_i$
$f_2 = f\left(x_i + \frac{1}{2}h, y_i + \frac{1}{2}F_1h, y'_i + \frac{1}{2}f_1h\right)$	$F_2 = y'_i + \frac{1}{2}f_1h$
$f_3 = f\left(x_i + \frac{1}{2}h, y_i + \frac{1}{2}F_2h + \frac{1}{4}f_1h^2, y'_i + \frac{1}{2}f_2h\right)$	$F_3 = y'_i + \frac{1}{2}f_2h$
$f_4 = f\left(x_i + h, y_i + F_3h + \frac{1}{2}f_2h, y'_i + f_3h\right)$	$F_4 = y'_i + f_3h$

Example 10.15

Use the fourth-order Runge–Kutta method to solve the following second-order differential equation with specified conditions:

$$\frac{d^2y(x)}{dx^2} - 2\frac{dy(x)}{dx} + y(x) = x^2 - 4x + 2 \qquad \textbf{a}$$

with

$$y(x)|_{x=0} = y(0) = y_0 = 0 \qquad \textbf{b1}$$

$$\left.\frac{dy(x)}{dx}\right|_{x=0} = y'(0) = y'_0 = 0 \qquad \textbf{b2}$$

Solution:

By comparing Equation (a) with Equation 10.35, we obtain the following expression for the function $f(x,y,y')$ in Equation 10.35:

$$f(x, y, y') = (x^2 - 4x + 2) - y + 2y' \qquad \textbf{c}$$

with the specified conditions $y_0 = 0$ and $y'_0 = 0$ for our subsequent numerical solution of the differential equation.

We will select step sizes $h = 0.2$, 0.1, and 0.4 for the three case illustrations using a procedure starting with the variable at $x = 0$. We begin the numerical solution for Equation (a) by letting $i = 0$ in Equation 10.37 with $x_0 = 0$ for the first solution point at $x = h = 0.2$:

We obtain the following coefficients by using the expressions for the coefficients given in Table 10.4:

$$F_1 = y'_0 = 0 \quad \text{(a given condition)}$$

$$f_1 = f(x_0, y_0, y'_0) = x_0^2 - 4x_0 + 2 - y_0 + 2y'_0 = 0 - 0 + 2 - 0 + 2 \times 0 = 2$$

$$F_2 = y'_0 + \frac{1}{2} f_1 h = 0 + \frac{1}{2} 2 \times 0.2 = 0.2$$

$$\begin{aligned} f_2 &= f\left(x_0 + \frac{h}{2}, y_0 + \frac{1}{2} F_1 h, y'_0 + \frac{1}{2} f_1 h\right) \\ &= \left(x_0 + \frac{0.2}{2}\right)^2 - 4\left(x_0 + \frac{0.2}{2}\right) + 2 - \left(y_0 + \frac{1}{2} \times 0 \times 0.2\right) + 2\left(y'_0 + \frac{1}{2} \times 2 \times 0.2\right) \\ &= 2.01 \end{aligned}$$

$$F_3 = y'_0 + \frac{1}{2} f_2 h = 0 + \frac{1}{2}(2.01) \times 0.2 = 0.201$$

$$\begin{aligned} f_3 &= f\left(x_0 + \frac{h}{2}, y_0 + \frac{1}{2} F_2 h, y'_0 + \frac{1}{2} f_2 h\right) \\ &= \left(x_0 + \frac{0.2}{2}\right)^2 - 4\left(x_0 + \frac{0.2}{2}\right) + 2 \\ &\quad - \left(y_0 + \frac{1}{2} 0.2 \times 0.2\right) + 2\left(y'_0 + \frac{1}{2}(2.01) \times 0.2\right) \\ &= 1.992 \end{aligned}$$

$$F_4 = y'_0 + f_3 h = 0 + 1.992 \times 0.2 = 0.3984$$

$$\begin{aligned} f_4 &= f(x_0 + h, y_0 + F_3 h + \frac{1}{2} f_2 h, y'_0 + f_3 h) \\ &= [(0+0.2)^2 - 4(0+0.2) + 2] - (0 + 0.201 \times 0.2 + \frac{1}{2} \times 2.01 \times 0.2) + 2(0 + 1.992 \times 0.2) \\ &= 1.8078 \end{aligned}$$

