

PARTIAL DIFFERENTIAL EQUATIONS

PT8.1 MOTIVATION

Given a function u that depends on both x and y , the partial derivative of u with respect to x at an arbitrary point (x, y) is defined as

$$\frac{\partial u}{\partial x} = \lim_{\Delta x \rightarrow 0} \frac{u(x + \Delta x, y) - u(x, y)}{\Delta x} \quad (\text{PT8.1})$$

Similarly, the partial derivative with respect to y is defined as

$$\frac{\partial u}{\partial y} = \lim_{\Delta y \rightarrow 0} \frac{u(x, y + \Delta y) - u(x, y)}{\Delta y} \quad (\text{PT8.2})$$

An equation involving partial derivatives of an unknown function of two or more independent variables is called a *partial differential equation*, or *PDE*. For example,

$$\frac{\partial^2 u}{\partial x^2} + 2xy \frac{\partial^2 u}{\partial y^2} + u = 1 \quad (\text{PT8.3})$$

$$\frac{\partial^3 u}{\partial x^2 \partial y} + x \frac{\partial^2 u}{\partial y^2} + 8u = 5y \quad (\text{PT8.4})$$

$$\left(\frac{\partial^2 u}{\partial x^2} \right)^3 + 6 \frac{\partial^3 u}{\partial x \partial y^2} = x \quad (\text{PT8.5})$$

$$\frac{\partial^2 u}{\partial x^2} + xu \frac{\partial u}{\partial y} = x \quad (\text{PT8.6})$$

The *order* of a PDE is that of the highest-order partial derivative appearing in the equation. For example, Eqs. (PT8.3) and (PT8.4) are second- and third-order, respectively.

A partial differential equation is said to be *linear* if it is linear in the unknown function and all its derivatives, with coefficients depending only on the independent variables. For example, Eqs. (PT8.3) and (PT8.4) are linear, whereas Eqs. (PT8.5) and (PT8.6) are not.

Because of their widespread application in engineering, our treatment of PDEs will focus on linear, second-order equations. For two independent variables, such equations can be expressed in the following general form:

$$A \frac{\partial^2 u}{\partial x^2} + B \frac{\partial^2 u}{\partial x \partial y} + C \frac{\partial^2 u}{\partial y^2} + D = 0 \quad (\text{PT8.7})$$

where A , B , and C are functions of x and y and D is a function of x , y , u , $\partial u / \partial x$, and $\partial u / \partial y$. Depending on the values of the coefficients of the second-derivative terms— A , B , C —

TABLE PT8.1 Categories into which linear, second-order partial differential equations in two variables can be classified.

$B^2 - 4AC$	Category	Example
< 0	Elliptic	Laplace equation (steady state with two spatial dimensions) $\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0$
$= 0$	Parabolic	Heat conduction equation (time variable with one spatial dimension) $\frac{\partial T}{\partial t} = k' \frac{\partial^2 T}{\partial x^2}$
> 0	Hyperbolic	Wave equation (time variable with one spatial dimension) $\frac{\partial^2 y}{\partial x^2} = \frac{1}{c^2} \frac{\partial^2 y}{\partial t^2}$

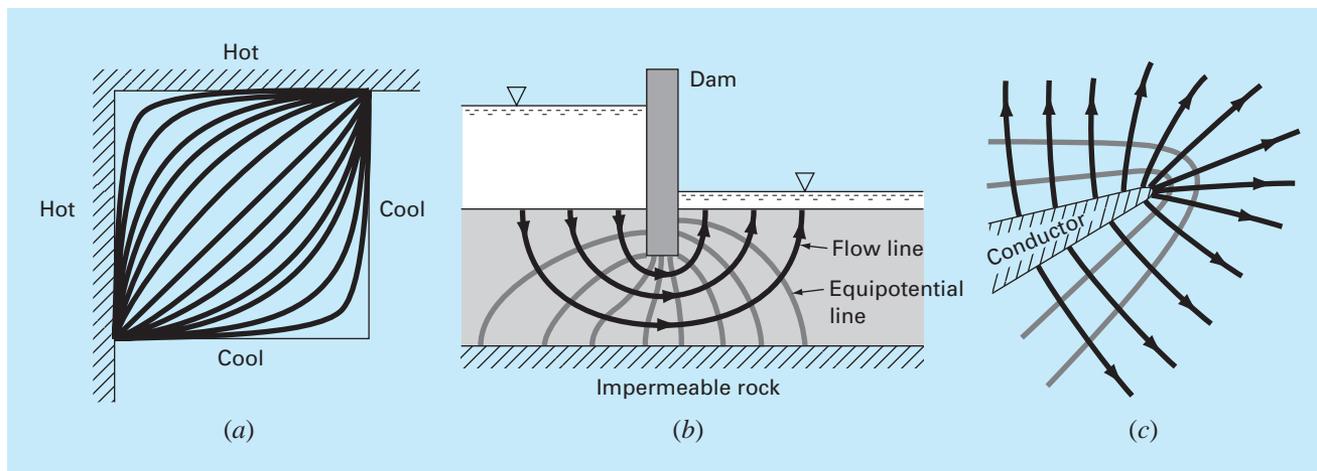
Eq. (PT8.7) can be classified into one of three categories (Table PT8.1). This classification, which is based on the method of characteristics (for example, see Vichnevetsky, 1981, or Lapidus and Pinder, 1981), is useful because each category relates to specific and distinct engineering problem contexts that demand special solution techniques. It should be noted that for cases where A , B , and C depend on x and y , the equation may actually fall into a different category, depending on the location in the domain for which the equation holds. For simplicity, we will limit the present discussion to PDEs that remain exclusively in one of the categories.

PT8.1.1 PDEs and Engineering Practice

Each of the categories of partial differential equations in Table PT8.1 conforms to specific kinds of engineering problems. The initial sections of the following chapters will be devoted to deriving each type of equation for a particular engineering problem context. For the time being, we will discuss their general properties and applications and show how they can be employed in different physical contexts.

Elliptic equations are typically used to characterize *steady-state* systems. As in the *Laplace equation* in Table PT8.1, this is indicated by the absence of a time derivative. Thus, these equations are typically employed to determine the steady-state distribution of an unknown in two spatial dimensions.

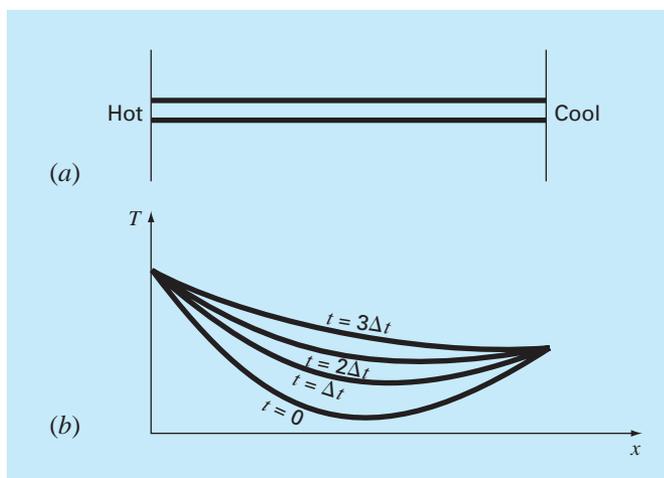
A simple example is the heated plate in Fig. PT8.1a. For this case, the boundaries of the plate are held at different temperatures. Because heat flows from regions of high to low temperature, the boundary conditions set up a potential that leads to heat flow from the hot to the cool boundaries. If sufficient time elapses, such a system will eventually reach the stable or steady-state distribution of temperature depicted in Fig. PT8.1a. The Laplace equation, along with appropriate boundary conditions, provides a means to determine this distribution. By analogy, the same approach can be employed to tackle other problems involving potentials, such as seepage of water under a dam (Fig. PT8.1b) or the distribution of an electric field (Fig. PT8.1c).

**FIGURE PT8.1**

Three steady-state distribution problems that can be characterized by elliptic PDEs. (a) Temperature distribution on a heated plate, (b) seepage of water under a dam, and (c) the electric field near the point of a conductor.

FIGURE PT8.2

(a) A long, thin rod that is insulated everywhere but at its end. The dynamics of the one-dimensional distribution of temperature along the rod's length can be described by a parabolic PDE. (b) The solution, consisting of distributions corresponding to the state of the rod at various times.

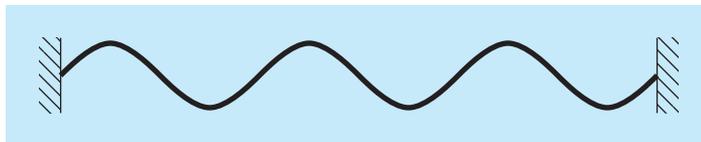


In contrast to the elliptic category, *parabolic equations* determine how an unknown varies in both space and time. This is manifested by the presence of both spatial and temporal derivatives in the *heat conduction equation* from Table PT8.1. Such cases are referred to as *propagation problems* because the solution “propagates,” or changes, in time.

A simple example is a long, thin rod that is insulated everywhere except at its end (Fig. PT8.2a). The insulation is employed to avoid complications due to heat loss along the

FIGURE PT8.3

A taut string vibrating at a low amplitude is a simple physical system that can be characterized by a hyperbolic PDE.



rod's length. As was the case for the heated plate in Fig. PT8.1a, the ends of the rod are set at fixed temperatures. However, in contrast to Fig. PT8.1a, the rod's thinness allows us to assume that heat is distributed evenly over its cross section—that is, laterally. Consequently, lateral heat flow is not an issue, and the problem reduces to studying the conduction of heat along the rod's longitudinal axis. Rather than focusing on the steady-state distribution in two spatial dimensions, the problem shifts to determining how the one-dimensional spatial distribution changes as a function of time (Fig. PT8.2b). Thus, the solution consists of a series of spatial distributions corresponding to the state of the rod at various times. Using an analogy from photography, the elliptic case yields a portrait of a system's stable state, whereas the parabolic case provides a motion picture of how it changes from one state to another. As with the other types of PDEs described herein, parabolic equations can be used to characterize a wide variety of other engineering problem contexts by analogy.

The final class of PDEs, the *hyperbolic* category, also deals with *propagation problems*. However, an important distinction manifested by the wave equation in Table PT8.1 is that the unknown is characterized by a second derivative with respect to time. As a consequence, the solution oscillates.

The vibrating string in Fig. PT8.3 is a simple physical model that can be described with the wave equation. The solution consists of a number of characteristic states with which the string oscillates. A variety of engineering systems such as vibrations of rods and beams, motion of fluid waves, and transmission of sound and electrical signals can be characterized by this model.

PT8.1.2 Precomputer Methods for Solving PDEs

Prior to the advent of digital computers, engineers relied on analytical or exact solutions of partial differential equations. Aside from the simplest cases, these solutions often required a great deal of effort and mathematical sophistication. In addition, many physical systems could not be solved directly but had to be simplified using linearizations, simple geometric representations, and other idealizations. Although these solutions are elegant and yield insight, they are limited with respect to how faithfully they represent real systems—especially those that are highly nonlinear and irregularly shaped.

PT8.2 ORIENTATION

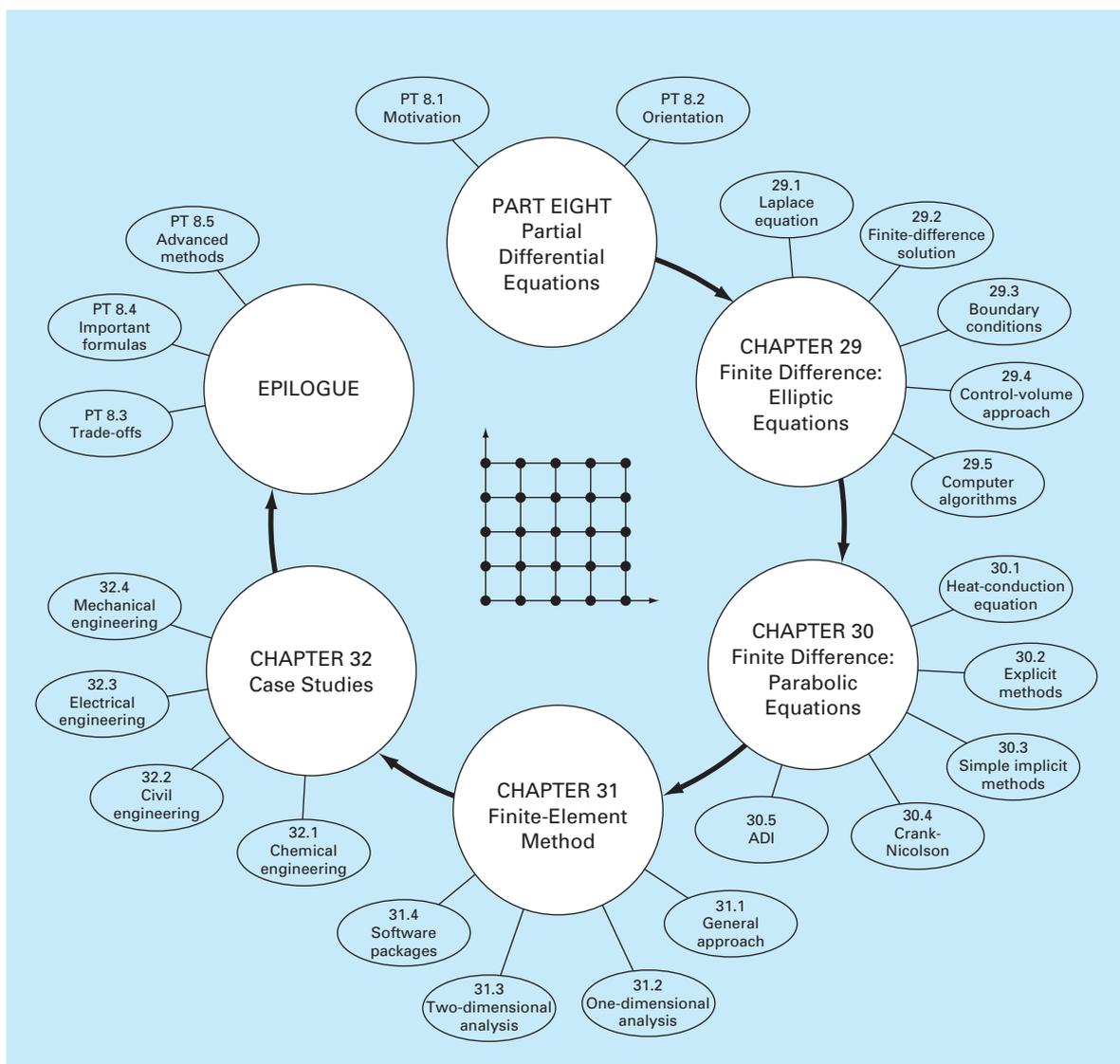
Before we proceed to the numerical methods for solving partial differential equations, some orientation might be helpful. The following material is intended to provide you with an overview of the material discussed in Part Eight. In addition, we have formulated objectives to focus your studies in the subject area.

PT8.2.1 Scope and Preview

Figure PT8.4 provides an overview of Part Eight. Two broad categories of numerical methods will be discussed in this part of this book. Finite-difference approaches, which are covered in Chaps. 29 and 30, are based on approximating the solution at a finite number of points. In contrast, finite-element methods, which are covered in Chap. 31, approximate

FIGURE PT8.4

Schematic representation of the organization of material in Part Eight: Partial Differential Equations.



the solution in pieces, or “elements.” Various parameters are adjusted until these approximations conform to the underlying differential equation in an optimal sense.

Chapter 29 is devoted to *finite-difference* solutions of *elliptic equations*. Before launching into the methods, we derive the Laplace equation for the physical problem context of the temperature distribution for a heated plate. Then, a standard solution approach, the *Liebmann method*, is described. We will illustrate how this approach is used to compute the distribution of the primary scalar variable, temperature, as well as a secondary vector variable, heat flux. The final section of the chapter deals with *boundary conditions*. This material includes procedures to handle different types of conditions as well as irregular boundaries.

In *Chap. 30*, we turn to *finite-difference* solutions of *parabolic equations*. As with the discussion of elliptic equations, we first provide an introduction to a physical problem context, the heat-conduction equation for a one-dimensional rod. Then we introduce both explicit and implicit algorithms for solving this equation. This is followed by an efficient and reliable implicit method—the *Crank-Nicolson technique*. Finally, we describe a particularly effective approach for solving two-dimensional parabolic equations—the *alternating-direction implicit*, or *ADI*, method.

Note that, because they are somewhat beyond the scope of this book, we have chosen to omit hyperbolic equations. The epilogue of this part of the book contains references related to this type of PDE.

In *Chap. 31*, we turn to the other major approach for solving PDEs—the *finite-element method*. Because it is so fundamentally different from the finite-difference approach, we devote the initial section of the chapter to a general overview. Then we show how the finite-element method is used to compute the steady-state temperature distribution of a heated rod. Finally, we provide an introduction to some of the issues involved in extending such an analysis to two-dimensional problem contexts.

Chapter 32 is devoted to applications from all fields of engineering. Finally, a short review section is included at the end of Part Eight. This epilogue summarizes important information related to PDEs. This material includes a discussion of trade-offs that are relevant to their implementation in engineering practice. The epilogue also includes references for advanced topics.

PT8.2.2 Goals and Objectives

Study Objectives. After completing Part Eight, you should have greatly enhanced your capability to confront and solve partial differential equations. General study goals should include mastering the techniques, having the capability to assess the reliability of the answers, and being able to choose the “best” method (or methods) for any particular problem. In addition to these general objectives, the specific study objectives in Table PT8.2 should be mastered.

Computer Objectives. Computer algorithms can be developed for many of the methods in Part Eight. For example, you may find it instructive to develop a general program to simulate the steady-state distribution of temperature on a heated plate. Further, you might want to develop programs to implement both the simple explicit and the Crank-Nicolson methods for solving parabolic PDEs in one spatial dimension.

TABLE PT8.2 Specific study objectives for Part Eight.

1. Recognize the difference between elliptic, parabolic, and hyperbolic PDEs.
2. Understand the fundamental difference between finite-difference and finite-element approaches.
3. Recognize that the Liebmann method is equivalent to the Gauss-Seidel approach for solving simultaneous linear algebraic equations.
4. Know how to determine secondary variables for two-dimensional field problems.
5. Recognize the distinction between Dirichlet and derivative boundary conditions.
6. Understand how to use weighting factors to incorporate irregular boundaries into a finite-difference scheme for PDEs.
7. Understand how to implement the control-volume approach for implementing numerical solutions of PDEs.
8. Know the difference between convergence and stability of parabolic PDEs.
9. Understand the difference between explicit and implicit schemes for solving parabolic PDEs.
10. Recognize how the stability criteria for explicit methods detract from their utility for solving parabolic PDEs.
11. Know how to interpret computational molecules.
12. Recognize how the ADI approach achieves high efficiency in solving parabolic equations in two spatial dimensions.
13. Understand the difference between the direct method and the method of weighted residuals for deriving element equations.
14. Know how to implement Galerkin's method.
15. Understand the benefits of integration by parts during the derivation of element equations; in particular, recognize the implications of lowering the highest derivative from a second to a first derivative.

Finally, one of your most important goals should be to master several of the general-purpose software packages that are widely available. In particular, you should become adept at using these tools to implement numerical methods for engineering problem solving.

Finite Difference: Elliptic Equations

Elliptic equations in engineering are typically used to characterize steady-state, boundary-value problems. Before demonstrating how they can be solved, we will illustrate how a simple case—the Laplace equation—is derived from a physical problem context.

29.1 THE LAPLACE EQUATION

As mentioned in the introduction to this part of the book, the Laplace equation can be used to model a variety of problems involving the potential of an unknown variable. Because of its simplicity and general relevance to most areas of engineering, we will use a heated plate as our fundamental context for deriving and solving this elliptic PDE. Homework problems and engineering applications (Chap. 32) will be employed to illustrate the applicability of the model to other engineering problem contexts.

Figure 29.1 shows an element on the face of a thin rectangular plate of thickness Δz . The plate is insulated everywhere but at its edges, where the temperature can be set at a prescribed level. The insulation and the thinness of the plate mean that heat transfer is limited to the x and y dimensions. At steady state, the flow of heat into the element over a unit time period Δt must equal the flow out, as in

$$q(x) \Delta y \Delta z \Delta t + q(y) \Delta x \Delta z \Delta t = q(x + \Delta x) \Delta y \Delta z \Delta t + q(y + \Delta y) \Delta x \Delta z \Delta t \quad (29.1)$$

where $q(x)$ and $q(y)$ = the heat fluxes at x and y , respectively [$\text{cal}/(\text{cm}^2 \cdot \text{s})$]. Dividing by Δz and Δt and collecting terms yields

$$[q(x) - q(x + \Delta x)] \Delta y + [q(y) - q(y + \Delta y)] \Delta x = 0$$

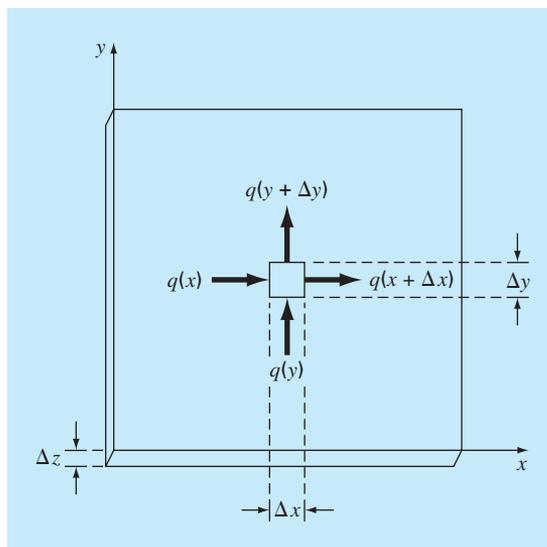
Multiplying the first term by $\Delta x/\Delta x$ and the second by $\Delta y/\Delta y$ gives

$$\frac{q(x) - q(x + \Delta x)}{\Delta x} \Delta x \Delta y + \frac{q(y) - q(y + \Delta y)}{\Delta y} \Delta y \Delta x = 0 \quad (29.2)$$

Dividing by $\Delta x \Delta y$ and taking the limit results in

$$-\frac{\partial q}{\partial x} - \frac{\partial q}{\partial y} = 0 \quad (29.3)$$

where the partial derivatives result from the definitions in Eqs. (PT7.1) and (PT7.2).