We are ready to find the numerical solution of the differential equation in Equation (a) by substituting the values of f_1, f_2, and f_3 into Equation 10.37, to obtain a solution point y_1 with $i = 0$ and $h = 0.2$:

$$\begin{aligned} y_1 = y(0.2) &= y_0 + y'_0 \times 0.2 + \frac{1}{6}(F_1 + 2F_2 + 2F_3 + F_4)h \\ &= 0 + 0 + \frac{1}{6}(0 + 2 \times 0.2 + 2 \times 0.201 + 0.3984) \times 0.2 \\ &= 0.4001333 \end{aligned} \quad \textbf{d}$$

The exact solution of Equation (a) is $y(x) = x^2$, which yields an exact solution of $y(0.2) = 0.04$. The numerical solution in Equation (d) has a 0.033% error from the exact solution.

We will also use Equation 10.38 to approximate the value of the first-order derivation $y'(0.2)$ as

$$\begin{aligned} y'_1 = y'(0.2) &= y_0 + \frac{1}{6}(f_1 + 2f_2 + 2f_3 + f_4)h \\ &= 0 + \frac{1}{6}(2 + 2 \times 2.01 + 2 \times 1.992 + 1.8078)0.2 \\ &= 0.3937 \end{aligned} \quad \textbf{e}$$

One may use the same procedure to obtain the solution of Equation (a) at point $x = x + h$. For instance, the next solution point is at $x = 0.2 + h = 0.4$, or $y(0.4)$ by letting $i = 1$ with $x_1 = 0.2$, $y_1 = 0.0400133$ and $y'_1 = 0.3937$.

Like all other numerical solution methods for solving differential equations, the error of the approximated solutions depends largely on the step size h chosen by the analyst. We will demonstrate the effects of the chosen increment size h for the same differential equation as in Equation (a) in two other cases with h-values of h = 0.1 and 0.4. The results of these two cases, together with the case h = 0.2, are summarized in Table 10.5.

Table 10.5 Solutions of a differential equation by Runge–Kutta methods with three different increment sizes.

	Case 1	Case 2	Case 3
x_0	0	0	0
h	0.2	0.1	0.4
F_1	0	0	0
F_2	0.2	0.1	0.4
F_3	0.201	0.100125	0.408
F_4	0.3984	1.999537	0.7904
f_1	2	2	2
f_2	2.01	2.0025	2.04
f_3	1.992	1.99775	1.976
f_4	1.8078	1.999537	1.9446
$y'(x)$	0.3937	0.2000625	0.79844
Approximated $y(x_0 + h)$	0.0400133	0.0100004	0.160427
Exact $y(x_0 + h)$	0.04	0.01	0.16
% Error on $y(h)$	**0.033**	**0.0042**	**0.2667**
% Error on y'(h)	**0.005**	**0.000313**	**0.195**

One may observe from the above tabulation that the step size indeed has a significant effect on the accuracy of the numerical solutions of differential equations. It is no surprise that the larger step size h = 0.4 resulted in the largest error in the solution, whereas the smaller step size h = 0.1 appeared to give the closest solution to the exact solution. This latter particular step size appears to be the optimal size for the accurate solution for this particular differential equation.

The three solution points at x = 0.2, 0.1, and 0.4 obtained by using the Runge–Kutta method appeared tedious and time-consuming. The same differential equation with the same specified conditions was solved using the software package MatLAB, with the input/output information included in Case 3 of Appendix 4. The results so obtained were remarkably accurate, with solution at the same three points being the exact values as shown tabulated above. MatLAB also offers graphical output such as that shown in Figure 10.17. An overview description of MatLAB software will be presented in Section 10.6.2.

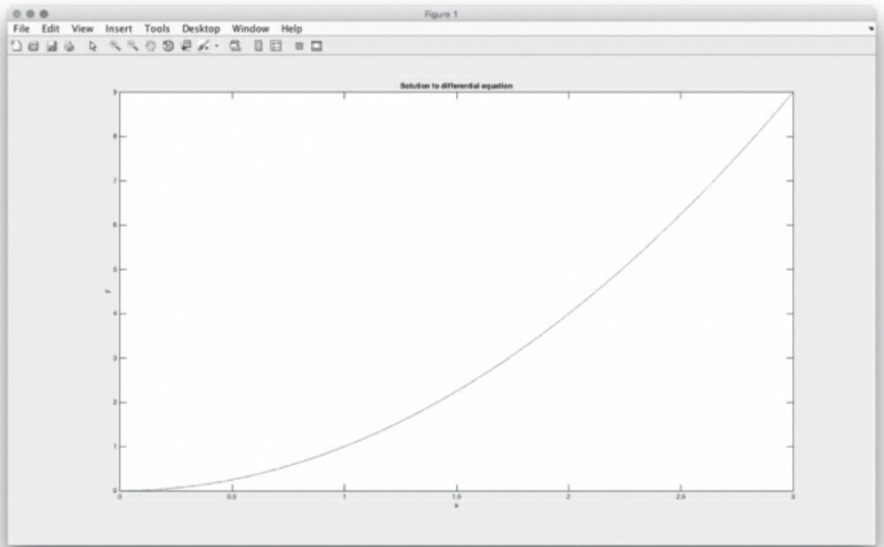

Figure 10.17 Graphical solution of a second-order ordinary differential equation by MatLAB.

Example 10.16

Use the fourth-order Runge–Kutta method to obtain the solution of the function $x(t)$ at t = 0.2 in the second-order differential equation in Example 10.12.

Solution:

The differential equation in Example 10.12 is

$$\frac{d^2x(t)}{dt^2} + x(t) = 0 \quad \text{a}$$

with the conditions $x(0) = 1$ and $x'(0) = 0$. Since we do not have the term $x'(t)$ in Equation (a), we will not need to evaluate F_1, F_2, F_3, and F_4 in Table 10.5.

We will use the same step size h = 0.05 as was used in Example 10.12. The numerical solution at the previous step at $x(0.15) = 0.9925$ will be computed in this example. Since the exact solution of Equation (a) with the specified conditions is $x(t) = \cos t$, we may obtain the first-order derivative of $x(t)$ at $x = 0.15$ as $x'(0.15) = -\sin(0.15) = -0.149\,44$. We thus have the following converted equation and conditions for the present case:

$$\frac{d^2x(t)}{dt^2} = x(t) = f(t, x, x')$$
$$\frac{dx(t)}{dt} = x'(t)$$

and $x_0 = 0.9925$ (the solution obtained in Example 10.12) and $x_0' = -0.149\,44$.

We will use the fourth-order Runge–Kutta method as shown in Equations 10.33 and 10.38 and the coefficients given in Table 10.4 to obtain the solution at $x(0.2)$ as follows. Let $i = 0$, $h = 0.05$, $x_0 = 0.9925$, and $x_0' = -0.149\,44$ for the solution at $t = 0.2$, or

$$x_1 = x(0.2) = x_0 + x_0'h + \frac{1}{6}(f_1 + f_2 + f_3)h^2 \quad \text{b}$$

We evaluate the coefficients f_1, f_2, f_3, and f_4 from Table 10.4 as follows:

$$f_1 = f(t_0, x_0, x_0') = x_0 = -0.149\,44$$
$$f_2 = x_0' + \frac{1}{2}f_1h = -0.14944 + \frac{(-0.14944) \times 0.05}{2} = -0.153\,176$$
$$f_3 = x_0' + \frac{1}{2}f_2h = -0.14944 + \frac{(-0.153176) \times 0.05}{2} = -0.153\,694$$
$$f_4 = x_0' + f_3h = -0.14944 - 0.153694 \times 0.05 = -0.157\,103\,47$$

According to Equation 10.38, we have the solution

$$\begin{aligned} x_1 = x(0.2) &= x_0 + x_0'h + \frac{1}{6}(f_1 + f_2 + f_3)h^2 \\ &= 0.9925 + (-0.14944) \times 0.05 + \frac{1}{6}(-0.14944 - 0.153176 - 0.153694)(0.05)^2 \\ &= 0.9848 \end{aligned}$$

This numerical solution has an error of 0.48% from the exact solution, and it is more accurate than that obtained from the simple forward difference scheme in Example 10.12.