**FIGURE 29.1**

A thin plate of thickness Δz . An element is shown about which a heat balance is taken.

Equation (29.3) is a partial differential equation that is an expression of the conservation of energy for the plate. However, unless heat fluxes are specified at the plate's edges, it cannot be solved. Because temperature boundary conditions are given, Eq. (29.3) must be reformulated in terms of temperature. The link between flux and temperature is provided by *Fourier's law of heat conduction*, which can be represented as

$$q_i = -k\rho C \frac{\partial T}{\partial i} \quad (29.4)$$

where q_i = heat flux in the direction of the i dimension [$\text{cal}/(\text{cm}^2 \cdot \text{s})$], k = coefficient of *thermal diffusivity* (cm^2/s), ρ = density of the material (g/cm^3), C = heat capacity of the material [$\text{cal}/(\text{g} \cdot ^\circ\text{C})$], and T = temperature ($^\circ\text{C}$), which is defined as

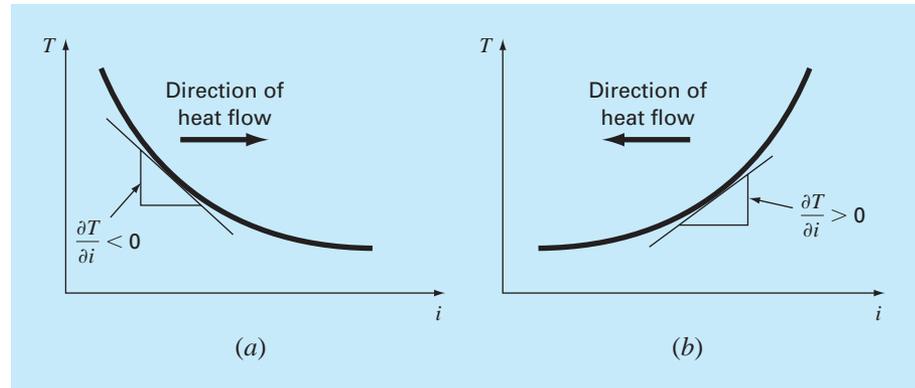
$$T = \frac{H}{\rho CV}$$

where H = heat (cal) and V = volume (cm^3). Sometimes the term in front of the differential in Eq. (29.3) is treated as a single term,

$$k' = k\rho C \quad (29.5)$$

where k' is referred to as the *coefficient of thermal conductivity* [$\text{cal}/(\text{s} \cdot \text{cm} \cdot ^\circ\text{C})$]. In either case, both k and k' are parameters that reflect how well the material conducts heat.

Fourier's law is sometimes referred to as a *constitutive equation*. It is given this label because it provides a mechanism that defines the system's internal interactions. Inspection

**FIGURE 29.2**

Graphical depiction of a temperature gradient. Because heat moves “downhill” from high to low temperature, the flow in (a) is from left to right in the positive i direction. However, due to the orientation of Cartesian coordinates, the slope is negative for this case. Thus, a negative gradient leads to a positive flow. This is the origin of the minus sign in Fourier’s law of heat conduction. The reverse case is depicted in (b), where the positive gradient leads to a negative heat flow from right to left.

of Eq. (29.4) indicates that Fourier’s law specifies that heat flux perpendicular to the i axis is proportional to the gradient or slope of temperature in the i direction. The negative sign ensures that a positive flux in the direction of i results from a negative slope from high to low temperature (Fig. 29.2). Substituting Eq. (29.4) into Eq. (29.3) results in

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0 \quad (29.6)$$

which is the *Laplace equation*. Note that for the case where there are sources or sinks of heat within the two-dimensional domain, the equation can be represented as

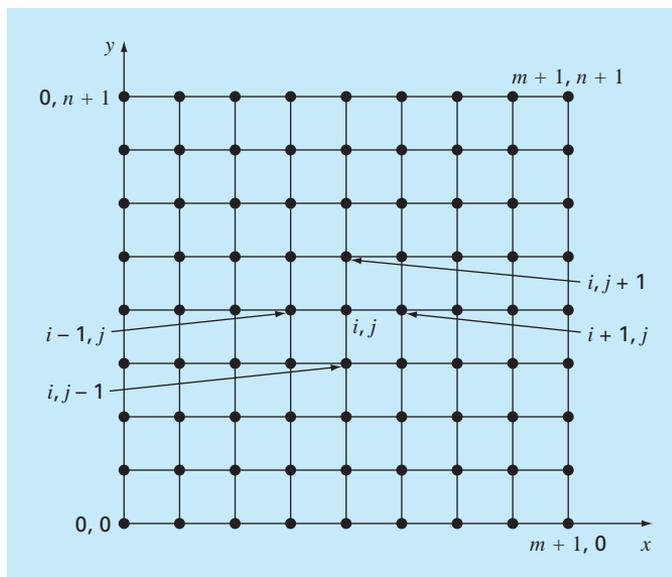
$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = f(x, y) \quad (29.7)$$

where $f(x, y)$ is a function describing the sources or sinks of heat. Equation (29.7) is referred to as the *Poisson equation*.

29.2 SOLUTION TECHNIQUE

The numerical solution of elliptic PDEs such as the Laplace equation proceeds in the reverse manner of the derivation of Eq. (29.6) from the preceding section. Recall that the derivation of Eq. (29.6) employed a balance around a discrete element to yield an algebraic difference equation characterizing heat flux for a plate. Taking the limit turned this difference equation into a differential equation [Eq. (29.3)].

For the numerical solution, finite-difference representations based on treating the plate as a grid of discrete points (Fig. 29.3) are substituted for the partial derivatives in Eq. (29.6). As described next, the PDE is transformed into an algebraic difference equation.

**FIGURE 29.3**

A grid used for the finite-difference solution of elliptic PDEs in two independent variables such as the Laplace equation.

29.2.1 The Laplacian Difference Equation

Central differences based on the grid scheme from Fig. 29.3 are (recall Fig. 23.3)

$$\frac{\partial^2 T}{\partial x^2} = \frac{T_{i+1,j} - 2T_{i,j} + T_{i-1,j}}{\Delta x^2}$$

and

$$\frac{\partial^2 T}{\partial y^2} = \frac{T_{i,j+1} - 2T_{i,j} + T_{i,j-1}}{\Delta y^2}$$

which have errors of $O[\Delta(x)^2]$ and $O[\Delta(y)^2]$, respectively. Substituting these expressions into Eq. (29.6) gives

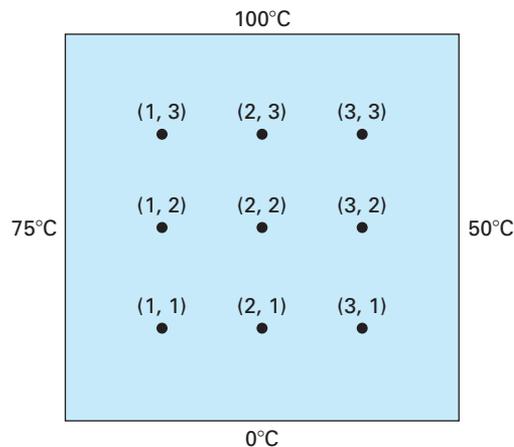
$$\frac{T_{i+1,j} - 2T_{i,j} + T_{i-1,j}}{\Delta x^2} + \frac{T_{i,j+1} - 2T_{i,j} + T_{i,j-1}}{\Delta y^2} = 0$$

For the square grid in Fig. 29.3, $\Delta x = \Delta y$, and by collection of terms, the equation becomes

$$T_{i+1,j} + T_{i-1,j} + T_{i,j+1} + T_{i,j-1} - 4T_{i,j} = 0 \quad (29.8)$$

This relationship, which holds for all interior points on the plate, is referred to as the *Laplacian difference equation*.

In addition, boundary conditions along the edges of the plate must be specified to obtain a unique solution. The simplest case is where the temperature at the boundary is set at a fixed value. This is called a *Dirichlet boundary condition*. Such is the case for Fig. 29.4,

**FIGURE 29.4**

A heated plate where boundary temperatures are held at constant levels. This case is called a Dirichlet boundary condition.

where the edges are held at constant temperatures. For the case illustrated in Fig. 29.4, a balance for node (1, 1) is, according to Eq. (29.8),

$$T_{21} + T_{01} + T_{12} + T_{10} - 4T_{11} = 0 \quad (29.9)$$

However, $T_{01} = 75$ and $T_{10} = 0$, and therefore, Eq. (29.9) can be expressed as

$$-4T_{11} + T_{12} + T_{21} = -75$$

Similar equations can be developed for the other interior points. The result is the following set of nine simultaneous equations with nine unknowns:

$$\begin{array}{cccccccc}
 4T_{11} & -T_{21} & & -T_{12} & & & & = 75 \\
 -T_{11} & +4T_{21} & -T_{31} & & -T_{22} & & & = 0 \\
 & -T_{21} & +4T_{31} & & & -T_{32} & & = 50 \\
 -T_{11} & & & +4T_{12} & -T_{22} & & -T_{13} & = 75 \\
 & -T_{21} & & -T_{12} & +4T_{22} & -T_{32} & -T_{23} & = 0 \\
 & & -T_{31} & & -T_{22} & +4T_{32} & & -T_{33} = 50 \\
 & & & -T_{12} & & & +4T_{13} & -T_{23} = 175 \\
 & & & & -T_{22} & & -T_{13} & +4T_{23} & -T_{33} = 100 \\
 & & & & & -T_{32} & & -T_{23} & +4T_{33} = 150
 \end{array} \quad (29.10)$$

29.2.2 The Liebmann Method

Most numerical solutions of the Laplace equation involve systems that are much larger than Eq. (29.10). For example, a 10-by-10 grid involves 100 linear algebraic equations. Solution techniques for these types of equations were discussed in Part Three.

Notice that there are a maximum of five unknown terms per line in Eq. (29.10). For larger-sized grids, this means that a significant number of the terms will be zero. When applied to such sparse systems, full-matrix elimination methods waste great amounts of computer memory storing these zeros. For this reason, approximate methods provide a viable approach for obtaining solutions for elliptical equations. The most commonly employed approach is *Gauss-Seidel*, which when applied to PDEs is also referred to as *Liebmann's method*. In this technique, Eq. (29.8) is expressed as

$$T_{i,j} = \frac{T_{i+1,j} + T_{i-1,j} + T_{i,j+1} + T_{i,j-1}}{4} \quad (29.11)$$

and solved iteratively for $j = 1$ to n and $i = 1$ to m . Because Eq. (29.8) is diagonally dominant, this procedure will eventually converge on a stable solution (recall Sec. 11.2.1). Overrelaxation is sometimes employed to accelerate the rate of convergence by applying the following formula after each iteration:

$$T_{i,j}^{\text{new}} = \lambda T_{i,j}^{\text{new}} + (1 - \lambda) T_{i,j}^{\text{old}} \quad (29.12)$$

where $T_{i,j}^{\text{new}}$ and $T_{i,j}^{\text{old}}$ are the values of $T_{i,j}$ from the present and the previous iteration, respectively, and λ is a weighting factor that is set between 1 and 2.

As with the conventional Gauss-Seidel method, the iterations are repeated until the absolute values of all the percent relative errors $(\varepsilon_a)_{i,j}$ fall below a prespecified stopping criterion ε_s . These percent relative errors are estimated by

$$|(\varepsilon_a)_{i,j}| = \left| \frac{T_{i,j}^{\text{new}} - T_{i,j}^{\text{old}}}{T_{i,j}^{\text{new}}} \right| 100\% \quad (29.13)$$

EXAMPLE 29.1

Temperature of a Heated Plate with Fixed Boundary Conditions

Problem Statement. Use Liebmann's method (Gauss-Seidel) to solve for the temperature of the heated plate in Fig. 29.4. Employ overrelaxation with a value of 1.5 for the weighting factor and iterate to $\varepsilon_s = 1\%$.

Solution. Equation (29.11) at $i = 1, j = 1$ is

$$T_{11} = \frac{0 + 75 + 0 + 0}{4} = 18.75$$

and applying overrelaxation yields

$$T_{11} = 1.5(18.75) + (1 - 1.5)0 = 28.125$$

For $i = 2, j = 1$,

$$T_{21} = \frac{0 + 28.125 + 0 + 0}{4} = 7.03125$$

$$T_{21} = 1.5(7.03125) + (1 - 1.5)0 = 10.54688$$

For $i = 3, j = 1$,

$$T_{31} = \frac{50 + 10.54688 + 0 + 0}{4} = 15.13672$$

$$T_{31} = 1.5(15.13672) + (1 - 1.5)0 = 22.70508$$

The computation is repeated for the other rows to give

$$\begin{array}{lll} T_{12} = 38.67188 & T_{22} = 18.45703 & T_{32} = 34.18579 \\ T_{13} = 80.12696 & T_{23} = 74.46900 & T_{33} = 96.99554 \end{array}$$

Because all the $T_{i,j}$'s are initially zero, all ε_a 's for the first iteration will be 100%.

For the second iteration the results are

$$\begin{array}{lll} T_{11} = 32.51953 & T_{21} = 22.35718 & T_{31} = 28.60108 \\ T_{12} = 57.95288 & T_{22} = 61.63333 & T_{32} = 71.86833 \\ T_{13} = 75.21973 & T_{23} = 87.95872 & T_{33} = 67.68736 \end{array}$$

The error for $T_{1,1}$ can be estimated as [Eq. (29.13)]

$$|(\varepsilon_a)_{1,1}| = \left| \frac{32.51953 - 28.12500}{32.51953} \right| 100\% = 13.5\%$$

Because this value is above the stopping criterion of 1%, the computation is continued. The ninth iteration gives the result

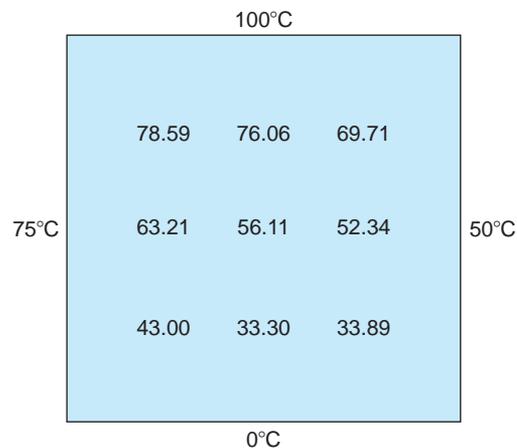
$$\begin{array}{lll} T_{11} = 43.00061 & T_{21} = 33.29755 & T_{31} = 33.88506 \\ T_{12} = 63.21152 & T_{22} = 56.11238 & T_{32} = 52.33999 \\ T_{13} = 78.58718 & T_{23} = 76.06402 & T_{33} = 69.71050 \end{array}$$

where the maximum error is 0.71%.

Figure 29.5 shows the results. As expected, a gradient is established as heat flows from high to low temperatures.

FIGURE 29.5

Temperature distribution for a heated plate subject to fixed boundary conditions.



29.2.3 Secondary Variables

Because its distribution is described by the Laplace equation, temperature is considered to be the primary variable in the heated-plate problem. For this case, as well as for other problems involving PDEs, secondary variables may also be of interest. As a matter of fact, in certain engineering contexts, the secondary variable may actually be more important.

For the heated plate, a secondary variable is the rate of heat flux across the plate's surface. This quantity can be computed from Fourier's law. Central finite-difference approximations for the first derivatives (recall Fig. 23.3) can be substituted into Eq. (29.4) to give the following values for heat flux in the x and y dimensions:

$$q_x = -k' \frac{T_{i+1,j} - T_{i-1,j}}{2 \Delta x} \quad (29.14)$$

and

$$q_y = -k' \frac{T_{i,j+1} - T_{i,j-1}}{2 \Delta y} \quad (29.15)$$

The resultant heat flux can be computed from these two quantities by

$$q_n = \sqrt{q_x^2 + q_y^2} \quad (29.16)$$

where the direction of q_n is given by

$$\theta = \tan^{-1} \left(\frac{q_y}{q_x} \right) \quad (29.17)$$

for $q_x > 0$ and

$$\theta = \tan^{-1} \left(\frac{q_y}{q_x} \right) + \pi \quad (29.18)$$

for $q_x < 0$. Recall that the angle can be expressed in degrees by multiplying it by $180^\circ/\pi$. If $q_x = 0$, θ is $\pi/2$ (90°) or $3\pi/2$ (270°), depending on whether q_y is positive or negative, respectively.

EXAMPLE 29.2

Flux Distribution for a Heated Plate

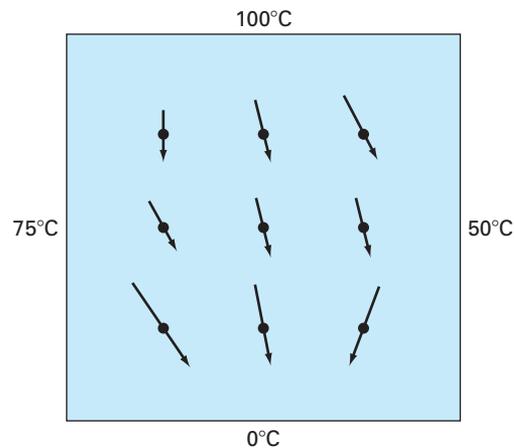
Problem Statement. Employ the results of Example 29.1 to determine the distribution of heat flux for the heated plate from Fig. 29.4. Assume that the plate is 40×40 cm and is made out of aluminum [$k' = 0.49$ cal/(s · cm · °C)].

Solution. For $i = j = 1$, Eq. (29.14) can be used to compute

$$q_x = -0.49 \frac{\text{cal}}{\text{s} \cdot \text{cm} \cdot ^\circ\text{C}} \frac{(33.29755 - 75)^\circ\text{C}}{2(10 \text{ cm})} = 1.022 \text{ cal}/(\text{cm}^2 \cdot \text{s})$$

and [Eq. (29.15)]

$$q_y = -0.49 \frac{\text{cal}}{\text{s} \cdot \text{cm} \cdot ^\circ\text{C}} \frac{(63.21152 - 0)^\circ\text{C}}{2(10 \text{ cm})} = -1.549 \text{ cal}/(\text{cm}^2 \cdot \text{s})$$

**FIGURE 29.6**

Heat flux for a plate subject to fixed boundary temperatures. Note that the lengths of the arrows are proportional to the magnitude of the flux.

The resultant flux can be computed with Eq. (29.16):

$$q_n = \sqrt{(1.022)^2 + (-1.549)^2} = 1.856 \text{ cal}/(\text{cm}^2 \cdot \text{s})$$

and the angle of its trajectory by Eq. (29.17)

$$\theta = \tan^{-1} \left(\frac{-1.549}{1.022} \right) = -0.98758 \times \frac{180^\circ}{\pi} = -56.584^\circ$$

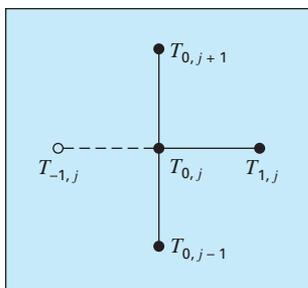
Thus, at this point, the heat flux is directed down and to the right. Values at the other grid points can be computed; the results are displayed in Fig. 29.6.

29.3 BOUNDARY CONDITIONS

Because it is free of complicating factors, the rectangular plate with fixed boundary conditions has been an ideal context for showing how elliptic PDEs can be solved numerically. We will now elaborate on other issues that will expand our capabilities to address more realistic problems. These involve boundaries at which the derivative is specified and boundaries that are irregularly shaped.

29.3.1 Derivative Boundary Conditions

The fixed or Dirichlet boundary condition discussed to this point is but one of several types that are used with partial differential equations. A common alternative is the case where the

**FIGURE 29.7**

A boundary node $(0, j)$ on the left edge of a heated plate. To approximate the derivative normal to the edge (that is, the x derivative), an imaginary point $(-1, j)$ is located a distance Δx beyond the edge.

derivative is given. This is commonly referred to as a *Neumann boundary condition*. For the heated-plate problem, this amounts to specifying the heat flux rather than the temperature at the boundary. One example is the situation where the edge is insulated. In this case, the derivative is zero. This conclusion is drawn directly from Eq. (29.4) because insulating a boundary means that the heat flux (and consequently the gradient) must be zero. Another example would be where heat is lost across the edge by predictable mechanisms such as radiation or convection.