10.6 Introduction to Numerical Analysis Software Packages

We have demonstrated the power, and thus the value of numerical methods in solving many problems in engineering analysis involving nonlinear equations, integrations, and differential equations in this chapter. These methods typically require significant time and efforts in arriving at approximate, not exact, solutions of the problem, and the solutions obtained are only available at discrete solution points. More accurate solutions are obtainable with small increment step sizes but with correspondingly more computational effort.

Since almost all the numerical methods involve massive computational effort and the solutions are available only at discrete solution points, sophisticated symbolic manipulation computer packages such as the popular commercially available Mathematica and MatLAB have proven to be valuable tools for engineers in engineering analyses associated with tedious computational processes. In this section, we will briefly survey these two numerical analysis packages, in particular their capabilities in solving engineering problems. Readers are referred to the literature for some excellent references (Malek-Madani, 1998; Chapra, 2012) providing more detailed descriptions and guidance on the effective use of these packages.

10.6.1 Introduction to Mathematica

Mathematica is a computational software program based on symbolic mathematics. It is used in many scientific, engineering, mathematical, and computing fields. The programming languages used in Mathematica are the Wolfram Language by Stephen Wolfram, together with C, C++, and Java. This software package has been in the marketplace since June 1988. It is capable of handling the following major functions and features in engineering analysis:

1. Determining roots of polynomial equations of cubic or higher orders.

2. Integrating and differentiating complicated expressions. For instance, the integral of the function $\sin(x^3)$, for which $\int_0^1 \sin(x^3)\,dx = 0.2338$; in contrast, similar integrals with the simple function $\sin(x)$ but with higher power of the function in the integral, such as $\int_0^1 \sin^3(x)\,dx = 0.3063$, are available from integration tables in mathematical handbooks (Zwillinger, 2003).

3. Solving linear and nonlinear differential equations.

4. Elementary and special mathematical function libraries.

5. Matrix and data manipulation tools.

6. Numeric and symbolic tools for discrete and continuous calculus.

It can also solve the following common analytical engineering problems:

1. Determining Laplace and Fourier transforms of functions.

2. Generating graphics in two and three dimensions.

3. Simplifying trigonometric and algebraic expressions.

According to the Wolfram Language and Systems Documentation Center, Mathematica has the following features and capabilities that are of great value in advanced engineering analyses:

- Support for complex number, arbitrary precision, interval arithmetic, and symbolic computation.
- Solvers for systems of equations, Diophantine equations, ODEs, PDEs, and so on.
- Multivariate statistics libraries including fitting, hypothesis testing, and probability and expectation calculations on over 140 distributions.
- Calculations and simulations on random processes and queues.
- Computational geometry in 2D, 3D, and higher dimensions.
- Finite-element analysis including 2D and 3D adaptive mesh generation.
- Constrained and unconstrained local and global optimization.

- Toolkit for adding user interfaces to calculations and applications.
- Tools for 2D and 3D image processing and morphological image processing including image recognition.
- Tools for visualizing and analyzing directed and undirected graphs.
- Tools for combinatorics problems.
- Data mining tools such as cluster analysis, sequence alignment, and pattern matching.
- Group theory and symbolic tensor functions.
- Libraries for signal processing including wavelet analysis on sounds, images, and data.
- Linear and nonlinear control systems libraries.
- Continuous and discrete integral transforms.
- Import and export filters for data, images, video, sound, CAD, GIS, documents, and biomedical formats.
- Database collection for mathematical, scientific, and socioeconomic information and access to Wolfram alpha data and computations.
- Technical word processing including formula editing and automated report generation.
- Tools for connecting to DLL, SQL, Java, .NET, C++, Fortran, CUDA, OpenCL, and http based systems.
- Tools for parallel programming.
- Mathematica language in notebook computers when connected to the Internet.

The last of the features listed is of particular value to engineers. For example, in Example 10.3 we were required to find the root of the cubic equation $L^3 + 70.3L^2 + 1647.39L - 18656.72 = 0$. A meaningful root of this equation found by the Newton–Raphson method was $L = 8.15$ as shown in Example 10.3. A similar solution of $L = 8.1566$ was obtained by the solution method offered via the internet at the Wolfram/Alpha Widgets website (www.wolframalpha.com/widgets/) with user input of the coefficients of this equation. It offered an instant solution and with an excellent user interface feature.

10.6.2 Introduction to MATLAB

MATLAB is an acronym of "**mat**rix **lab**oratory." This numerical analysis package was designed by Cleve Moler in the late 1970s with an initial release to the public in 1984. The latest version, Version 8.6 was released in September 2015.

MATLAB provides a multiparadigm numerical computing environment and fourth-generation programming language, a proprietary programming language developed by MathWorks. It allows matrix manipulations, plotting of functions and data, implementation of algorithms, and creation of user interfaces that include interfacing with programs written in other languages, including C, C++, Java, Fortran, and Python. It is a popular numerical analysis package mainly because of it has graphics and graphical user interfacing programming capability.

Like Mathematica, MATLAB is capable of handling the following common problems in engineering analysis (Malek-Madani, 1998):

1. Finding roots of polynomials, summing series, and determining limits of sequences.
2. Symbolically integrating and differentiating complicated expressions.
3. Plotting graphics in two and three dimensions.
4. Simplify trigonometric and algebraic expressions,
5. Solving linear and nonlinear differential equations.
6. Determining the Laplace transforms of functions.

Additionally it can handle a variety of other mathematical operations.

Operation of MATLAB requires the user to input simple programs for the solution of the problems. These programs usually consists of three "windows:" (1) the "command window" for the user to enter commands and data; (2) the "graphics window" to display the results in plots and graphs; and (3) the "edit window" to create and edit the M-files, which provide alternative ways of performing operations that can expand MATLAB's problem-solving capabilities.

Detailed instructions for using MATLAB for solving a variety of mathematical problems are available in *MATLAB Primer* published by MathWorks, Inc.(see www.mathworks.com) and two excellent references (Malek-Madani, 1998 Chapra, 2012). Appendix 4 will present the procedure for the input/output (I/O) of three cases of engineering analysis using the MatLAB package:

Case 1: Graphic solution of the amplitudes with "beats" offered by the solution in Equation (8.40) for the near-resonant vibration of a metal stamping machine.

Case 2: The numerical solution with graphic representations of the amplitudes and mode shapes of a flexible rectangular pad subjected to transverse vibration as presented in Equation (9.76).

Case 3: The solution with graphic representation of a nonhomogeneous second-order differential equation.

These cases will demonstrate the value of the MatLAB software package in solving complicated engineering analysis problems.

Problems

10.1 Use the Newton–Raphson method to solve the following nonlinear equation in Example 10.1: $x^4 - 2x^3 + x^2 - 3x + 3 = 0$.

10.2 A measuring cup illustrated in Figure 10.18a has the overall dimensions shown in Figure 10.18b. (a) Determine the overall volume of the cup. (b) Derive the equation to locate the mark for the volume of 150 ml at the height L from the bottom of the cup, and use the Newton–Raphson method to solve this equation.

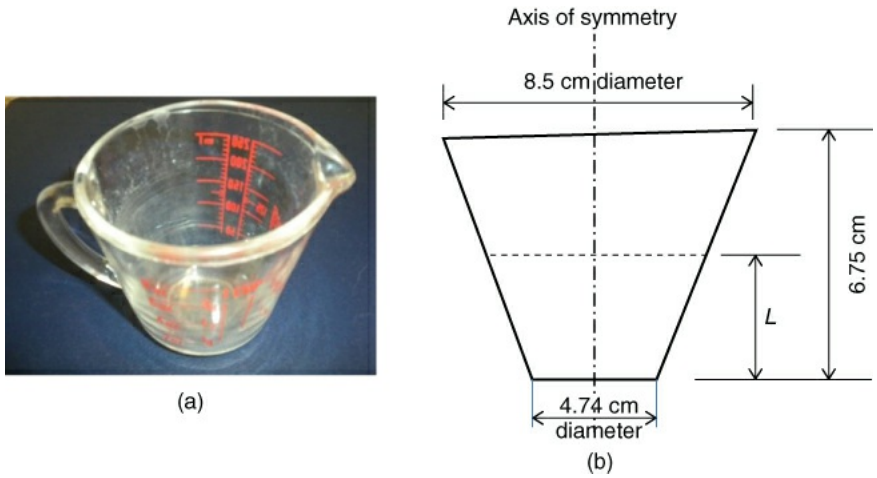

Figure 10.18 Design of a measuring cup. (a) A measuring cup. (b) The overall dimensions of the cup.