Figure 29.7 depicts a node $(0, j)$ at the left edge of a heated plate. Applying Eq. (29.8) at the point gives

$$T_{1,j} + T_{-1,j} + T_{0,j+1} + T_{0,j-1} - 4T_{0,j} = 0 \quad (29.19)$$

Notice that an imaginary point $(-1, j)$ lying outside the plate is required for this equation. Although this exterior fictitious point might seem to represent a problem, it actually serves as the vehicle for incorporating the derivative boundary condition into the problem. This is done by representing the first derivative in the x dimension at $(0, j)$ by the finite divided difference

$$\frac{\partial T}{\partial x} \cong \frac{T_{1,j} - T_{-1,j}}{2 \Delta x}$$

which can be solved for

$$T_{-1,j} = T_{1,j} - 2 \Delta x \frac{\partial T}{\partial x}$$

Now we have a relationship for $T_{-1,j}$ that actually includes the derivative. It can be substituted into Eq. (29.19) to give

$$2T_{1,j} - 2 \Delta x \frac{\partial T}{\partial x} + T_{0,j+1} + T_{0,j-1} - 4T_{0,j} = 0 \quad (29.20)$$

Thus, we have incorporated the derivative into the balance.

Similar relationships can be developed for derivative boundary conditions at the other edges. The following example shows how this is done for the heated plate.

EXAMPLE 29.3**Heated Plate with an Insulated Edge**

Problem Statement. Repeat the same problem as in Example 29.1, but with the lower edge insulated.

Solution. The general equation to characterize a derivative at the lower edge (that is, at $j = 0$) of a heated plate is

$$T_{i+1,0} + T_{i-1,0} + 2T_{i,1} - 2 \Delta y \frac{\partial T}{\partial y} - 4T_{i,0} = 0$$

For an insulated edge, the derivative is zero and the equation becomes

$$T_{i+1,0} + T_{i-1,0} + 2T_{i,1} - 4T_{i,0} = 0$$

These results and computed fluxes (for the same parameters as in Example 29.2) are displayed in Fig. 29.8. Note that, because the lower edge is insulated, the plate's temperature is higher than for Fig. 29.5, where the bottom edge temperature is fixed at zero. In addition, the heat flow (in contrast to Fig. 29.6) is now deflected to the right and moves parallel to the insulated wall.

29.3.2 Irregular Boundaries

Although the rectangular plate from Fig. 29.4 has served well to illustrate the fundamental aspects of solving elliptic PDEs, many engineering problems do not exhibit such an idealized geometry. For example, a great many systems have irregular boundaries (Fig. 29.9).

Figure 29.9 is a system that can serve to illustrate how nonrectangular boundaries can be handled. As depicted, the plate's lower left boundary is circular. Notice that we have affixed parameters— α_1 , α_2 , β_1 , β_2 —to each of the lengths surrounding the node. Of course, for the plate depicted in Fig. 29.9, $\alpha_2 = \beta_2 = 1$. However, we will retain these parameters throughout the following derivation so that the resulting equation is generally applicable to any irregular boundary—not just one on the lower left-hand corner of a heated plate. The first derivatives in the x dimension can be approximated as

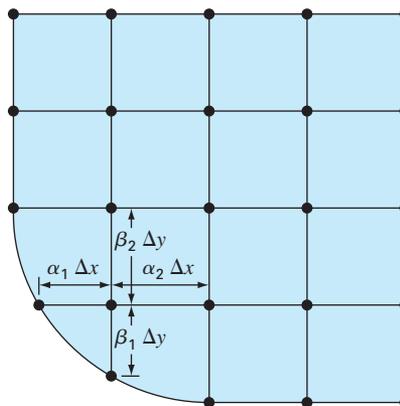
$$\left(\frac{\partial T}{\partial x}\right)_{i-1,i} \cong \frac{T_{i,j} - T_{i-1,j}}{\alpha_1 \Delta x} \quad (29.21)$$

and

$$\left(\frac{\partial T}{\partial x}\right)_{i,i+1} \cong \frac{T_{i+1,j} - T_{i,j}}{\alpha_2 \Delta x} \quad (29.22)$$

FIGURE 29.9

A grid for a heated plate with an irregularly shaped boundary. Note how weighting coefficients are used to account for the nonuniform spacing in the vicinity of the nonrectangular boundary.



The second derivatives can be developed from these first derivatives. For the x dimension, the second derivative is

$$\frac{\partial^2 T}{\partial x^2} = \frac{\partial}{\partial x} \left(\frac{\partial T}{\partial x} \right) = \frac{\left(\frac{\partial T}{\partial x} \right)_{i,i+1} - \left(\frac{\partial T}{\partial x} \right)_{i-1,i}}{\alpha_1 \Delta x + \alpha_2 \Delta x} \quad (29.23)$$

Substituting Eqs. (29.21) and (29.22) into (29.23) gives

$$\frac{\partial^2 T}{\partial x^2} = 2 \frac{\frac{T_{i-1,j} - T_{i,j}}{\alpha_1 \Delta x} - \frac{T_{i+1,j} - T_{i,j}}{\alpha_2 \Delta x}}{\alpha_1 \Delta x + \alpha_2 \Delta x}$$

Collecting terms yields

$$\frac{\partial^2 T}{\partial x^2} = \frac{2}{\Delta x^2} \left[\frac{T_{i-1,j} - T_{i,j}}{\alpha_1(\alpha_1 + \alpha_2)} + \frac{T_{i+1,j} - T_{i,j}}{\alpha_2(\alpha_1 + \alpha_2)} \right]$$

A similar equation can be developed in the y dimension:

$$\frac{\partial^2 T}{\partial y^2} = \frac{2}{\Delta y^2} \left[\frac{T_{i,j-1} - T_{i,j}}{\beta_1(\beta_1 + \beta_2)} + \frac{T_{i,j+1} - T_{i,j}}{\beta_2(\beta_1 + \beta_2)} \right]$$

Substituting these equations in Eq. (29.6) yields

$$\begin{aligned} & \frac{2}{\Delta x^2} \left[\frac{T_{i-1,j} - T_{i,j}}{\alpha_1(\alpha_1 + \alpha_2)} + \frac{T_{i+1,j} - T_{i,j}}{\alpha_2(\alpha_1 + \alpha_2)} \right] \\ & + \frac{2}{\Delta y^2} \left[\frac{T_{i,j-1} - T_{i,j}}{\beta_1(\beta_1 + \beta_2)} + \frac{T_{i,j+1} - T_{i,j}}{\beta_2(\beta_1 + \beta_2)} \right] = 0 \end{aligned} \quad (29.24)$$

As illustrated in the following example, Eq. (29.24) can be applied to any node that lies adjacent to an irregular, Dirichlet-type boundary.

EXAMPLE 29.4

Heated Plate with an Irregular Boundary

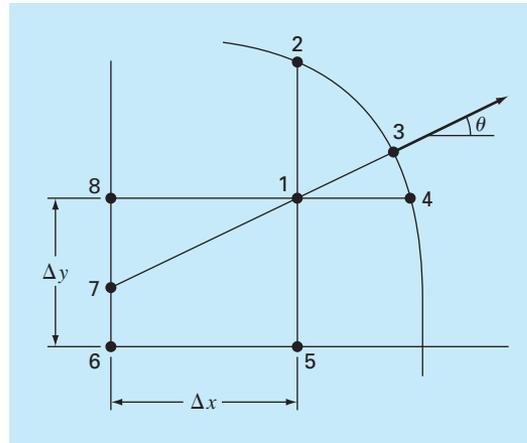
Problem Statement. Repeat the same problem as in Example 29.1, but with the lower edge as depicted in Fig. 29.9.

Solution. For the case in Fig. 29.9, $\Delta x = \Delta y$, $\alpha_1 = \beta_1 = 0.732$, and $\alpha_2 = \beta_2 = 1$. Substituting these values into Eq. (29.24) yields the following balance for node (1, 1):

$$\begin{aligned} & 0.788675(T_{01} - T_{11}) + 0.57735(T_{21} - T_{11}) \\ & + 0.788675(T_{10} - T_{11}) + 0.57735(T_{12} - T_{11}) = 0 \end{aligned}$$

Collecting terms, we can express this equation as

$$-4T_{11} + 0.8453T_{21} + 0.8453T_{12} = -1.1547T_{01} - 1.1547T_{10}$$

**FIGURE 29.11**

A curved boundary where the normal gradient is specified.

Derivative conditions for irregularly shaped boundaries are more difficult to formulate. Figure 29.11 shows a point near an irregular boundary where the normal derivative is specified.

The normal derivative at node 3 can be approximated by the gradient between nodes 1 and 7,

$$\left. \frac{\partial T}{\partial \eta} \right|_3 = \frac{T_1 - T_7}{L_{17}} \quad (29.25)$$

When θ is less than 45° as shown, the distance from node 7 to 8 is $\Delta x \tan \theta$, and linear interpolation can be used to estimate

$$T_7 = T_8 + (T_6 - T_8) \frac{\Delta x \tan \theta}{\Delta y}$$

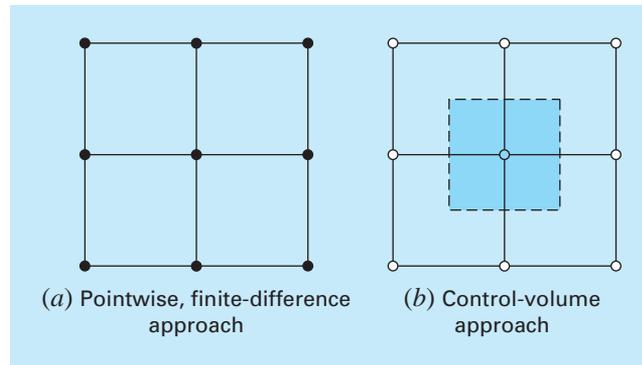
The length L_{17} is equal to $\Delta x / \cos \theta$. This length, along with the approximation for T_7 , can be substituted into Eq. (29.25) to give

$$T_1 = \left(\frac{\Delta x}{\cos \theta} \right) \left. \frac{\partial T}{\partial \eta} \right|_3 + T_6 \frac{\Delta x \tan \theta}{\Delta y} + T_8 \left(1 - \frac{\Delta x \tan \theta}{\Delta y} \right) \quad (29.26)$$

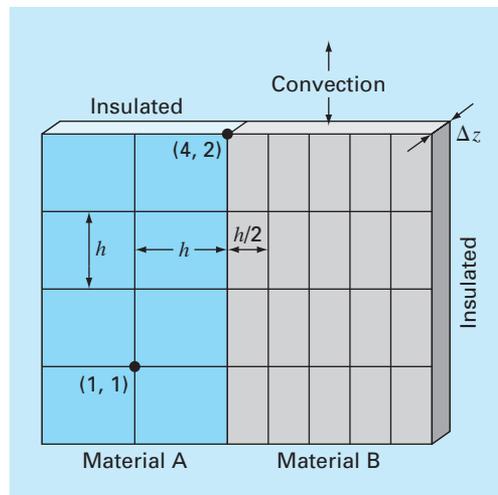
Such an equation provides a means for incorporating the normal gradient into the finite-difference approach. For cases where θ is greater than 45° , a different equation would be used. The determination of this formula will be left as a homework exercise.

29.4 THE CONTROL-VOLUME APPROACH

To summarize, the finite-difference or Taylor series approach divides the continuum into nodes (Fig. 29.12a). The underlying partial differential equation is written for each of these nodes. Finite-difference approximations are then substituted for the derivatives to convert the equations to an algebraic form.

**FIGURE 29.12**

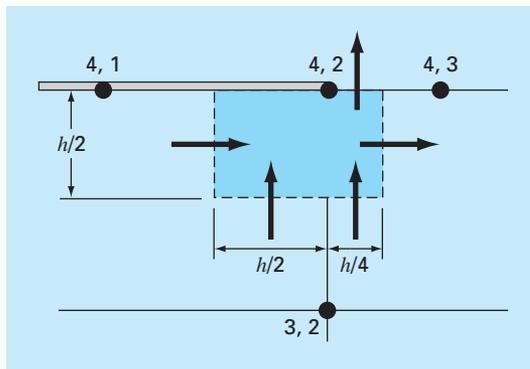
Two different perspectives for developing approximate solutions of PDEs: (a) finite-difference and (b) control volume.

**FIGURE 29.13**

A heated plate with unequal grid spacing, two materials, and mixed boundary conditions.

Such an approach is quite simple and straightforward for orthogonal (that is, rectangular) grids and constant coefficients. However, the approach becomes a more difficult endeavor for derivative conditions on irregularly shaped boundaries.

Figure 29.13 is an example of a system where additional difficulties arise. This plate is made of two different materials and has unequal grid spacing. In addition, half of its top edge is subject to convective heat transfer, whereas half is insulated. Developing equations for node (4, 2) would require some additional derivation beyond the approaches developed to this point.

**FIGURE 29.14**

A control volume for node (4, 2) with arrows indicating heat transfer through the boundaries.

The *control-volume approach* (also called the *volume-integral approach*) offers an alternative way to numerically approximate PDEs that is especially useful for cases such as Fig. 29.13. As in Fig. 29.12b, the approach resembles the point-wise approach in that points are determined across the domain. However, rather than approximating the PDE at a point, the approximation is applied to a volume surrounding the point. For an orthogonal grid, the volume is formed by the perpendicular lines through the midpoint of each line joining adjacent nodes. A heat balance can then be developed for each volume in a fashion similar to Eq. (29.1).

As an example, we will apply the control-volume approach to node (4, 2). First, the volume is defined by bisecting the lines joining the nodes. As in Fig. 29.14, the volume has conductive heat transfer through its left, right, and lower boundaries and convective heat transfer through half of its upper boundary. Notice that the transfer through the lower boundary involves both materials.

A steady-state heat balance for the volume can be written in qualitative terms as

$$0 = \left(\begin{array}{c} \text{left-side} \\ \text{conduction} \end{array} \right) - \left(\begin{array}{c} \text{right-side} \\ \text{conduction} \end{array} \right) + \left(\begin{array}{c} \text{lower conduction} \\ \text{material "a"} \end{array} \right) \\ + \left(\begin{array}{c} \text{lower conduction} \\ \text{material "b"} \end{array} \right) - \left(\begin{array}{c} \text{upper} \\ \text{convection} \end{array} \right) \quad (29.27)$$

Now the conduction flux rate can be represented by the finite-difference version of Fourier's law. For example, for the left-side conduction gain, it would be

$$q = -k'_a \frac{T_{42} - T_{41}}{h}$$

where q has units of cal/cm²/s. This flux rate must be then multiplied by the area across which it enters ($\Delta z \times h/2$) to give the rate of heat entering the volume per unit time,

$$Q = -k'_a \frac{T_{42} - T_{41}}{h} \frac{h}{2} \Delta z$$

where Q has units of cal/s.

The heat flux due to convection can be formulated as

$$q = h_c(T_a - T_{42})$$

where h_c = a heat convection coefficient [cal/(s · cm² · °C)] and T_a = the air temperature (°C). Again, multiplication by the proper area yields the rate of heat flow per time,

$$Q = h_c(T_a - T_{42}) \frac{h}{4} \Delta z$$

The other transfers can be developed in a similar fashion and substituted into Eq. (29.27) to yield

$$0 = -k'_a \frac{T_{42} - T_{41}}{h} \frac{h}{2} \Delta z + k'_b \frac{T_{43} - T_{42}}{h/2} \frac{h}{2} \Delta z$$

(left-side conduction) (right-side conduction)

$$-k'_a \frac{T_{42} - T_{32}}{h} \frac{h}{2} \Delta z - k'_b \frac{T_{42} - T_{32}}{h} \frac{h}{4} \Delta z + h_c(T_a - T_{42}) \frac{h}{4} \Delta z$$

(lower conduction material "a") (lower conduction material "b") (upper convection)

Parameter values can then be substituted to yield the final heat balance equation. For example, if $\Delta z = 0.5$ cm, $h = 10$ cm, $k'_a = 0.3$ cal/(s · cm · °C), $k'_b = 0.5$ cal/(s · cm · °C), and $h_c = 0.1$ cal/(s · cm² · °C), the equation becomes

$$0.5875T_{42} - 0.075T_{41} - 0.25T_{43} - 0.1375T_{32} = 2.5$$

To make the equation comparable to the standard Laplacian, this equation can be multiplied by 4/0.5875 so that the coefficient of the base node has a coefficient of 4,

$$4T_{42} - 0.510638T_{41} - 1.702128T_{43} - 0.93617T_{32} = 17.02128$$

For the standard cases covered to this point, the control-volume and pointwise finite-difference approaches yield identical results. For example, for node (1, 1) in Fig. 29.13, the balance would be

$$0 = -k'_a \frac{T_{11} - T_{01}}{h} h \Delta z + k'_a \frac{T_{21} - T_{11}}{h} h \Delta z - k'_a \frac{T_{11} - T_{10}}{h} h \Delta z + k'_a \frac{T_{12} - T_{11}}{h} h \Delta z$$

which simplifies to the standard Laplacian,

$$0 = 4T_{11} - T_{01} - T_{21} - T_{12} - T_{10}$$

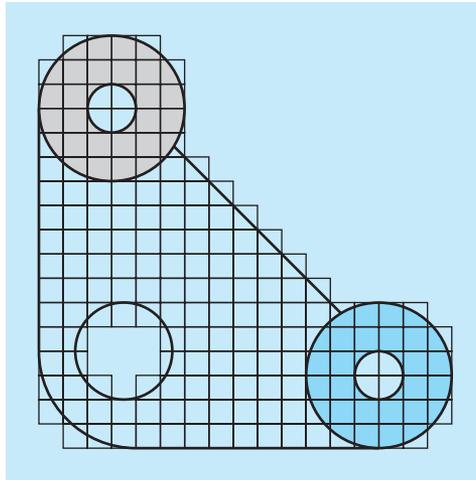
We will look at other standard cases (for example, the derivative boundary condition) and explore the control-volume approach in additional detail in the problems at the end of this chapter.

29.5 SOFTWARE TO SOLVE ELLIPTIC EQUATIONS

Modifying a computer program to include derivative boundary conditions for rectangular systems is a relatively straightforward task. It merely involves ensuring that additional equations are generated to characterize the boundary nodes at which the derivatives are specified. In addition, the code must be modified so that these equations incorporate the derivative as seen in Eq. (29.20).

FIGURE 29.15

A finite-difference grid superimposed on an irregularly shaped gasket.



Developing general software to characterize systems with irregular boundaries is a much more difficult proposition. For example, a fairly involved algorithm would be required to model the simple gasket depicted in Fig. 29.15. This would involve two major modifications. First, a scheme would have to be developed to conveniently input the configuration of the nodes and to identify which were at the boundary. Second, an algorithm would be required to generate the proper simultaneous equations on the basis of the input information. The net result is that general software for solving elliptic (and for that matter, all) PDEs is relatively complicated.

One method used to simplify such efforts is to require a very fine grid. For such cases, it is often assumed that the closest node serves as the boundary point. In this way, the analysis does not have to consider the weighting parameters from Sec. 29.3.2. Although this introduces some error, the use of a sufficiently fine mesh can make the resulting discrepancy negligible. However, this involves a trade-off due to the computational burden introduced by the increased number of simultaneous equations.

As a consequence of these considerations, numerical analysts have developed alternative approaches that differ radically from finite-difference methods. Although these finite-element methods are more conceptually difficult, they can much more easily accommodate irregular boundaries. We will turn to these methods in Chap. 31. Before doing this, however, we will first describe finite-difference approaches for another category of PDEs—parabolic equations.

PROBLEMS

29.1 Use Liebmann's method to solve for the temperature of the square heated plate in Fig. 29.4, but with the upper boundary condition increased to 150°C and the left boundary insulated. Use a relaxation factor of 1.2 and iterate to $\varepsilon_s = 1\%$.

29.2 Use Liebmann's method to solve for the temperature of the square heated plate in Fig. 29.4, but with the upper boundary con-

dition increased to 120°C and the left boundary decreased to 60°C . Use a relaxation factor of 1.2 and iterate to $\varepsilon_s = 1\%$.

29.3 Compute the fluxes for Prob. 29.2 using the parameters from Example 29.3.

29.4 Repeat Example 29.1, but use 49 interior nodes (that is, $\Delta x = \Delta y = 5\text{ cm}$).

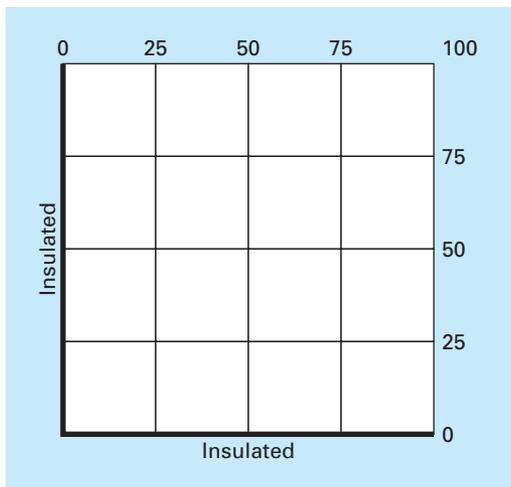


Figure P29.8

29.5 Repeat Prob. 29.4, but for the case where the lower edge is insulated.

29.6 Repeat Examples 29.1 and 29.3, but for the case where the flux at the lower edge is directed downward with a value of $2 \text{ cal/cm}^2 \cdot \text{s}$.

29.7 Repeat Example 29.4 for the case where both the lower left and the upper right corners are rounded in the same fashion as the lower left corner of Fig. 29.9. Note that all boundary temperatures on the upper and right sides are fixed at 100°C and all on the lower and left sides are fixed at 50°C .

29.8 With the exception of the boundary conditions, the plate in Fig. P29.8 has the exact same characteristics as the plate used in Examples 23.1 through 23.4. Simulate both the temperatures and fluxes for the plate.

29.9 Write equations for the darkened nodes in the grid in Fig. P29.9. Note that all units are cgs. The coefficient of thermal conductivity for the plate is $0.75 \text{ cal}/(\text{s} \cdot \text{cm} \cdot ^\circ\text{C})$, the convection coefficient is $h_c = 0.015 \text{ cal}/(\text{cm}^2 \cdot \text{C} \cdot \text{s})$, and the thickness of the plate is 0.5 cm .

29.10 Write equations for the darkened nodes in the grid in Fig. P29.10. Note that all units are cgs. The convection coefficient is $h_c = 0.01 \text{ cal}/(\text{cm}^2 \cdot \text{C} \cdot \text{s})$ and the thickness of the plate is 2 cm .

29.11 Apply the control volume approach to develop the equation for node $(0, j)$ in Fig. 29.7.

29.12 Derive an equation like Eq. (29.26) for the case where θ is greater than 45° for Fig. 29.11.

29.13 Develop a user-friendly computer program to implement Liebmann's method for a rectangular plate with Dirichlet boundary conditions. Design the program so that it can compute both

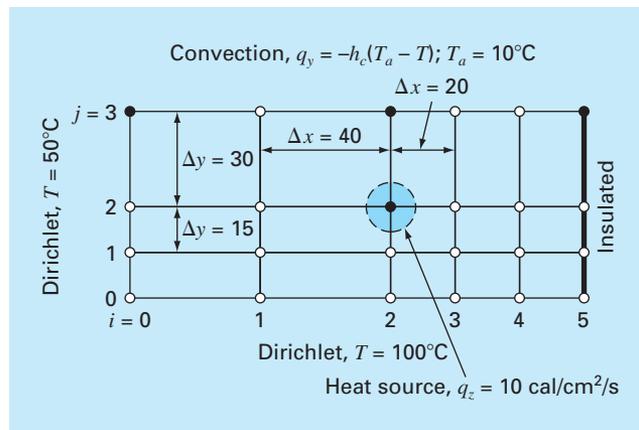


Figure P29.9

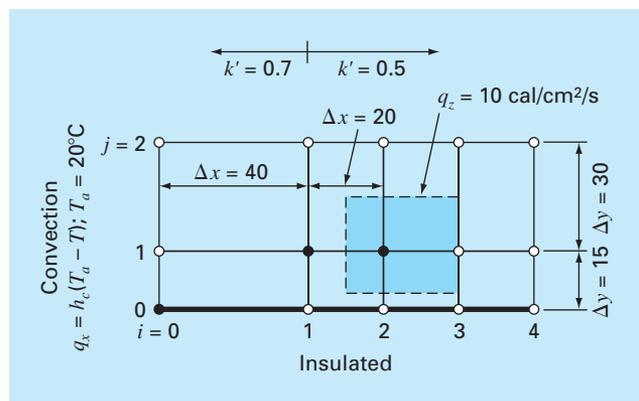


Figure P29.10

temperature and flux. Test the program by duplicating the results of Examples 29.1 and 29.2.

29.14 Employ the program from Prob. 29.13 to solve Probs. 29.2 and 29.3.

29.15 Employ the program from Prob 29.13 to solve Prob. 29.4.

29.16 Use the control-volume approach and derive the node equation for node $(2, 2)$ in Fig. 29.13 and include a heat source at this point. Use the following values for the constants: $\Delta z = 0.25 \text{ cm}$, $h = 10 \text{ cm}$, $k_A = 0.25 \text{ W/cm} \cdot \text{C}$, and $k_B = 0.45 \text{ W/cm} \cdot \text{C}$. The heat source comes only from material A at the rate of $= 6 \text{ W/cm}^3$.

29.17 Calculate heat flux (W/cm^2) for node $(2, 2)$ in Fig. 29.13 using finite-difference approximations for the temperature gradients

at this node. Calculate the flux in the horizontal direction in materials A and B , and determine if these two fluxes should be equal. Also, calculate the vertical flux in materials A and B . Should these two fluxes be equal? Use the following values for the constants: $\Delta z = 0.5$ cm, $h = 10$ cm, $k_A = 0.25$ W/cm \cdot C, $k_B = 0.45$ W/cm \cdot C, and nodal temperatures: $T_{22} = 51.6^\circ\text{C}$, $T_{21} = 74.2^\circ\text{C}$, $T_{23} = 45.3^\circ\text{C}$, $T_{32} = 38.6^\circ\text{C}$, and $T_{12} = 87.4^\circ\text{C}$.

29.18 Compute the temperature distribution for the L-shaped plate in Fig. P29.18.

29.19 The Poisson equation can be written in three dimensions as

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} = f(x, y, z)$$

Solve for the distribution of temperature within a unit (1×1) cube with zero boundary conditions and $f = -10$. Employ $\Delta x = \Delta y = \Delta z = 1/6$.

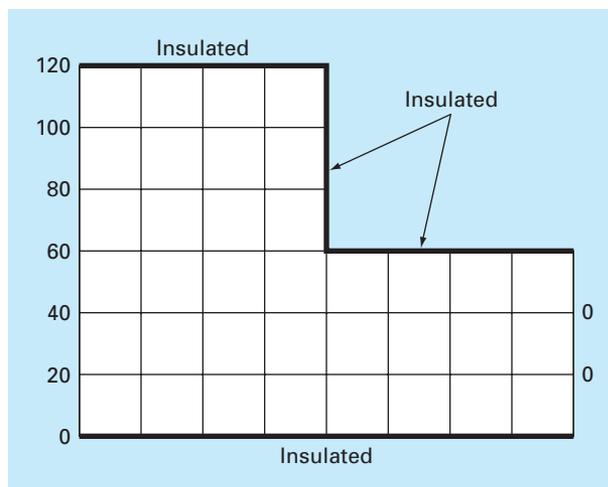


Figure P29.18

Finite Difference: Parabolic Equations

Chapter 29 dealt with steady-state PDEs. We now turn to the parabolic equations that are employed to characterize time-variable problems. In the latter part of this chapter, we will illustrate how this is done in two spatial dimensions for the heated plate. Before doing this, we will first show how the simpler one-dimensional case is approached.

30.1 THE HEAT-CONDUCTION EQUATION

In a fashion similar to the derivation of the Laplace equation [Eq. (29.6)], conservation of heat can be used to develop a heat balance for the differential element in the long, thin insulated rod shown in Fig. 30.1. However, rather than examine the steady-state case, the present balance also considers the amount of heat stored in the element over a unit time period Δt . Thus, the balance is in the form, inputs – outputs = storage, or

$$q(x) \Delta y \Delta z \Delta t - q(x + \Delta x) \Delta y \Delta z \Delta t = \Delta x \Delta y \Delta z \rho C \Delta T$$

Dividing by the volume of the element ($= \Delta x \Delta y \Delta z$) and Δt gives

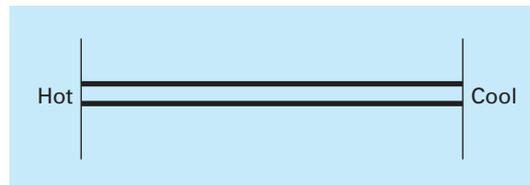
$$\frac{q(x) - q(x + \Delta x)}{\Delta x} = \rho C \frac{\Delta T}{\Delta t}$$

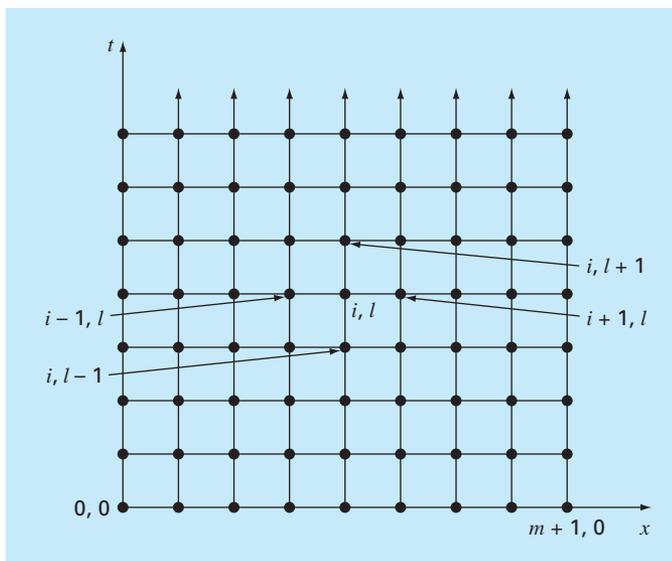
Taking the limit yields

$$-\frac{\partial q}{\partial x} = \rho C \frac{\partial T}{\partial t}$$

FIGURE 30.1

A thin rod, insulated at all points except at its ends.



**FIGURE 30.2**

A grid used for the finite-difference solution of parabolic PDEs in two independent variables such as the heat-conduction equation. Note how, in contrast to Fig. 29.3, this grid is open-ended in the temporal dimension.

Substituting Fourier's law of heat conduction [Eq. (29.4)] results in

$$k \frac{\partial^2 T}{\partial x^2} = \frac{\partial T}{\partial t} \quad (30.1)$$

which is the *heat-conduction equation*.

Just as with elliptic PDEs, parabolic equations can be solved by substituting finite divided differences for the partial derivatives. However, in contrast to elliptic PDEs, we must now consider changes in time as well as in space. Whereas elliptic equations were bounded in all relevant dimensions, parabolic PDEs are temporally open-ended (Fig. 30.2). Because of their time-variable nature, solutions to these equations involve a number of new issues, notably stability. This, as well as other aspects of parabolic PDEs, will be examined in the following sections as we present two fundamental solution approaches—explicit and implicit schemes.

30.2 EXPLICIT METHODS

The heat-conduction equation requires approximations for the second derivative in space and the first derivative in time. The former is represented in the same fashion as for the Laplace equation by a centered finite-divided difference:

$$\frac{\partial^2 T}{\partial x^2} = \frac{T_{i+1}^l - 2T_i^l + T_{i-1}^l}{\Delta x^2} \quad (30.2)$$

which has an error (recall Fig. 23.3) of $O[(\Delta x)^2]$. Notice the slight change in notation of the superscripts is used to denote time. This is done so that a second subscript can be used to designate a second spatial dimension when the approach is expanded to two spatial dimensions.

A forward finite-divided difference is used to approximate the time derivative

$$\frac{\partial T}{\partial t} = \frac{T_i^{l+1} - T_i^l}{\Delta t} \quad (30.3)$$

which has an error (recall Fig. 23.1) of $O(\Delta t)$.

Substituting Eqs. (30.2) and (30.3) into Eq. (30.1) yields

$$k \frac{T_{i+1}^l - 2T_i^l + T_{i-1}^l}{(\Delta x)^2} = \frac{T_i^{l+1} - T_i^l}{\Delta t} \quad (30.4)$$

which can be solved for

$$T_i^{l+1} = T_i^l + \lambda(T_{i+1}^l - 2T_i^l + T_{i-1}^l) \quad (30.5)$$

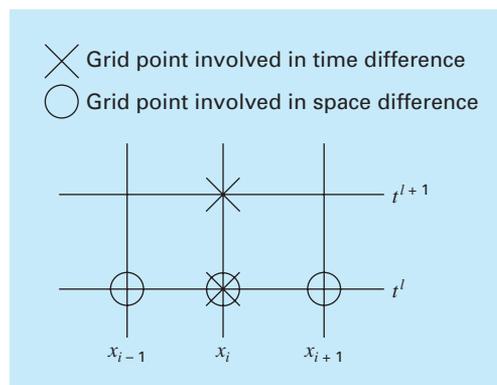
where $\lambda = k \Delta t / (\Delta x)^2$.

This equation can be written for all the interior nodes on the rod. It then provides an explicit means to compute values at each node for a future time based on the present values at the node and its neighbors. Notice that this approach is actually a manifestation of Euler's method for solving systems of ODEs. That is, if we know the temperature distribution as a function of position at an initial time, we can compute the distribution at a future time based on Eq. (30.5).

A computational molecule for the explicit method is depicted in Fig. 30.3, showing the nodes that constitute the spatial and temporal approximations. This molecule can be contrasted with others in this chapter to illustrate the differences between approaches.

FIGURE 30.3

A computational molecule for the explicit form.



EXAMPLE 30.1

Explicit Solution of the One-Dimensional Heat-Conduction Equation

Problem Statement. Use the explicit method to solve for the temperature distribution of a long, thin rod with a length of 10 cm and the following values: $k' = 0.49$ cal/(s · cm · °C), $\Delta x = 2$ cm, and $\Delta t = 0.1$ s. At $t = 0$, the temperature of the rod is zero and the boundary conditions are fixed for all times at $T(0) = 100^\circ\text{C}$ and $T(10) = 50^\circ\text{C}$. Note that the rod is aluminum with $C = 0.2174$ cal/(g · °C) and $\rho = 2.7$ g/cm³. Therefore, $k = 0.49/(2.7 \cdot 0.2174) = 0.835$ cm²/s and $\lambda = 0.835(0.1)/(2)^2 = 0.020875$.

Solution. Applying Eq. (30.5) gives the following value at $t = 0.1$ s for the node at $x = 2$ cm:

$$T_1^1 = 0 + 0.020875[0 - 2(0) + 100] = 2.0875$$

At the other interior points, $x = 4, 6,$ and 8 cm, the results are

$$T_2^1 = 0 + 0.020875[0 - 2(0) + 0] = 0$$

$$T_3^1 = 0 + 0.020875[0 - 2(0) + 0] = 0$$

$$T_4^1 = 0 + 0.020875[50 - 2(0) + 0] = 1.0438$$

At $t = 0.2$ s, the values at the four interior nodes are computed as

$$T_1^2 = 2.0875 + 0.020875[0 - 2(2.0875) + 100] = 4.0878$$

$$T_2^2 = 0 + 0.020875[0 - 2(0) + 2.0875] = 0.043577$$

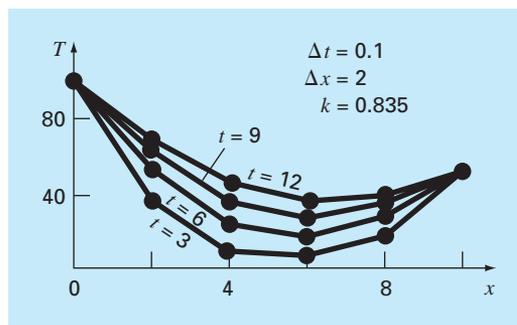
$$T_3^2 = 0 + 0.020875[1.0438 - 2(0) + 0] = 0.021788$$

$$T_4^2 = 1.0438 + 0.020875[50 - 2(1.0438) + 0] = 2.0439$$

The computation is continued, and the results at 3-s intervals are depicted in Fig. 30.4. The general rise in temperature with time indicates that the computation captures the diffusion of heat from the boundaries into the bar.

FIGURE 30.4

Temperature distribution in a long, thin rod as computed with the explicit method described in Sec. 30.2.



30.2.1 Convergence and Stability

Convergence means that as Δx and Δt approach zero, the results of the finite-difference technique approach the true solution. *Stability* means that errors at any stage of the computation are not amplified but are attenuated as the computation progresses. It can be shown (Carnahan et al., 1969) that the explicit method is both convergent and stable if $\lambda \leq 1/2$, or

$$\Delta t \leq \frac{1}{2} \frac{\Delta x^2}{k} \quad (30.6)$$

In addition, it should be noted that setting $\lambda \leq 1/2$ could result in a solution in which errors do not grow, but oscillate. Setting $\lambda \leq 1/4$ ensures that the solution will not oscillate. It is also known that setting $\lambda = 1/6$ tends to minimize truncation error (Carnahan et al., 1969).

Figure 30.5 is an example of instability caused by violating Eq. (30.6). This plot is for the same case as in Example 30.1 but with $\lambda = 0.735$, which is considerably greater than 0.5. As in Fig. 30.5, the solution undergoes progressively increasing oscillations. This situation will continue to deteriorate as the computation continues.

Although satisfaction of Eq. (30.6) will alleviate the instabilities of the sort manifested in Fig. 30.5, it also places a strong limitation on the explicit method. For example, suppose that Δx is halved to improve the approximation of the spatial second derivative. According to Eq. (30.6), the time step must be quartered to maintain convergence and stability. Thus, to perform comparable computations, the time steps must be increased by a factor of 4. Furthermore, the computation for each of these time steps will take twice as long because halving Δx doubles the total number of nodes for which equations must be written. Consequently, for the one-dimensional case, halving Δx results in an eightfold increase in the number of calculations. Thus, the computational burden may be large to attain acceptable accuracy. As will be described shortly, other techniques are available that do not suffer from such severe limitations.

30.2.2 Derivative Boundary Conditions

As was the case for elliptic PDEs (recall Sec. 29.3.1), derivative boundary conditions can be readily incorporated into parabolic equations. For a one-dimensional rod, this necessitates adding two equations to characterize the heat balance at the end nodes. For example, the node at the left end ($i = 0$) would be represented by

$$T_0^{l+1} = T_0^l + \lambda(T_1^l - 2T_0^l + T_{-1}^l)$$

Thus, an imaginary point is introduced at $i = -1$ (recall Fig. 29.7). However, as with the elliptic case, this point provides a vehicle for incorporating the derivative boundary condition into the analysis. Problem 30.2 at the end of the chapter deals with this exercise.

30.2.3 Higher-Order Temporal Approximations

The general idea of reexpressing the PDE as a system of ODEs is sometimes called the *method of lines*. Obviously, one way to improve on the Euler approach used above would be to employ a more accurate integration scheme for solving the ODEs. For example, the Heun method can be employed to obtain second-order temporal accuracy. An example of

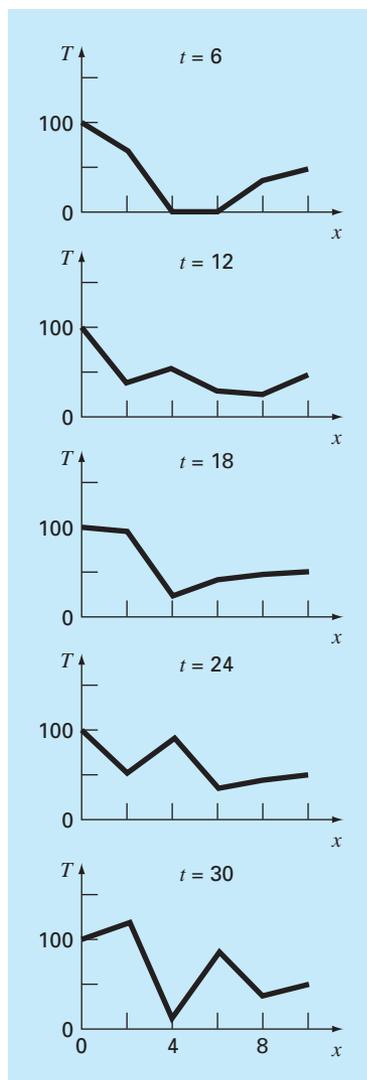


FIGURE 30.5

An illustration of instability. Solution of Example 30.1 but with $\lambda = 0.735$.

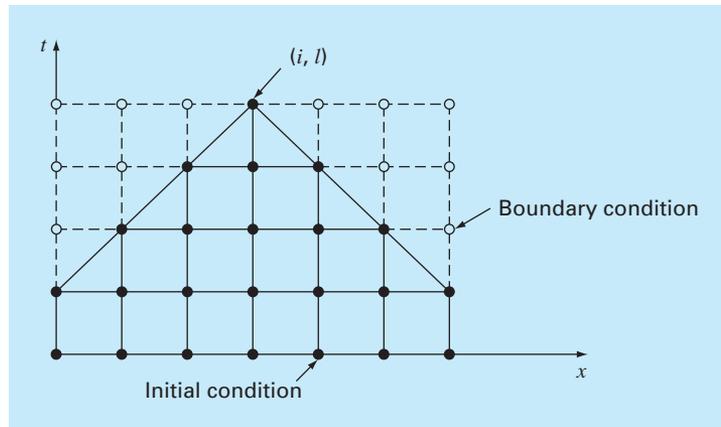
this approach is called *MacCormack's method*. This and other improved explicit methods are discussed elsewhere (for example, Hoffman, 1992).

30.3 A SIMPLE IMPLICIT METHOD

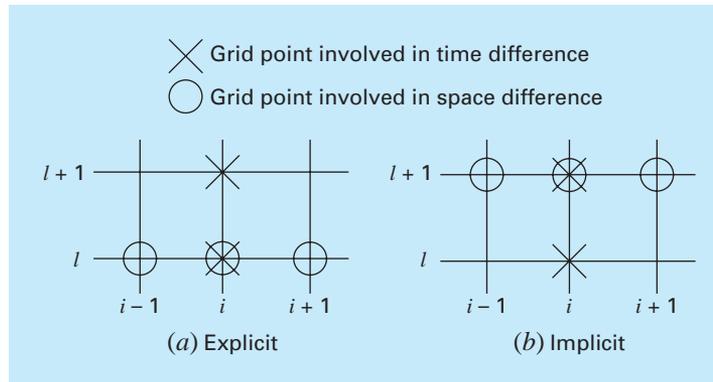
As noted previously, explicit finite-difference formulations have problems related to stability. In addition, as depicted in Fig. 30.6, they exclude information that has a bearing on the solution. Implicit methods overcome both these difficulties at the expense of somewhat more complicated algorithms.