10.3 In Example 8.10, we derived a nonlinear equation for the amplitude of the mass attached to a spring in resonant vibration as Equation (m) in that example of the form

$$y(t) = \frac{11}{20}\sin 10t - \frac{t}{2}\cos 10t$$

where $y(t)$ is the amplitude of the vibrating mass at time t. Use the Newton–Raphson method to find the time t_e, at which the spring reaches the breaking stretching extent of 0.005 m.

10.4 Use the three numerical integration methods—the trapezoidal rule, Simpson's one-third rule, and Gaussian quadrature with two-sampling points—to determine the values of the following three integrals:

a.

$$I_1 = \int_0^1 \sin x^3 \, dx$$

b.

$$I_2 = \int_0^1 \sin^3 x \, dx$$

c.

$$I_3 = \int_0^1 e^{2x} \sin^3 x \, dx$$

10.5 Use the three numerical integration methods—the trapezoidal rule, Simpson's one-third rule, and Gaussian quadrature with two-sampling points—to determine the plane area of a plate in the form of an ellipse as shown in Figure 10.19.

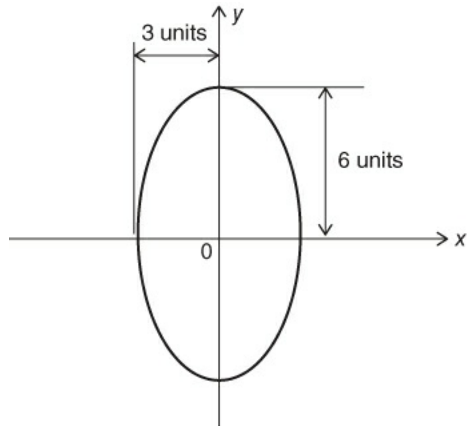

Figure 10.19 Plane area of an ellipse.

10.6 Derive the second-order derivative of the function $f(x)$ as shown in Equation 10.23 for the "backward difference" scheme, and also for the "central difference scheme."

10.7 Use the backward and central difference schemes to solve the differential equation in Example 10.12.

10.8 Use the forward difference scheme to solve the differential equation derived for the modeling of the vibration of a metal stamping machine in Example 8.9. Compare your results obtained from this finite-difference approximation and the exact solution given in the example.

10.9 Use both second-order and fourth-order Runge–Kutta methods to solve the first-order differential equation, Equation (7.13), for the instantaneous water level $h(t)$ in a straight-sided cylindrical tank:

$$\frac{dh(t)}{dt} = -\sqrt{2g}\left(\frac{d^2}{D^2}\right)\sqrt{h(t)}$$

with $g = 32.2\ \mathrm{ft/s^2}$, $D = 12$ inches, $d = 1$ inch, and initial water level $h(0) = h_0 = 12$ inches. Compare your result at two solution points with that from Equation (7.14).

10.10 Use the forward difference scheme to solve the following differential equation with a step size of $\Delta t = 0.1$ and with prescribed conditions $y(0) = 0$ and $y'(0) = 10$.

$$2\frac{d^2y(t)}{dt^2} + 32\frac{dy(t)}{dt} + 100y(t) = 0$$

You need to present the formulations for each solution step for a total of three steps.

10.11 Use the fourth-order Runge–Kutta method to solve the same equation in Problem 10.10 with the same step size. Compare your results obtained from these two methods.

10.12 Use the fourth-order Runge–Kutta method to solve the following differential equation in Equation (a) in Example 8.10:

$$\frac{d^2y(t)}{dt^2} + 16\frac{dy(t)}{dt} + 100y(t) = 10\sin 10t$$

with $y(0) = y_0 = 0$ and

$$\left.\frac{dy(t)}{dt}\right|_{t=0} = y'(0) = 5\ \ \mathrm{m/s}$$

Compare your result with the exact solution of $y(2) = 0.25$ cm obtained from the analytical solution of the same equation as indicated in Example 8.10.