FIGURE 30.6

Representation of the effect of other nodes on the finite-difference approximation at node (i, l) using an explicit finite-difference scheme. The shaded nodes have an influence on (i, l) , whereas the unshaded nodes, which in reality affect (i, l) , are excluded.

**FIGURE 30.7**

Computational molecules demonstrating the fundamental differences between (a) explicit and (b) implicit methods.



The fundamental difference between explicit and implicit approximations is depicted in Fig. 30.7. For the explicit form, we approximate the spatial derivative at time level l (Fig. 30.7a). Recall that when we substituted this approximation into the partial differential equation, we obtained a difference equation (30.4) with a single unknown T_i^{l+1} . Thus, we can solve “explicitly” for this unknown as in Eq. (30.5).

In implicit methods, the spatial derivative is approximated at an advanced time level $l + 1$. For example, the second derivative would be approximated by (Fig. 30.7b)

$$\frac{\partial^2 T}{\partial x^2} \cong \frac{T_{i+1}^{l+1} - 2T_i^{l+1} + T_{i-1}^{l+1}}{(\Delta x)^2} \quad (30.7)$$

which is second-order accurate. When this relationship is substituted into the original PDE, the resulting difference equation contains several unknowns. Thus, it cannot be solved

explicitly by simple algebraic rearrangement as was done in going from Eq. (30.4) to (30.5). Instead, the entire system of equations must be solved simultaneously. This is possible because, along with the boundary conditions, the implicit formulations result in a set of linear algebraic equations with the same number of unknowns. Thus, the method reduces to the solution of a set of simultaneous equations at each point in time.

To illustrate how this is done, substitute Eqs. (30.3) and (30.7) into Eq. (30.1) to give

$$k \frac{T_{i+1}^{l+1} - 2T_i^{l+1} + T_{i-1}^{l+1}}{(\Delta x)^2} = \frac{T_i^{l+1} - T_i^l}{\Delta t}$$

which can be expressed as

$$-\lambda T_{i-1}^{l+1} + (1 + 2\lambda)T_i^{l+1} - \lambda T_{i+1}^{l+1} = T_i^l \quad (30.8)$$

where $\lambda = k \Delta t / (\Delta x)^2$. This equation applies to all but the first and the last interior nodes, which must be modified to reflect the boundary conditions. For the case where the temperature levels at the ends of the rod are given, the boundary condition at the left end of the rod ($i = 0$) can be expressed as

$$T_0^{l+1} = f_0(t^{l+1}) \quad (30.9)$$

where $f_0(t^{l+1})$ is a function describing how the boundary temperature changes with time. Substituting Eq. (30.9) into Eq. (30.8) gives the difference equation for the first interior node ($i = 1$):

$$(1 + 2\lambda)T_1^{l+1} - \lambda T_2^{l+1} = T_1^l + \lambda f_0(t^{l+1}) \quad (30.10)$$

Similarly, for the last interior node ($i = m$),

$$-\lambda T_{m-1}^{l+1} + (1 + 2\lambda)T_m^{l+1} = T_m^l + \lambda f_{m+1}(t^{l+1}) \quad (30.11)$$

where $f_{m+1}(t^{l+1})$ describes the specified temperature changes at the right end of the rod ($i = m + 1$).

When Eqs. (30.8), (30.10), and (30.11) are written for all the interior nodes, the resulting set of m linear algebraic equations has m unknowns. In addition, the method has the added bonus that the system is tridiagonal. Thus, we can utilize the extremely efficient solution algorithms (recall Sec. 11.1.1) that are available for tridiagonal systems.

EXAMPLE 30.2

Simple Implicit Solution of the Heat-Conduction Equation

Problem Statement. Use the simple implicit finite-difference approximation to solve the same problem as in Example 30.1.

Solution. For the rod from Example 30.1, $\lambda = 0.020875$. Therefore, at $t = 0$, Eq. (30.10) can be written for the first interior node as

$$1.04175T_1^1 - 0.020875T_2^1 = 0 + 0.020875(100)$$

or

$$1.04175T_1^1 - 0.020875T_2^1 = 2.0875$$

In a similar fashion, Eqs. (30.8) and (30.11) can be applied to the other interior nodes. This leads to the following set of simultaneous equations:

$$\begin{bmatrix} 1.04175 & -0.020875 & & \\ -0.020875 & 1.04175 & -0.020875 & \\ & -0.020875 & 1.04175 & -0.020875 \\ & & -0.020875 & 1.04175 \end{bmatrix} \begin{Bmatrix} T_1^1 \\ T_2^1 \\ T_3^1 \\ T_4^1 \end{Bmatrix} = \begin{Bmatrix} 2.0875 \\ 0 \\ 0 \\ 1.04375 \end{Bmatrix}$$

which can be solved for the temperature at $t = 0.1$ s:

$$T_1^1 = 2.0047$$

$$T_2^1 = 0.0406$$

$$T_3^1 = 0.0209$$

$$T_4^1 = 1.0023$$

Notice how in contrast to Example 30.1, all the points have changed from the initial condition during the first time step.

To solve for the temperatures at $t = 0.2$, the right-hand-side vector must be modified to account for the results of the first step, as in

$$\begin{Bmatrix} 4.09215 \\ 0.04059 \\ 0.02090 \\ 2.04069 \end{Bmatrix}$$

The simultaneous equations can then be solved for the temperatures at $t = 0.2$ s:

$$T_1^2 = 3.9305$$

$$T_2^2 = 0.1190$$

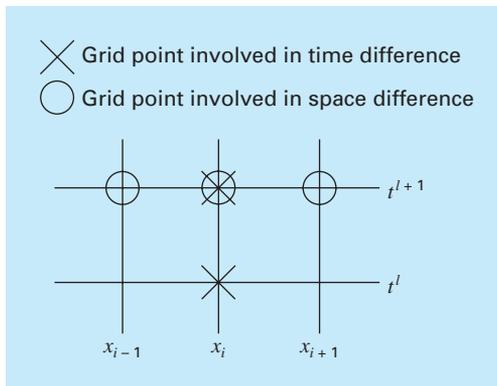
$$T_3^2 = 0.0618$$

$$T_4^2 = 1.9653$$

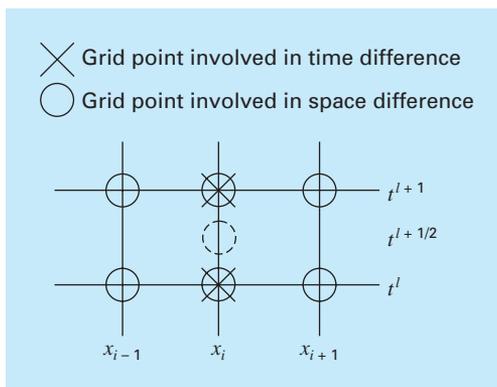
Whereas the implicit method described is stable and convergent, it has the defect that the temporal difference approximation is first-order accurate, whereas the spatial difference approximation is second-order accurate (Fig. 30.8). In the next section we present an alternative implicit method that remedies the situation.

Before proceeding, it should be mentioned that, although the simple implicit method is unconditionally stable, there is an accuracy limit to the use of large time steps. Consequently, it is not that much more efficient than the explicit approaches for most time-variable problems.

Where it does shine is for steady-state problems. Recall from Chap. 29 that a form of Gauss-Seidel (Liebmann's method) can be used to obtain steady-state solutions for elliptic equations. An alternative approach would be to run a time-variable solution until it reached a steady state. In such cases, because inaccurate intermediate results are not an issue, implicit methods allow you to employ larger time steps, and hence, can generate steady-state results in an efficient manner.

**FIGURE 30.8**

A computational molecule for the simple implicit method.

**FIGURE 30.9**

A computational molecule for the Crank-Nicolson method.

30.4 THE CRANK-NICOLSON METHOD

The *Crank-Nicolson method* provides an alternative implicit scheme that is second-order accurate in both space and time. To provide this accuracy, difference approximations are developed at the midpoint of the time increment (Fig. 30.9). To do this, the temporal first derivative can be approximated at $t^{l+1/2}$ by

$$\frac{\partial T}{\partial t} \cong \frac{T_i^{l+1} - T_i^l}{\Delta t} \quad (30.12)$$

The second derivative in space can be determined at the midpoint by averaging the difference approximations at the beginning (t^l) and at the end (t^{l+1}) of the time increment

$$\frac{\partial^2 T}{\partial x^2} \cong \frac{1}{2} \left[\frac{T_{i+1}^l - 2T_i^l + T_{i-1}^l}{(\Delta x)^2} + \frac{T_{i+1}^{l+1} - 2T_i^{l+1} + T_{i-1}^{l+1}}{(\Delta x)^2} \right] \quad (30.13)$$

Substituting Eqs. (30.12) and (30.13) into Eq. (30.1) and collecting terms gives

$$-\lambda T_{i-1}^{l+1} + 2(1 + \lambda)T_i^{l+1} - \lambda T_{i+1}^{l+1} = \lambda T_{i-1}^l + 2(1 - \lambda)T_i^l + \lambda T_{i+1}^l \quad (30.14)$$

where $\lambda = k \Delta t / (\Delta x)^2$. As was the case with the simple implicit approach, boundary conditions of $T_0^{l+1} = f_0(t^{l+1})$ and $T_{m+1}^{l+1} = f_{m+1}(t^{l+1})$ can be prescribed to derive versions of Eq. (30.14) for the first and the last interior nodes. For the first interior node

$$2(1 + \lambda)T_1^{l+1} - \lambda T_2^{l+1} = \lambda f_0(t^l) + 2(1 - \lambda)T_1^l + \lambda T_2^l + \lambda f_0(t^{l+1}) \quad (30.15)$$

and for the last interior node,

$$-\lambda T_{m-1}^{l+1} + 2(1 + \lambda)T_m^{l+1} = \lambda f_{m+1}(t^l) + 2(1 - \lambda)T_m^l + \lambda T_{m-1}^l + \lambda f_{m+1}(t^{l+1}) \quad (30.16)$$

Although Eqs. (30.14) through (30.16) are slightly more complicated than Eqs. (30.8), (30.10), and (30.11), they are also tridiagonal and, therefore, efficient to solve.

EXAMPLE 30.3

Crank-Nicolson Solution to the Heat-Conduction Equation

Problem Statement. Use the Crank-Nicolson method to solve the same problem as in Examples 30.1 and 30.2.

Solution. Equations (30.14) through (30.16) can be employed to generate the following tridiagonal set of equations:

$$\begin{bmatrix} 2.04175 & -0.020875 & & & \\ -0.020875 & 2.04175 & -0.020875 & & \\ & -0.020875 & 2.04175 & -0.020875 & \\ & & -0.020875 & 2.04175 & \\ & & & & \end{bmatrix} \begin{Bmatrix} T_1^1 \\ T_2^1 \\ T_3^1 \\ T_4^1 \end{Bmatrix} = \begin{Bmatrix} 4.175 \\ 0 \\ 0 \\ 2.0875 \end{Bmatrix}$$

which can be solved for the temperatures at $t = 0.1$ s:

$$T_1^1 = 2.0450$$

$$T_2^1 = 0.0210$$

$$T_3^1 = 0.0107$$

$$T_4^1 = 1.0225$$

To solve for the temperatures at $t = 0.2$ s, the right-hand-side vector must be changed to

$$\begin{Bmatrix} 8.1801 \\ 0.0841 \\ 0.0427 \\ 4.0901 \end{Bmatrix}$$

The simultaneous equations can then be solved for

$$T_1^2 = 4.0073$$

$$T_2^2 = 0.0826$$

$$T_3^2 = 0.0422$$

$$T_4^2 = 2.0036$$

30.4.1 Comparison of One-Dimensional Methods

Equation (30.1) can be solved analytically. For example, a solution is available for the case where the rod's temperature is initially at zero. At $t = 0$, the boundary condition at $x = L$ is instantaneously increased to a constant level of T while $T(0)$ is held at zero. For this case, the temperature can be computed by

$$T = \bar{T} \left[\frac{x}{L} + \sum_{n=0}^{\infty} \frac{2}{n\pi} (-1)^n \sin\left(\frac{n\pi x}{L}\right) \exp\left(\frac{-n^2\pi^2 kt}{L^2}\right) \right] \quad (30.17)$$

where L = total length of the rod. This equation can be employed to compute the evolution of the temperature distribution for each boundary condition. Then, the total solution can be determined by superposition.

EXAMPLE 30.4

Comparison of Analytical and Numerical Solutions

Problem Statement. Compare the analytical solution from Eq. (30.17) with numerical results obtained with the explicit, simple implicit, and Crank-Nicolson techniques. Perform the comparison for the rod employed in Examples 30.1, 30.2, and 30.3.

Solution. Recall from the previous examples that $k = 0.835 \text{ cm}^2/\text{s}$, $L = 10 \text{ cm}$, and $\Delta x = 2 \text{ cm}$. For this case, Eq. (30.17) can be used to predict that the temperature at $x = 2 \text{ cm}$, and $t = 10 \text{ s}$ would equal 64.8018. Table 30.1 presents numerical predictions of $T(2, 10)$. Notice that a range of time steps are employed. These results indicate a number of properties of the numerical methods. First, it can be seen that the explicit method is unstable for high values of λ . This instability is not manifested by either implicit approach. Second, the Crank-Nicolson method converges more rapidly as λ is decreased and provides moderately accurate results even when λ is relatively high. These outcomes are as expected because Crank-Nicolson is second-order accurate with respect to both independent variables. Finally, notice that as λ decreases, the methods seem to be converging on a value of 64.73 that is different than the analytical result of 64.80. This should not be surprising because a fixed value of $\Delta x = 2$ is used to characterize the x dimension. If both Δx and Δt were decreased as λ was decreased (that is, more spatial segments were used), the numerical solution would more closely approach the analytical result.

TABLE 30.1 Comparison of three methods of solving a parabolic PDE: the heated rod. The results shown are for temperature at $t = 10 \text{ s}$ at $x = 2 \text{ cm}$ for the rod from Examples 30.1 through 30.3. Note that the analytical solution is $T(2, 10) = 64.8018$.

Δt	λ	Explicit	Implicit	Crank-Nicolson
10	2.0875	208.75	53.01	79.77
5	1.04375	-9.13	58.49	64.79
2	0.4175	67.12	62.22	64.87
1	0.20875	65.91	63.49	64.77
0.5	0.104375	65.33	64.12	64.74
0.2	0.04175	64.97	64.49	64.73

The Crank-Nicolson method is often used for solving linear parabolic PDEs in one spatial dimension. Its advantages become even more pronounced for more complicated applications such as those involving unequally spaced meshes. Such nonuniform spacing is often advantageous where we have foreknowledge that the solution varies rapidly in local portions of the system. Further discussion of such applications and the Crank-Nicolson method in general can be found elsewhere (Ferziger, 1981; Lapidus and Pinder, 1981; Hoffman 1992).

30.5 PARABOLIC EQUATIONS IN TWO SPATIAL DIMENSIONS

The heat-conduction equation can be applied to more than one spatial dimension. For two dimensions, its form is

$$\frac{\partial T}{\partial t} = k \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right) \quad (30.18)$$

One application of this equation is to model the temperature distribution on the face of a heated plate. However, rather than characterizing its steady-state distribution, as was done in Chap. 29, Eq. (30.18) provides a means to compute the plate's temperature distribution as it changes in time.

30.5.1 Standard Explicit and Implicit Schemes

An explicit solution can be obtained by substituting finite-difference approximations of the form of Eqs. (30.2) and (30.3) into Eq. (30.18). However, as with the one-dimensional case, this approach is limited by a stringent stability criterion. For the two-dimensional case, the criterion is

$$\Delta t \leq \frac{1}{8} \frac{(\Delta x)^2 + (\Delta y)^2}{k}$$

Thus, for a uniform grid ($\Delta x = \Delta y$), $\lambda = k \Delta t / (\Delta x)^2$ must be less than or equal to 1/4. Consequently, halving the step size results in a fourfold increase in the number of nodes and a 16-fold increase in computational effort.

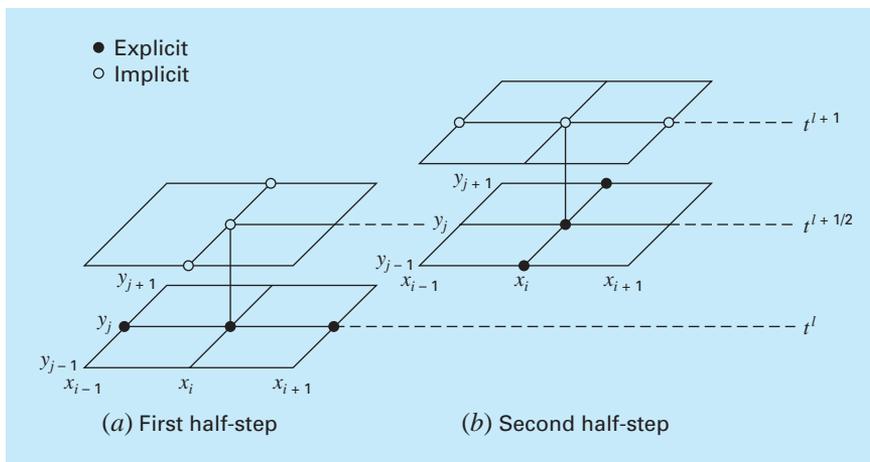
As was the case with one-dimensional systems, implicit techniques offer alternatives that guarantee stability. However, the direct application of implicit methods such as the Crank-Nicolson technique leads to the solution of $m \times n$ simultaneous equations. Additionally, when written for two or three spatial dimensions, these equations lose the valuable property of being tridiagonal. Thus, matrix storage and computation time can become exorbitantly large. The method described in the next section offers one way around this dilemma.

30.5.2 The ADI Scheme

The alternating-direction implicit, or ADI, scheme provides a means for solving parabolic equations in two spatial dimensions using tridiagonal matrices. To do this, each time

FIGURE 30.10

The two half-steps used in implementing the alternating-direction implicit scheme for solving parabolic equations in two spatial dimensions.



increment is executed in two steps (Fig. 30.10). For the first step, Eq. (30.18) is approximated by

$$\frac{T_{i,j}^{l+1/2} - T_{i,j}^l}{\Delta t/2} = k \left[\frac{T_{i+1,j}^l - 2T_{i,j}^l + T_{i-1,j}^l}{(\Delta x)^2} + \frac{T_{i,j+1}^{l+1/2} - 2T_{i,j}^{l+1/2} + T_{i,j-1}^{l+1/2}}{(\Delta y)^2} \right] \quad (30.19)$$

Thus, the approximation of $\partial^2 T / \partial x^2$ is written explicitly—that is, at the base point t^l where values of temperature are known. Consequently, only the three temperature terms in the approximation of $\partial^2 T / \partial y^2$ are unknown. For the case of a square grid ($\Delta y = \Delta x$), this equation can be expressed as

$$-\lambda T_{i,j-1}^{l+1/2} + 2(1 + \lambda)T_{i,j}^{l+1/2} - \lambda T_{i,j+1}^{l+1/2} = \lambda T_{i-1,j}^l + 2(1 - \lambda)T_{i,j}^l + \lambda T_{i+1,j}^l \quad (30.20)$$

which, when written for the system, results in a tridiagonal set of simultaneous equations.

For the second step from $t^{l+1/2}$ to t^{l+1} , Eq. (30.18) is approximated by

$$\frac{T_{i,j}^{l+1} - T_{i,j}^{l+1/2}}{\Delta t/2} = k \left[\frac{T_{i+1,j}^{l+1} - 2T_{i,j}^{l+1} + T_{i-1,j}^{l+1}}{(\Delta x)^2} + \frac{T_{i,j+1}^{l+1/2} - 2T_{i,j}^{l+1/2} + T_{i,j-1}^{l+1/2}}{(\Delta y)^2} \right] \quad (30.21)$$

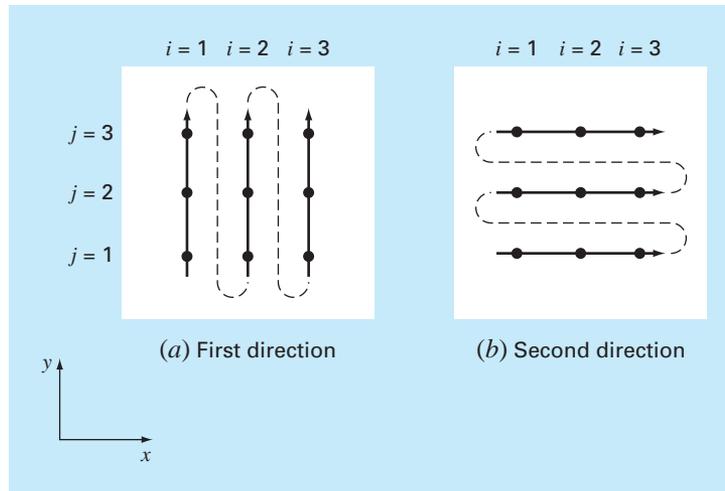
In contrast to Eq. (30.19), the approximation of $\partial^2 T / \partial x^2$ is now implicit. Thus, the bias introduced by Eq. (30.19) will be partially corrected. For a square grid, Eq. (30.21) can be written as

$$-\lambda T_{i-1,j}^{l+1} + 2(1 + \lambda)T_{i,j}^{l+1} - \lambda T_{i+1,j}^{l+1} = \lambda T_{i,j-1}^{l+1/2} + 2(1 - \lambda)T_{i,j}^{l+1/2} + \lambda T_{i,j+1}^{l+1/2} \quad (30.22)$$

Again, when written for a two-dimensional grid, the equation results in a tridiagonal system (Fig. 30.11). As in the following example, this leads to an efficient numerical solution.

FIGURE 30.11

The ADI method only results in tridiagonal equations if it is applied along the dimension that is implicit. Thus, on the first step (a), it is applied along the y dimension and, on the second step (b), along the x dimension. These “alternating directions” are the root of the method’s name.

**EXAMPLE 30.5** ADI Method

Problem Statement. Use the ADI method to solve for the temperature of the plate in Examples 29.1 and 29.2. At $t = 0$, assume that the temperature of the plate is zero and the boundary temperatures are instantaneously brought to the levels shown in Fig. 29.4. Employ a time step of 10 s. Recall from Example 30.1 that the coefficient of thermal diffusivity for aluminum is $k = 0.835 \text{ cm}^2/\text{s}$.

Solution. A value of $\Delta x = 10 \text{ cm}$ was employed to characterize the $40 \times 40\text{-cm}$ plate from Examples 29.1 and 29.2. Therefore, $\lambda = 0.835(10)/(10)^2 = 0.0835$. For the first step to $t = 5$ (Fig. 30.11a), Eq. (30.20) is applied to nodes (1, 1), (1, 2), and (1, 3) to yield the following tridiagonal equations:

$$\begin{bmatrix} 2.167 & -0.0835 & \\ -0.0835 & 2.167 & -0.0835 \\ & -0.0835 & 2.167 \end{bmatrix} \begin{Bmatrix} T_{1,1} \\ T_{1,2} \\ T_{1,3} \end{Bmatrix} = \begin{Bmatrix} 6.2625 \\ 6.2625 \\ 14.6125 \end{Bmatrix}$$

which can be solved for

$$T_{1,1} = 3.01597 \quad T_{1,2} = 3.2708 \quad T_{1,3} = 6.8692$$

In a similar fashion, tridiagonal equations can be developed and solved for

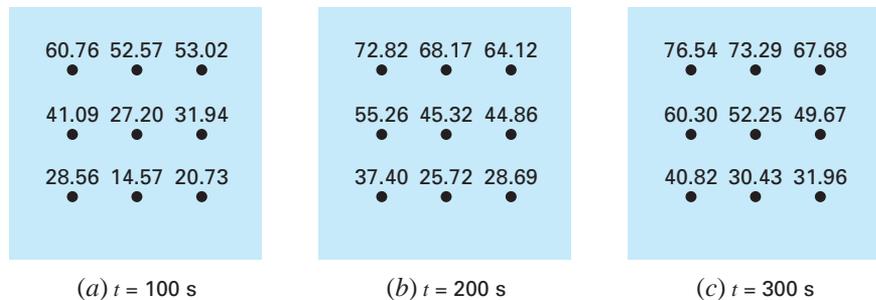
$$T_{2,1} = 0.1274 \quad T_{2,2} = 0.2900 \quad T_{2,3} = 4.1291$$

and

$$T_{3,1} = 2.0181 \quad T_{3,2} = 2.2477 \quad T_{3,3} = 6.0256$$

For the second step to $t = 10$ (Fig. 30.11b), Eq. (30.22) is applied to nodes (1, 1), (2, 1), and (3, 1) to yield

$$\begin{bmatrix} 2.167 & -0.0835 & \\ -0.0835 & 2.167 & -0.0835 \\ & -0.0835 & 2.167 \end{bmatrix} \begin{Bmatrix} T_{1,1} \\ T_{2,1} \\ T_{3,1} \end{Bmatrix} = \begin{Bmatrix} 12.0639 \\ 0.2577 \\ 8.0619 \end{Bmatrix}$$

**FIGURE 30.12**

Solution for the heated plate from Example 30.5 at (a) $t = 100$ s, (b) $t = 200$ s, and (c) $t = 300$ s.

which can be solved for

$$T_{1,1} = 5.5855 \quad T_{2,1} = 0.4782 \quad T_{3,1} = 3.7388$$

Tridiagonal equations for the other rows can be developed and solved for

$$T_{1,2} = 6.1683 \quad T_{2,2} = 0.8238 \quad T_{3,2} = 4.2359$$

and

$$T_{1,3} = 13.1120 \quad T_{2,3} = 8.3207 \quad T_{3,3} = 11.3606$$

The computation can be repeated, and the results for $t = 100$, 200, and 300 s are depicted in Fig. 30.12a through c, respectively. As expected, the temperature of the plate rises. After a sufficient time elapses, the temperature will approach the steady-state distribution of Fig. 29.5.

The ADI method is but one of a group of techniques called splitting methods. Some of these represent efforts to circumvent shortcomings of ADI. Discussion of other splitting methods as well as more information on ADI can be found elsewhere (Ferziger, 1981; Lapidus and Pinder, 1981).

PROBLEMS

30.1 Repeat Example 30.1, but use the midpoint method to generate your solution.

30.2 Repeat Example 30.1, but for the case where the rod is initially at 50°C and the derivative at $x = 0$ is equal to 1 and at $x = 10$ is equal to 0. Interpret your results.

30.3 (a) Repeat Example 30.1, but for a time step of $\Delta t = 0.05$ s. Compute results to $t = 0.2$. **(b)** In addition, perform the same computation with the Heun method (without iteration of the corrector)

with a much smaller step size of $\Delta t = 0.001$ s. Assuming that the results of **(b)** are a valid approximation of the true solution, determine percent relative errors for the results obtained in Example 30.1 as well as for part **(a)**.

30.4 Repeat Example 30.2, but for the case where the derivative at $x = 10$ is equal to zero.

30.5 Repeat Example 30.3, but for $\Delta x = 1$ cm.

30.6 Repeat Example 30.5, but for the plate described in Prob. 29.2.

30.7 The advection-diffusion equation is used to compute the distribution of concentration along the length of a rectangular chemical reactor (see Sec. 32.1),

$$\frac{\partial c}{\partial t} = D \frac{\partial^2 c}{\partial x^2} - U \frac{\partial c}{\partial x} - kc$$

where c = concentration (mg/m³), t = time (min), D = a diffusion coefficient (m²/min), x = distance along the tank's longitudinal axis (m) where $x = 0$ at the tank's inlet, U = velocity in the x direction (m/min), and k = a reaction rate (min⁻¹) whereby the chemical decays to another form. Develop an explicit scheme to solve this equation numerically. Test it for $k = 0.15$, $D = 100$, and $U = 1$ for a tank of length 10 m. Use a $\Delta x = 1$ m, and a step size $\Delta t = 0.005$. Assume that the inflow concentration is 100 and that the initial concentration in the tank is zero. Perform the simulation from $t = 0$ to 100 and plot the final resulting concentrations versus x .

30.8 Develop a user-friendly computer program for the simple explicit method from Sec. 30.2. Test it by duplicating Example 30.1.

30.9 Modify the program in Prob. 30.8 so that it employs either Dirichlet or derivative boundary conditions. Test it by solving Prob. 30.2.

30.10 Develop a user-friendly computer program to implement the simple implicit scheme from Sec. 30.3. Test it by duplicating Example 30.2.

30.11 Develop a user-friendly computer program to implement the Crank-Nicolson method from Sec. 30.4. Test it by duplicating Example 30.3.

30.12 Develop a user-friendly computer program for the ADI method described in Sec. 30.5. Test it by duplicating Example 30.5.

30.13 The nondimensional form for the transient heat conduction in an insulated rod (Eq. 30.1) can be written as

$$\frac{\partial^2 u}{\partial \bar{x}^2} = \frac{\partial u}{\partial \bar{t}}$$

where nondimensional space, time, and temperature are defined as

$$\bar{x} = \frac{x}{L} \quad \bar{t} = \frac{T}{(\rho C L^2 / k)} \quad u = \frac{T - T_o}{T_L - T_o}$$

where L = the rod length, k = thermal conductivity of the rod material, ρ = density, C = specific heat, T_o = temperature at $x = 0$, and T_L = temperature at $x = L$. This makes for the following boundary and initial conditions:

Boundary conditions	$u(0, \bar{t}) = 0$	$u(1, \bar{t}) = 0$
Initial conditions	$u(\bar{x}, 0) = 0$	$0 \leq \bar{x} \leq 1$

Solve this nondimensional equation for the temperature distribution using finite-difference methods and a second-order accurate Crank-Nicolson formulation to integrate in time. Write a computer program to obtain the solution. Increase the value of $\Delta \bar{t}$ by 10% for

each time step to more quickly obtain the steady-state solution, and select values of $\Delta \bar{x}$ and $\Delta \bar{t}$ for good accuracy. Plot the nondimensional temperature versus nondimensional length for various values of nondimensional times.

30.14 The problem of transient radial heat flow in a circular rod in nondimensional form is described by

$$\frac{\partial^2 u}{\partial \bar{r}^2} + \frac{1}{\bar{r}} \frac{\partial u}{\partial \bar{r}} = \frac{\partial u}{\partial \bar{t}}$$

Boundary conditions	$u(1, \bar{t}) = 1$	$\frac{\partial u}{\partial \bar{r}}(0, \bar{t}) = 0$
Initial conditions	$u(\bar{x}, 0) = 0$	$0 \leq \bar{x} \leq 1$

Solve the nondimensional transient radial heat-conduction equation in a circular rod for the temperature distribution at various times as the rod temperature approaches steady state. Use second-order accurate finite-difference analogues for the derivatives with a Crank-Nicolson formulation. Write a computer program for the solution. Select values of $\Delta \bar{r}$ and $\Delta \bar{t}$ for good accuracy. Plot the temperature u versus radius \bar{r} for various times \bar{t} .

30.15 Solve the following PDE:

$$\frac{\partial^2 u}{\partial x^2} + b \frac{\partial u}{\partial x} = \frac{\partial u}{\partial t}$$

Boundary conditions	$u(0, t) = 0$	$u(1, t) = 0$
Initial conditions	$u(x, 0) = 0$	$0 \leq x \leq 1$

Use second-order accurate finite-difference analogues for the derivatives with a Crank-Nicolson formulation to integrate in time. Write a computer program for the solution. Increase the value of Δt by 10% for each time step to more quickly obtain the steady-state solution, and select values of Δx and Δt for good accuracy. Plot u versus x for various values of t . Solve for values of $b = 4, 2, 0, -2, -4$.

30.16 Determine the temperatures along a 1-m horizontal rod described by the heat-conduction equation (Eq. 30.1). Assume that the right boundary is insulated and that the left boundary ($x = 0$) is represented by

$$-k' \frac{\partial T}{\partial x} \Big|_{x=0} = h(T_a - T_0)$$

where k' = coefficient of thermal conductivity (W/m · °C), h = convective heat transfer coefficient (W/m² · °C), T_a = ambient temperature (°C), and T_0 = temperature of the rod at $x = 0$ (°C). Solve for temperature as a function of time using a spatial step of $\Delta x = 1$ cm and the following parameter values: $k = 2 \times 10^{-5}$ m²/s, $k' = 10$ W/m · °C, $h = 25$ W/m² · °C, and $T_a = 50$ °C. Assume that the initial temperature of the rod is zero.

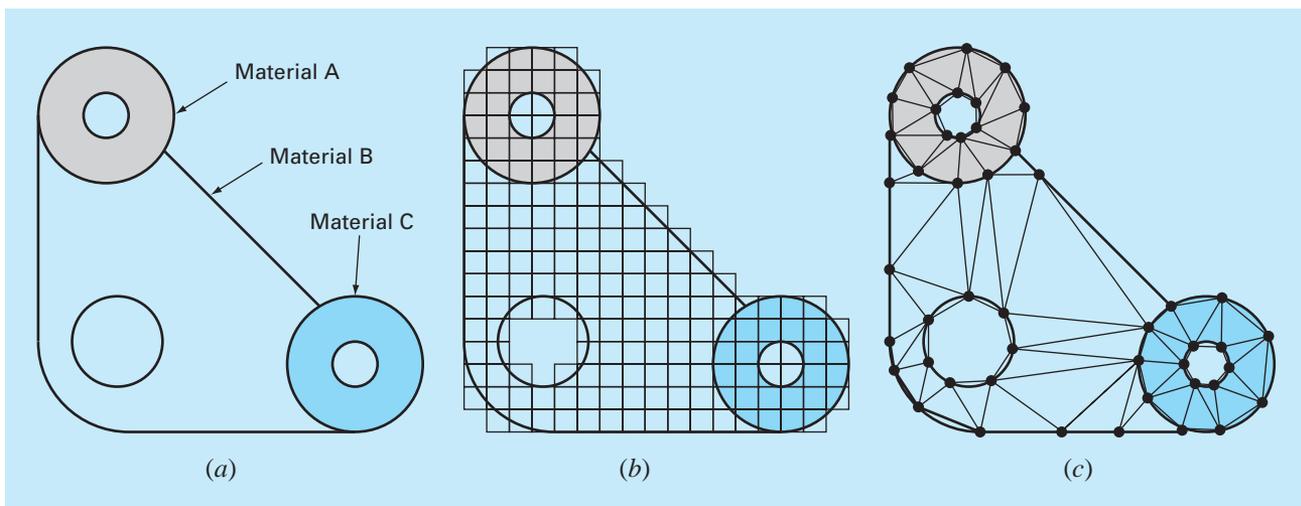
Finite-Element Method

To this juncture, we have employed *finite-difference* methods to solve partial differential equations. In these methods, the solution domain is divided into a grid of discrete points or nodes (Fig. 31.1*b*). The PDE is then written for each node and its derivatives replaced by finite-divided differences. Although such “pointwise” approximation is conceptually easy to understand, it has a number of shortcomings. In particular, it becomes harder to apply for systems with irregular geometry, unusual boundary conditions, or heterogenous composition.

The *finite-element* method provides an alternative that is better suited for such systems. In contrast to finite-difference techniques, the finite-element method divides the solution domain into simply shaped regions, or “elements” (Fig. 31.1*c*). An approximate solution for

FIGURE 31.1

(*a*) A gasket with irregular geometry and nonhomogeneous composition. (*b*) Such a system is very difficult to model with a finite-difference approach. This is due to the fact that complicated approximations are required at the boundaries of the system and at the boundaries between regions of differing composition. (*c*) A finite-element discretization is much better suited for such systems.



the PDE can be developed for each of these elements. The total solution is then generated by linking together, or “assembling,” the individual solutions taking care to ensure continuity at the interelement boundaries. Thus, the PDE is satisfied in a piecewise fashion.

As in Fig. 31.1c, the use of elements, rather than a rectangular grid, provides a much better approximation for irregularly shaped systems. Further, values of the unknown can be generated continuously across the entire solution domain rather than at isolated points.

Because a comprehensive description is beyond the scope of this book, this chapter provides a general introduction to the finite-element method. Our primary objective is to make you comfortable with the approach and cognizant of its capabilities. In this spirit, the following section is devoted to a general overview of the steps involved in a typical finite-element solution of a problem. This is followed by a simple example: a steady-state, one-dimensional heated rod. Although this example does not involve PDEs, it allows us to develop and demonstrate major aspects of the finite-element approach unencumbered by complicating factors. We can then discuss some issues involved in employing the finite-element method for PDEs.

31.1 THE GENERAL APPROACH

Although the particulars will vary, the implementation of the finite-element approach usually follows a standard step-by-step procedure. The following provides a brief overview of each of these steps. The application of these steps to engineering problem contexts will be developed in subsequent sections.

31.1.1 Discretization

This step involves dividing the solution domain into finite elements. Figure 31.2 provides examples of elements employed in one, two, and three dimensions. The points of intersection of the lines that make up the sides of the elements are referred to as nodes and the sides themselves are called *nodal lines* or *planes*.

31.1.2 Element Equations

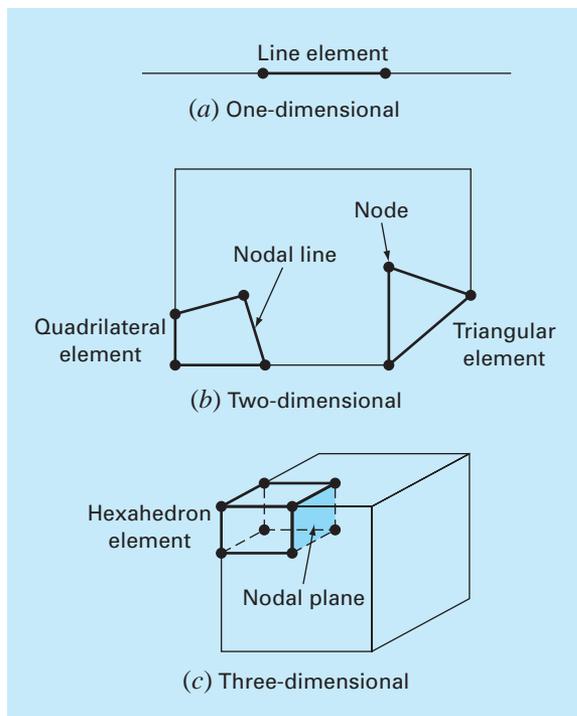
The next step is to develop equations to approximate the solution for each element. This involves two steps. First, we must choose an appropriate function with unknown coefficients that will be used to approximate the solution. Second, we evaluate the coefficients so that the function approximates the solution in an optimal fashion.

Choice of Approximation Functions. Because they are easy to manipulate mathematically, polynomials are often employed for this purpose. For the one-dimensional case, the simplest alternative is a first-order polynomial or straight line,

$$u(x) = a_0 + a_1x \quad (31.1)$$

where $u(x)$ = the dependent variable, a_0 and a_1 = constants, and x = the independent variable. This function must pass through the values of $u(x)$ at the end points of the element at x_1 and x_2 . Therefore,

$$\begin{aligned} u_1 &= a_0 + a_1x_1 \\ u_2 &= a_0 + a_1x_2 \end{aligned}$$

**FIGURE 31.2**

Examples of elements employed in (a) one, (b) two, and (c) three dimensions.

where $u_1 = u(x_1)$ and $u_2 = u(x_2)$. These equations can be solved using Cramer's rule for

$$a_0 = \frac{u_1 x_2 - u_2 x_1}{x_2 - x_1} \quad a_1 = \frac{u_2 - u_1}{x_2 - x_1}$$

These results can then be substituted into Eq. (31.1) which, after collection of terms, can be written as

$$u = N_1 u_1 + N_2 u_2 \quad (31.2)$$

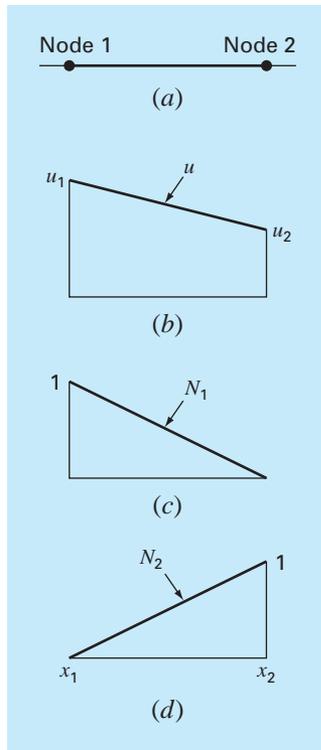
where

$$N_1 = \frac{x_2 - x}{x_2 - x_1} \quad (31.3)$$

and

$$N_2 = \frac{x - x_1}{x_2 - x_1} \quad (31.4)$$

Equation (31.2) is called an *approximation*, or *shape function*, and N_1 and N_2 are called *interpolation functions*. Close inspection reveals that Eq. (31.2) is, in fact, the Lagrange

**FIGURE 31.3**

(b) A linear approximation or shape function for (a) a line element. The corresponding interpolation functions are shown in (c) and (d).

first-order interpolating polynomial. It provides a means to predict intermediate values (that is, to interpolate) between given values u_1 and u_2 at the nodes.

Figure 31.3 shows the shape function along with the corresponding interpolation functions. Notice that the sum of the interpolation functions is equal to one.

In addition, the fact that we are dealing with linear equations facilitates operations such as differentiation and integration. Such manipulations will be important in later sections. The derivative of Eq. (31.2) is

$$\frac{du}{dx} = \frac{dN_1}{dx}u_1 + \frac{dN_2}{dx}u_2 \quad (31.5)$$

According to Eqs. (31.3) and (31.4), the derivatives of the N 's can be calculated as

$$\frac{dN_1}{dx} = -\frac{1}{x_2 - x_1} \quad \frac{dN_2}{dx} = \frac{1}{x_2 - x_1} \quad (31.6)$$

and, therefore, the derivative of u is

$$\frac{du}{dx} = \frac{1}{x_2 - x_1}(-u_1 + u_2) \quad (31.7)$$

In other words, it is a divided difference representing the slope of the straight line connecting the nodes.

The integral can be expressed as

$$\int_{x_1}^{x_2} u \, dx = \int_{x_1}^{x_2} N_1 u_1 + N_2 u_2 \, dx$$

Each term on the right-hand side is merely the integral of a right triangle with base $x_2 - x_1$ and height u . That is,

$$\int_{x_1}^{x_2} N u \, dx = \frac{1}{2}(x_2 - x_1)u$$

Thus, the entire integral is

$$\int_{x_1}^{x_2} u \, dx = \frac{u_1 + u_2}{2}(x_2 - x_1) \quad (31.8)$$

In other words, it is simply the trapezoidal rule.

Obtaining an Optimal Fit of the Function to the Solution. Once the interpolation function is chosen, the equation governing the behavior of the element must be developed. This equation represents a fit of the function to the solution of the underlying differential equation. Several methods are available for this purpose. Among the most common are the direct approach, the method of weighted residuals, and the variational approach. The outcome of all of these methods is analogous to curve fitting. However, instead of fitting functions to data, these methods specify relationships between the unknowns in Eq. (31.2) that satisfy the underlying PDE in an optimal fashion.

Mathematically, the resulting element equations will often consist of a set of linear algebraic equations that can be expressed in matrix form,

$$[k]\{u\} = \{F\} \quad (31.9)$$

where $[k]$ = an *element property* or *stiffness matrix*, $\{u\}$ = a column vector of unknowns at the nodes, and $\{F\}$ = a column vector reflecting the effect of any external influences applied at the nodes. Note that, in some cases, the equations can be nonlinear. However, for the elementary examples described herein, and for many practical problems, the systems are linear.

31.1.3 Assembly

After the individual element equations are derived, they must be linked together or assembled to characterize the unified behavior of the entire system. The assembly process is governed by the concept of continuity. That is, the solutions for contiguous elements are matched so that the unknown values (and sometimes the derivatives) at their common nodes are equivalent. Thus, the total solution will be continuous.

When all the individual versions of Eq. (31.9) are finally assembled, the entire system is expressed in matrix form as

$$[K]\{u'\} = \{F'\} \quad (31.10)$$

where $[K]$ = the *assemblage property matrix* and $\{u'\}$ and $\{F'\}$ = column vectors for unknowns and external forces that are marked with primes to denote that they are an assemblage of the vectors $\{u\}$ and $\{F\}$ from the individual elements.

31.1.4 Boundary Conditions

Before Eq. (31.10) can be solved, it must be modified to account for the system's boundary conditions. These adjustments result in

$$[\bar{k}]\{u'\} = \{\bar{F}'\} \quad (31.11)$$

where the overbars signify that the boundary conditions have been incorporated.

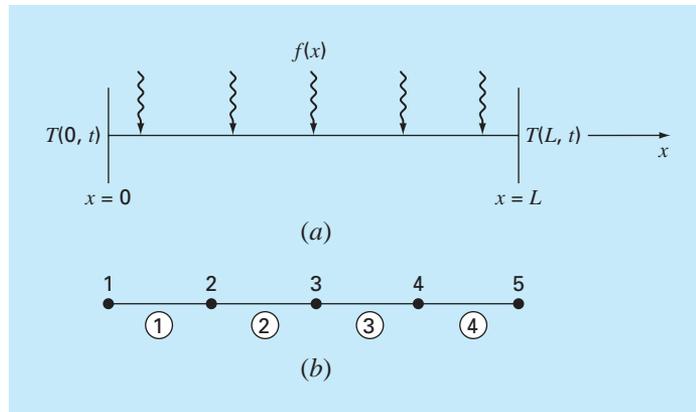
31.1.5 Solution

Solutions of Eq. (31.11) can be obtained with techniques described previously in Part Three, such as *LU* decomposition. In many cases, the elements can be configured so that the resulting equations are banded. Thus, the highly efficient solution schemes available for such systems can be employed.

31.1.6 Postprocessing

Upon obtaining a solution, it can be displayed in tabular form or graphically. In addition, secondary variables can be determined and displayed.

Although the preceding steps are very general, they are common to most implementations of the finite-element approach. In the following section, we illustrate how they can be applied to obtain numerical results for a simple physical system—a heated rod.

**FIGURE 31.4**

(a) A long, thin rod subject to fixed boundary conditions and a continuous heat source along its axis. (b) The finite-element representation consisting of four equal-length elements and five nodes.

31.2 FINITE-ELEMENT APPLICATION IN ONE DIMENSION

Figure 31.4 shows a system that can be modeled by a one-dimensional form of Poisson's equation

$$\frac{d^2T}{dx^2} = -f(x) \quad (31.12)$$

where $f(x)$ = a function defining a heat source along the rod and where the ends of the rod are held at fixed temperatures,

$$T(0, t) = T_1$$

and

$$T(L, t) = T_2$$

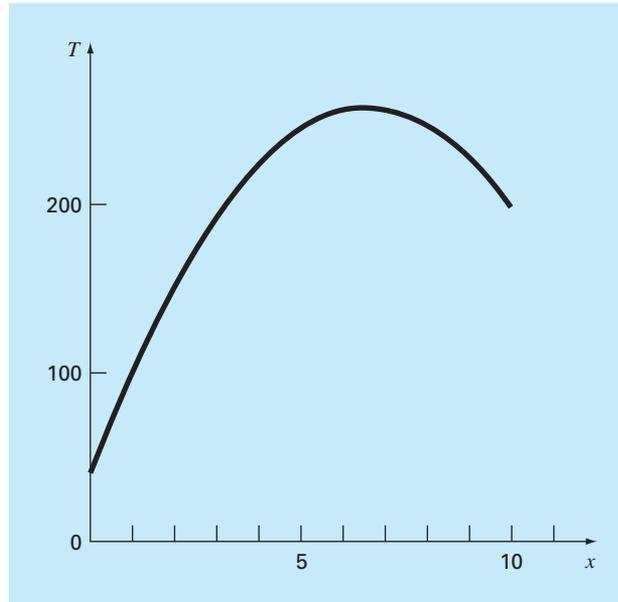
Notice that this is not a partial differential equation but rather is a boundary-value ODE. This simple model is used because it will allow us to introduce the finite-element approach without some of the complications involved in, for example, a two-dimensional PDE.

EXAMPLE 31.1 Analytical Solution for a Heated Rod

Problem Statement. Solve Eq. (31.12) for a 10-cm rod with boundary conditions of $T(0, t) = 40$ and $T(10, t) = 200$ and a uniform heat source of $f(x) = 10$.

Solution. The equation to be solved is

$$\frac{d^2T}{dx^2} = -10$$

**FIGURE 31.5**

The temperature distribution along a heated rod subject to a uniform heat source and held at fixed end temperatures.

Assume a solution of the form

$$T = ax^2 + bx + c$$

which can be differentiated twice to give $T'' = 2a$. Substituting this result into the differential equation gives $a = -5$. The boundary conditions can be used to evaluate the remaining coefficients. For the first condition at $x = 0$,

$$40 = -5(0)^2 + b(0) + c$$

or $c = 40$. Similarly, for the second condition,

$$200 = -5(10)^2 + b(10) + 40$$

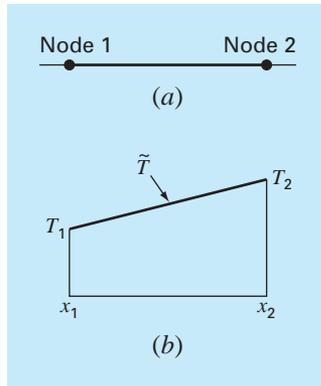
which can be solved for $b = 66$. Therefore, the final solution is

$$T = -5x^2 + 66x + 40$$

The results are plotted in Fig. 31.5.

31.2.1 Discretization

A simple configuration to model the system is a series of equal-length elements (Fig. 31.4b). Thus, the system is treated as four equal-length elements and five nodes.

**FIGURE 31.6**

(a) An individual element.
 (b) The approximation function used to characterize the temperature distribution along the element.

31.2.2 Element Equations

An individual element is shown in Fig. 31.6a. The distribution of temperature for the element can be represented by the approximation function

$$\tilde{T} = N_1 T_1 + N_2 T_2 \quad (31.13)$$

where N_1 and N_2 = linear interpolation functions specified by Eqs. (31.3) and (31.4), respectively. Thus, as depicted in Fig. 31.6b, the approximation function amounts to a linear interpolation between the two nodal temperatures.

As noted in Sec. 31.1, there are a variety of approaches for developing the element equation. In this section, we employ two of these. First, a *direct approach* will be used for the simple case where $f(x) = 0$. Then, because of its general applicability in engineering, we will devote most of the section to the *method of weighted residuals*.

The Direct Approach. For the case where $f(x) = 0$, a direct method can be employed to generate the element equations. The relationship between heat flux and temperature gradient can be represented by Fourier's law:

$$q = -k' \frac{dT}{dx}$$

where q = flux [cal/(cm² · s)] and k' = the coefficient of thermal conductivity [cal/(s · cm · °C)]. If a linear approximation function is used to characterize the element's temperature, the heat flow into the element through node 1 can be represented by

$$q_1 = k' \frac{T_1 - T_2}{x_2 - x_1}$$

where q_1 is heat flux at node 1. Similarly, for node 2,

$$q_2 = k' \frac{T_2 - T_1}{x_2 - x_1}$$

These two equations express the relationship of the element's internal temperature distribution (as reflected by the nodal temperatures) to the heat flux at its ends. As such, they constitute our desired element equations. They can be simplified further by recognizing that Fourier's law can be used to couch the end fluxes themselves in terms of the temperature gradients at the boundaries. That is,

$$q_1 = -k' \frac{dT(x_1)}{dx} \quad q_2 = k' \frac{dT(x_2)}{dx}$$

which can be substituted into the element equations to give

$$\frac{1}{x_2 - x_1} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} T_1 \\ T_2 \end{Bmatrix} = \begin{Bmatrix} -\frac{dT(x_1)}{dx} \\ \frac{dT(x_2)}{dx} \end{Bmatrix} \quad (31.14)$$

Notice that Eq. (31.14) has been cast in the format of Eq. (31.9). Thus, we have succeeded in generating a matrix equation that describes the behavior of a typical element in our system.

The direct approach has great intuitive appeal. Additionally, in areas such as mechanics, it can be employed to solve meaningful problems. However, in other contexts, it is often difficult or impossible to derive finite-element equations directly. Consequently, as described next, more general mathematical techniques are available.

The Method of Weighted Residuals. The differential equation (31.12) can be reexpressed as

$$0 = \frac{d^2 T}{dx^2} + f(x)$$

The approximate solution [Eq. (31.13)] can be substituted into this equation. Because Eq. (31.13) is not the exact solution, the left side of the resulting equation will not be zero but will equal a residual,

$$R = \frac{d^2 \tilde{T}}{dx^2} + f(x) \quad (31.15)$$

The *method of weighted residuals (MWR)* consists of finding a minimum for the residual according to the general formula

$$\int_D R W_i dD = 0 \quad i = 1, 2, \dots, m \quad (31.16)$$

where D = the solution domain and the W_i = linearly independent weighting functions.

At this point, there are a variety of choices that could be made for the weighting function (Box 31.1). The most common approach for the finite-element method is to employ the interpolation functions N_i as the weighting functions. When these are substituted into Eq. (31.16), the result is referred to as Galerkin's method,

$$\int_D R N_i dD = 0 \quad i = 1, 2, \dots, m$$

For our one-dimensional rod, Eq. (31.15) can be substituted into this formulation to give

$$\int_{x_1}^{x_2} \left[\frac{d^2 \tilde{T}}{dx^2} + f(x) \right] N_i dx \quad i = 1, 2$$

which can be reexpressed as

$$\int_{x_1}^{x_2} \frac{d^2 \tilde{T}}{dx^2} N_i(x) dx = - \int_{x_1}^{x_2} f(x) N_i(x) dx \quad i = 1, 2 \quad (31.17)$$

At this point, a number of mathematical manipulations will be applied to simplify and evaluate Eq. (31.17). Among the most important is the simplification of the left-hand side using integration by parts. Recall from calculus that this operation can be expressed generally as

$$\int_a^b u dv = uv|_a^b - \int_a^b v du$$

Box 31.1 Alternative Residual Schemes for the MWR

Several choices can be made for the weighting functions of Eq. (31.16). Each represents an alternative approach for the MWR.

In the *collocation approach*, we choose as many locations as there are unknown coefficients. Then, the coefficients are adjusted until the residual vanishes at each of these locations. Consequently, the approximating function will yield perfect results at the chosen locations but will have a nonzero residual elsewhere. Thus, it is akin to the interpolation methods in Chap. 18. Note that collocation amounts to using the weighting function

$$W = \delta(x - x_i) \quad \text{for } i = 1, 2, \dots, n$$

where $n =$ the number of unknown coefficients and $\delta(x - x_i) =$ the *Dirac delta function* that vanishes everywhere but at $x = x_i$, where it equals 1.

In the *subdomain method*, the interval is divided into as many segments, or “subdomains,” as there are unknown coefficients. Then, the coefficients are adjusted until the average value of the residual is zero in each subdomain. Thus, for each subdomain, the weighting function is equal to 1 and Eq. (31.16) is

$$\int_{x_{i-1}}^{x_i} R \, dx = 0 \quad \text{for } i = 1, 2, \dots, n$$

where x_{i-1} and x_i are the bounds of the subdomain.

For the *least-squares* case, the coefficients are adjusted so as to minimize the integral of the square of the residual. Thus, the weighting functions are

$$W_i = \frac{\partial R}{\partial a_i}$$

which can be substituted into Eq. (31.16) to give

$$\int_D R \frac{\partial R}{\partial a_i} \, dD = 0 \quad i = 1, 2, \dots, n$$

or

$$\frac{\partial}{\partial a_i} \int_D R^2 \, dD = 0 \quad i = 1, 2, \dots, n$$

Comparison of the formulation with those of Chap. 17 shows that this is the continuous form of regression.

Galerkin’s method employs the interpolation functions N_i as weighting functions. Recall that these functions always sum to 1 at any position in an element. For many problem contexts, Galerkin’s method yields the same results as are obtained by variational methods. Consequently, it is the most commonly employed version of MWR used in finite-element analysis.

If u and v are chosen properly, the new integral on the right-hand side will be easier to evaluate than the original one on the left-hand side. This can be done for the term on the left-hand side of Eq. (31.17) by choosing $N_i(x)$ as u and $(d^2T/dx^2) \, dx$ as dv to yield

$$\int_{x_1}^{x_2} N_i(x) \frac{d^2\tilde{T}}{dx^2} \, dx = N_i(x) \frac{d\tilde{T}}{dx} \Big|_{x_1}^{x_2} - \int_{x_1}^{x_2} \frac{d\tilde{T}}{dx} \frac{dN_i}{dx} \, dx \quad i = 1, 2 \quad (31.18)$$

Thus, we have taken the significant step of lowering the highest-order term in the formulation from a second to a first derivative.

Next, we can evaluate the individual terms that we have created in Eq. (31.18). For $i = 1$, the first term on the right-hand side of Eq. (31.18) can be evaluated as

$$N_1(x) \frac{d\tilde{T}}{dx} \Big|_{x_1}^{x_2} = N_1(x_2) \frac{d\tilde{T}(x_2)}{dx} - N_1(x_1) \frac{d\tilde{T}(x_1)}{dx}$$

However, recall from Fig. 31.3 that $N_1(x_2) = 0$ and $N_1(x_1) = 1$, and therefore,

$$N_1(x) \frac{d\tilde{T}}{dx} \Big|_{x_1}^{x_2} = - \frac{d\tilde{T}(x_1)}{dx} \quad (31.19)$$

Similarly, for $i = 2$,

$$N_2(x) \left. \frac{d\tilde{T}}{dx} \right|_{x_1}^{x_2} = \frac{d\tilde{T}(x_2)}{dx} \quad (31.20)$$

Thus, the first term on the right-hand side of Eq. (31.18) represents the natural boundary conditions at the ends of the elements.

Now, before proceeding let us regroup by substituting our results back into the original equation. Substituting Eqs. (31.18) through (31.20) into Eq. (31.17) and rearranging gives for $i = 1$,

$$\int_{x_1}^{x_2} \frac{d\tilde{T}}{dx} \frac{dN_1}{dx} dx = -\frac{d\tilde{T}(x_1)}{dx} + \int_{x_1}^{x_2} f(x)N_1(x) dx \quad (31.21)$$

and for $i = 2$,

$$\int_{x_1}^{x_2} \frac{d\tilde{T}}{dx} \frac{dN_2}{dx} dx = \frac{d\tilde{T}(x_2)}{dx} + \int_{x_1}^{x_2} f(x)N_2(x) dx \quad (31.22)$$

Notice that the integration by parts has led to two important outcomes. First, it has incorporated the boundary conditions directly into the element equations. Second, it has lowered the highest-order evaluation from a second to a first derivative. This latter outcome yields the significant result that the approximation functions need to preserve continuity of value but not slope at the nodes.

Also notice that we can now begin to ascribe some physical significance to the individual terms we have derived. On the right-hand side of each equation, the first term represents one of the element's boundary conditions and the second is the effect of the system's forcing function—in the present case, the heat source $f(x)$. As will now become evident, the left-hand side embodies the internal mechanisms that govern the element's temperature distribution. That is, in terms of the finite-element method, the left-hand side will become the element property matrix.

To see this, let us concentrate on the terms on the left-hand side. For $i = 1$, the term is

$$\int_{x_1}^{x_2} \frac{d\tilde{T}}{dx} \frac{dN_1}{dx} dx \quad (31.23)$$

Recall from Sec. 31.1.2 that the linear nature of the shape function makes differentiation and integration simple. Substituting Eqs. (31.6) and (31.7) into Eq. (31.23) gives

$$\int_{x_1}^{x_2} \frac{T_1 - T_2}{(x_2 - x_1)^2} dx = \frac{1}{x_2 - x_1} (T_1 - T_2) \quad (31.24)$$

Similar substitutions for $i = 2$ [Eq. (31.22)] yield

$$\int_{x_1}^{x_2} \frac{-T_1 + T_2}{(x_2 - x_1)^2} dx = \frac{1}{x_2 - x_1} (-T_1 + T_2) \quad (31.25)$$

Comparison with Eq. (31.14) shows that these are similar to the relationships that were developed with the direct method using Fourier's law. This can be made even clearer by

reexpressing Eqs. (31.24) and (31.25) in matrix form as

$$\frac{1}{x_2 - x_1} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} T_1 \\ T_2 \end{Bmatrix}$$

Substituting this result into Eqs. (31.21) and (31.22) and expressing the result in matrix form gives the final version of the element equations

$$\underbrace{\frac{1}{x_2 - x_1} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \{T\}}_{\text{Element stiffness matrix}} = \underbrace{\begin{Bmatrix} -\frac{dT(x_1)}{dx} \\ \frac{dT(x_2)}{dx} \end{Bmatrix}}_{\text{Boundary condition}} + \underbrace{\begin{Bmatrix} \int_{x_1}^{x_2} f(x)N_1(x) dx \\ \int_{x_1}^{x_2} f(x)N_2(x) dx \end{Bmatrix}}_{\text{External effects}} \quad (31.26)$$

Note that aside from the direct and the weighted residual methods, the element equations can also be derived using variational calculus (for example, see Allaire, 1985). For the present case, this approach yields equations that are identical to those derived above.

EXAMPLE 31.2

Element Equation for a Heated Rod

Problem Statement. Employ Eq. (31.26) to develop the element equations for a 10-cm rod with boundary conditions of $T(0, t) = 40$ and $T(10, t) = 200$ and a uniform heat source of $f(x) = 10$. Employ four equal-size elements of length = 2.5 cm.

Solution. The heat source term in the first row of Eq. (31.26) can be evaluated by substituting Eq. (31.3) and integrating to give

$$\int_0^{2.5} 10 \frac{2.5 - x}{2.5} dx = 12.5$$

Similarly, Eq. (31.4) can be substituted into the heat source term of the second row of Eq. (31.26), which can also be integrated to yield

$$\int_0^{2.5} 10 \frac{x - 0}{2.5} dx = 12.5$$

These results along with the other parameter values can be substituted into Eq. (31.26) to give

$$0.4T_1 - 0.4T_2 = -\frac{dT}{dx}(x_1) + 12.5$$

and

$$-0.4T_1 + 0.4T_2 = \frac{dT}{dx}(x_2) + 12.5$$

31.2.3 Assembly

Before the element equations are assembled, a global numbering scheme must be established to specify the system's topology or spatial layout. As in Table 31.1, this defines the

TABLE 31.1 The system topology for the finite-element segmentation scheme from Fig. 31.4b.

Element	Node Numbers	
	Local	Global
1	1	1
	2	2
2	1	2
	2	3
3	1	3
	2	4
4	1	4
	2	5

FIGURE 31.7

The assembly of the equations for the total system.

$$\begin{aligned}
 (a) \quad & \begin{bmatrix} 0.4 & -0.4 & 0 & 0 & 0 \\ -0.4 & 0.4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} T_1 \\ T_2 \\ 0 \\ 0 \\ 0 \end{Bmatrix} = \begin{Bmatrix} -dT(x_1)/dx + 12.5 \\ dT(x_2)/dx + 12.5 \\ 0 \\ 0 \\ 0 \end{Bmatrix} \\
 (b) \quad & \begin{bmatrix} 0.4 & -0.4 & 0 & 0 & 0 \\ -0.4 & 0.4 & +0.4 & -0.4 & 0 \\ 0 & 0 & -0.4 & 0.4 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} T_1 \\ T_2 \\ T_3 \\ 0 \\ 0 \end{Bmatrix} = \begin{Bmatrix} -dT(x_1)/dx + 12.5 \\ 12.5 + 12.5 \\ dT(x_3)/dx + 12.5 \\ 0 \\ 0 \end{Bmatrix} \\
 (c) \quad & \begin{bmatrix} 0.4 & -0.4 & 0 & 0 & 0 \\ -0.4 & 0.8 & -0.4 & 0 & 0 \\ 0 & -0.4 & 0.4 & +0.4 & -0.4 \\ 0 & 0 & -0.4 & 0.4 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ 0 \end{Bmatrix} = \begin{Bmatrix} -dT(x_1)/dx + 12.5 \\ 25 \\ 12.5 + 12.5 \\ dT(x_4)/dx + 12.5 \\ 0 \end{Bmatrix} \\
 (d) \quad & \begin{bmatrix} 0.4 & -0.4 & 0 & 0 & 0 \\ -0.4 & 0.8 & -0.4 & 0 & 0 \\ 0 & -0.4 & 0.8 & -0.4 & 0 \\ 0 & 0 & -0.4 & 0.4 & +0.4 \\ 0 & 0 & 0 & -0.4 & 0.4 \end{bmatrix} \begin{Bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{Bmatrix} = \begin{Bmatrix} -dT(x_1)/dx + 12.5 \\ 25 \\ 25 \\ 12.5 + 12.5 \\ dT(x_5)/dx + 12.5 \end{Bmatrix} \\
 (e) \quad & \begin{bmatrix} 0.4 & -0.4 & 0 & 0 & 0 \\ -0.4 & 0.8 & -0.4 & 0 & 0 \\ 0 & -0.4 & 0.8 & -0.4 & 0 \\ 0 & 0 & -0.4 & 0.8 & -0.4 \\ 0 & 0 & 0 & -0.4 & 0.4 \end{bmatrix} \begin{Bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{Bmatrix} = \begin{Bmatrix} -dT(x_1)/dx + 12.5 \\ 25 \\ 25 \\ 25 \\ dT(x_5)/dx + 12.5 \end{Bmatrix}
 \end{aligned}$$

connectivity of the element mesh. Because the present case is one-dimensional, the numbering scheme might seem so predictable that it is trivial. However, for two- and three-dimensional problems it offers the only means to specify which nodes belong to which elements.

Once the topology is specified, the element equation (31.26) can be written for each element using the global coordinates. Then they can be added one at a time to assemble the total system matrix (note that this process is explored further in Sec. 32.4). The process is depicted in Fig. 31.7.

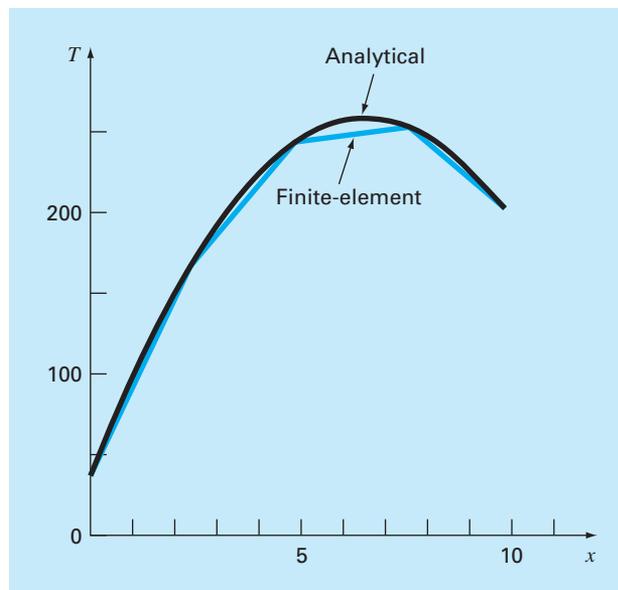
31.2.4 Boundary Conditions

Notice that, as the equations are assembled, the internal boundary conditions cancel. Thus, the final result for $\{F\}$ in Fig. 31.7e has boundary conditions for only the first and the last nodes. Because T_1 and T_5 are given, these natural boundary conditions at the ends of the bar, $dT(x_1)/dx$ and $dT(x_5)/dx$, represent unknowns. Therefore, the equations can be re-expressed as

$$\begin{array}{rcl}
 \frac{dT}{dx}(x_1) & -0.4T_2 & = -3.5 \\
 & 0.8T_2 & -0.4T_3 & = 41 \\
 -0.4T_2 & +0.8T_3 & -0.4T_4 & = 25 \\
 & -0.4T_3 & +0.8T_4 & = 105 \\
 & & -0.4T_4 & -\frac{dT}{dx}(x_5) = -67.5
 \end{array} \quad (31.27)$$

FIGURE 31.8

Results of applying the finite-element approach to a heated bar. The exact solution is also shown.



31.2.5 Solution

Equation (31.27) can be solved for

$$\begin{aligned} \frac{dT}{dx}(x_1) &= 66 & T_2 &= 173.75 & T_3 &= 245 \\ T_4 &= 253.75 & \frac{dT}{dx}(x_5) &= -34 \end{aligned}$$

31.2.6 Postprocessing

The results can be displayed graphically. Figure 31.8 shows the finite-element results along with the exact solution. Notice that the finite-element calculation captures the overall trend of the exact solution and, in fact, provides an exact match at the nodes. However, a discrepancy exists in the interior of each element due to the linear nature of the shape functions.

31.3 TWO-DIMENSIONAL PROBLEMS

Although the mathematical “bookkeeping” increases markedly, the extension of the finite-element approach to two dimensions is conceptually similar to the one-dimensional applications discussed to this point. It thus follows the same steps as were outlined in Sec. 31.1.

31.3.1 Discretization

A variety of simple elements such as triangles or quadrilaterals are usually employed for the finite-element mesh in two dimensions. In the present discussion, we will limit ourselves to triangular elements of the type depicted in Fig. 31.9.

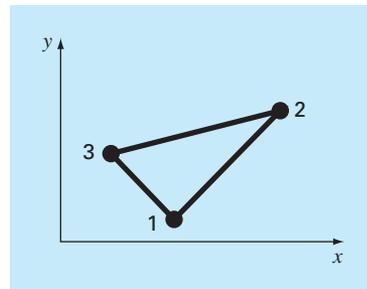
31.3.2 Element Equations

Just as for the one-dimensional case, the next step is to develop an equation to approximate the solution for the element. For a triangular element, the simplest approach is the linear polynomial [compare with Eq. (31.1)]

$$u(x, y) = a_0 + a_{1,1}x + a_{1,2}y \quad (31.28)$$

FIGURE 31.9

A triangular element.



where $u(x, y)$ = the dependent variable, the a 's = coefficients, and x and y = independent variables. This function must pass through the values of $u(x, y)$ at the triangle's nodes (x_1, y_1) , (x_2, y_2) , and (x_3, y_3) . Therefore,

$$u_1(x, y) = a_0 + a_{1,1}x_1 + a_{1,2}y_1$$

$$u_2(x, y) = a_0 + a_{1,1}x_2 + a_{1,2}y_2$$

$$u_3(x, y) = a_0 + a_{1,1}x_3 + a_{1,2}y_3$$

or in matrix form,

$$\begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix} \begin{Bmatrix} a_0 \\ a_{1,1} \\ a_{1,2} \end{Bmatrix} = \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \end{Bmatrix}$$

which can be solved for

$$a_0 = \frac{1}{2A_e} [u_1(x_2y_3 - x_3y_2) + u_2(x_3y_1 - x_1y_3) + u_3(x_1y_2 - x_2y_1)] \quad (31.29)$$

$$a_{1,1} = \frac{1}{2A_e} [u_1(y_2 - y_3) + u_2(y_3 - y_1) + u_3(y_1 - y_2)] \quad (31.30)$$

$$a_{1,2} = \frac{1}{2A_e} [u_1(x_3 - x_2) + u_2(x_1 - x_3) + u_3(x_2 - x_1)] \quad (31.31)$$

where A_e is the area of the triangular element,

$$A_e = \frac{1}{2} [(x_2y_3 - x_3y_2) + (x_3y_1 - x_1y_3) + (x_1y_2 - x_2y_1)]$$

Equations (31.29) through (31.31) can be substituted into Eq. (31.28). After a collection of terms, the result can be expressed as

$$u = N_1u_1 + N_2u_2 + N_3u_3 \quad (31.32)$$

where

$$N_1 = \frac{1}{2A_e} [(x_2y_3 - x_3y_2) + (y_2 - y_3)x + (x_3 - x_2)y]$$

$$N_2 = \frac{1}{2A_e} [(x_3y_1 - x_1y_3) + (y_3 - y_1)x + (x_1 - x_3)y]$$

$$N_3 = \frac{1}{2A_e} [(x_1y_2 - x_2y_1) + (y_1 - y_2)x + (x_2 - x_1)y]$$

Equation (31.32) provides a means to predict intermediate values for the element on the basis of the values at its nodes. Figure 31.10 shows the shape function along with the corresponding interpolation functions. Notice that the sum of the interpolation functions is always equal to 1.

As with the one-dimensional case, various methods are available for developing element equations based on the underlying PDE and the approximating functions. The resulting equations are considerably more complicated than Eq. (31.26). However, because the

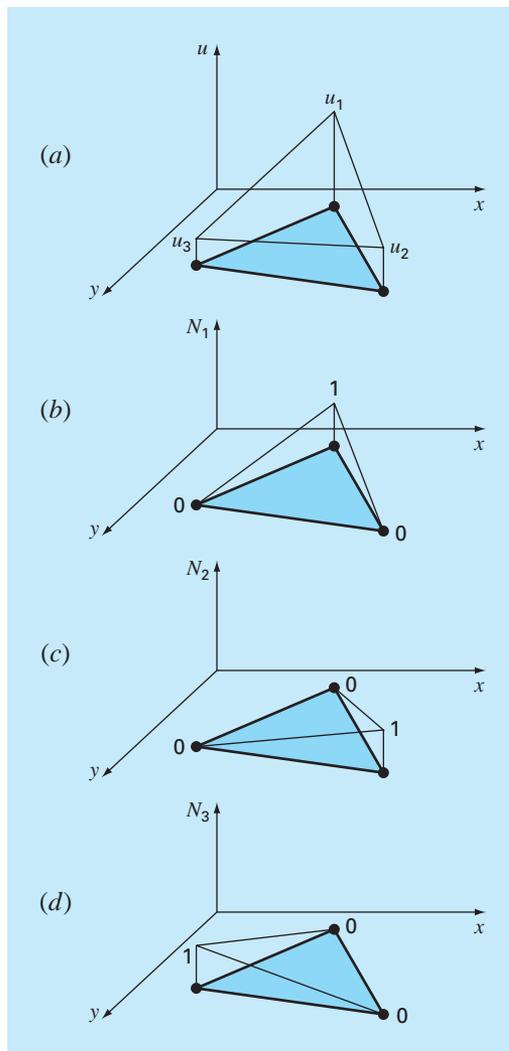


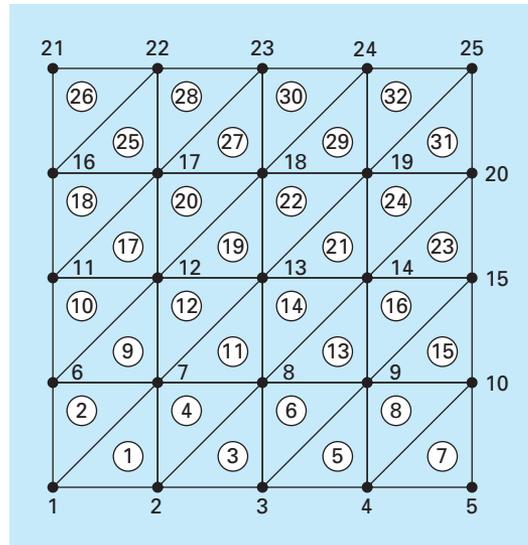
FIGURE 31.10

(a) A linear approximation function for a triangular element. The corresponding interpolation functions are shown in (b) through (d).

approximating functions are usually lower-order polynomials like Eq. (31.28), the terms of the final element matrix will consist of lower-order polynomials and constants.

31.3.3 Boundary Conditions and Assembly

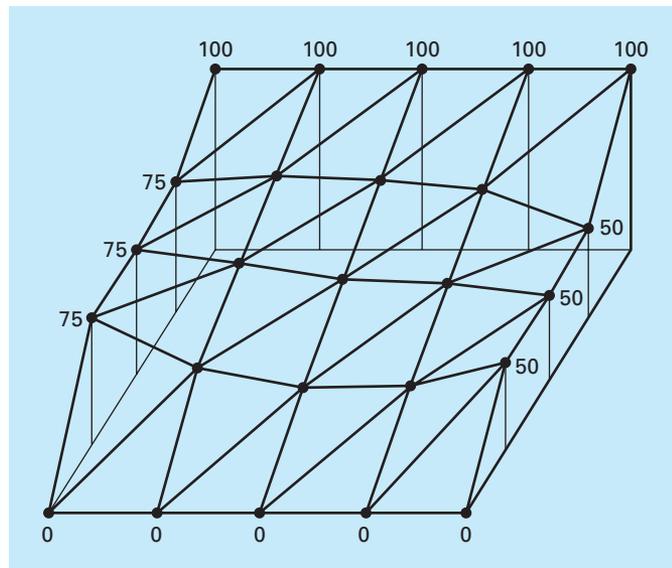
The incorporation of boundary conditions and the assembly of the system matrix also become more complicated when the finite-element technique is applied to two- and three-dimensional problems. However, as with the derivation of the element matrix, the difficulty

**FIGURE 31.11**

A numbering scheme for the nodes and elements of a finite-element approximation of the heated plate that was previously characterized by finite differences in Chap. 29.

FIGURE 31.12

The temperature distribution of a heated plate as calculated with a finite-element method.



relates to the mechanics of the process rather than to conceptual complexity. For example, the establishment of the system topology which was trivial for the one-dimensional case becomes a matter of great importance in two and three dimensions. In particular, the choice of a numbering scheme will dictate the bandedness of the resulting system matrix and hence the efficiency with which it can be solved. Figure 31.11 shows a scheme that was developed for the heated plate formerly solved by finite-difference methods in Chap. 29.

31.3.4 Solution and Postprocessing

Although the mechanics are complicated, the system matrix is merely a set of n simultaneous equations that can be used to solve for the values of the dependent variable at the n nodes. Figure 31.12 shows a solution that corresponds to the finite-difference solution from Fig. 29.5.

31.4 SOLVING PDES WITH SOFTWARE PACKAGES

Software packages have some capabilities for directly solving PDEs. However, as described in the following sections, many of the solutions are limited to simple problems. This is particularly true of two- and three-dimensional cases. For these situations, generic packages (that is, ones not expressly developed to solve PDEs such as finite-element packages) are often limited to simple rectangular domains.

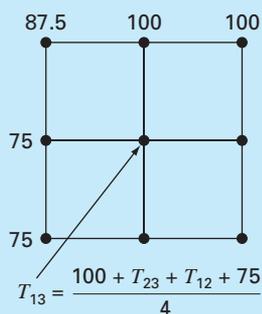
Although this might seem limiting, simple applications can be of great utility in a pedagogical sense. This is particularly true when the packages' visualization tools are used to display calculation results.

31.4.1 Excel

Although Excel does not have the direct capability to solve PDEs, it is a nice environment to develop simple solutions of elliptic PDEs. For example, the orthogonal layout of the spreadsheet cells (Fig. 31.13b) is directly analogous to the grid used in Chap. 29 to model the heated plate (Fig. 31.13a).

FIGURE 31.13

The analogy between (a) a rectangular grid and (b) the cells of a spreadsheet.



(a) Grid

	A	B	C	D	E
1	87.5	100	100	100	75
2	75	78.57	76.12	69.64	50
3	75	63.17	56.25	52.46	50
4	75	42.86	33.26	33.93	50
5	37.5	0	0	0	25

$$B2 = \frac{B1 + C2 + B3 + A2}{4}$$

(b) Spreadsheet

As in Fig. 31.13*b*, the Dirichlet boundary conditions can first be entered along the periphery of the cell block. The formula for the Liebmann method can be implemented by entering Eq. (29.12) in one of the cells in the interior (like cell B2 in Fig. 31.13*b*). Thus, the value for the cell can be computed as a function of its adjacent cells. Then the cell can be copied to the other interior cells. Because of the relative nature of the Excel copy command, all the other cells will properly be dependent on their adjacent cells.

Once you have copied the formula, you will probably get an error message: **Cannot resolve circular references**. You can rectify this by going to the *T*(ools) menu and selecting *O*(ptions). Then select the **Calculation tab** and check the **Iteration** box. This will allow the spreadsheet to recalculate (the default is 100 iterations) and solve Liebmann's method iteratively. After this occurs, strike the F9 key to manually recalculate the sheet until the answers do not vary. This means that the solution has converged.

Once the problem has been solved, Excel's graphics tools can be used to visualize the results. An example is shown in Fig. 31.14*a*. For this case, we have

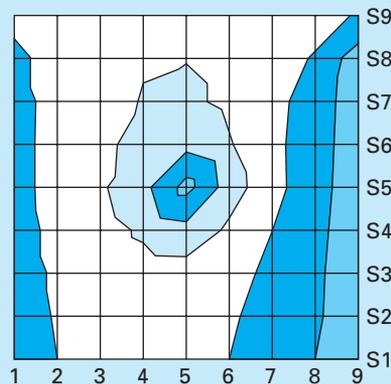
- Used a finer grid
- Made the lower boundary insulated
- Added a heat source of 150 to the middle of the plate (cell E5).

FIGURE 31.14

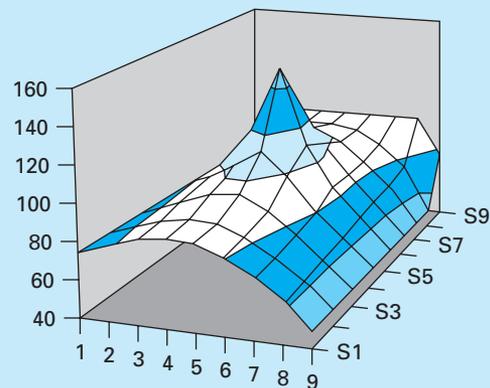
(a) Excel solution of the Poisson equation for a plate with an insulated lower edge and a heat source. A (b) "topographic map" and (c) a 3-D display of the temperatures.

	A	B	C	D	E	F	G	H	I
1	87.5	100.0	100.0	100.0	100.0	100.0	100.0	100.0	75.0
2	75.0	89.2	95.8	99.1	99.7	96.6	89.9	77.6	50.0
3	75.0	86.2	94.7	100.9	103.1	96.7	85.5	70.3	50.0
4	75.0	85.7	96.1	106.7	115.3	101.4	85.2	68.2	50.0
5	75.0	85.5	97.4	114.3	150.0	108.6	85.6	67.3	50.0
6	75.0	84.0	93.4	103.4	111.6	97.4	81.3	65.6	50.0
7	75.0	82.2	88.9	94.2	95.6	88.1	76.6	63.6	50.0
8	75.0	80.9	85.9	88.9	88.4	82.8	73.5	62.2	50.0
9	75.0	80.4	84.9	87.3	86.3	81.1	72.4	61.7	50.0

(a)



(b)



(c)

The numerical results from Fig. 31.14a can then be displayed with Excel's Chart Wizard. Figure 31.14b and c show 3-D surface plots. The y orientation of these are normally the reverse of the spreadsheet. Thus, the top high-temperature edge (100) would normally be displayed at the bottom of the plot. We reversed the y values on our sheet prior to plotting so that the graphs would be consistent with the spreadsheet.

Notice how the graphs help you visualize what is going on. Heat flows down from the source toward the boundaries, forming a mountainlike shape. Heat also flows from the high-temperature boundary down to the two side edges. Notice how the heat flows preferentially toward the lower-temperature edge (50). Finally, notice how the temperature gradient in the y dimension goes to zero at the lower insulated edge ($\partial T/\partial y \rightarrow 0$).

31.4.2 MATLAB

Although the standard MATLAB software package does not presently have great capabilities for solving PDEs, M-files and functions can certainly be developed for this purpose. In addition, its display capabilities are very nice, particularly for visualization of 2-D spatial problems.

To illustrate this capability, we first set up the Excel spreadsheet in Fig. 31.14a. These results can be saved as a text (Tab delimited) file with a name like **plate.txt**. This file can then be moved to the MATLAB directory.

Once in MATLAB, the file can be loaded by typing

```
>> load plate.txt
```

Next, the gradients can be simply calculated as

```
>> [px,py]=gradient(plate);
```

Note that this is the simplest method to compute gradients using default values of $dx = dy = 1$. Therefore, the directions and relative magnitudes will be correct.

Finally, a series of commands can be used to develop the plot. The command `contour` develops a contour plot of the data. The command `clabel` adds contour labels to the plot. Finally, `quiver` takes the gradient data and adds it to the plot as arrows,

```
>> cs=contour(plate);clabel(cs);hold on
>> quiver(-px,-py);hold off
```

Note that the minus signs are added because of the minus sign in Fourier's law [Eq. (29.4)]. As see in Fig. 31.15, the resulting plot provides an excellent representation of the solution.

Note that any file in the proper format can be entered into MATLAB and displayed in this way. This sharing of files between tools is becoming commonplace. In addition, files can be created in one location on one tool, transmitted over the internet to another location, where the file might be displayed with another tool. This is one of the exciting aspects of modern numerical applications.

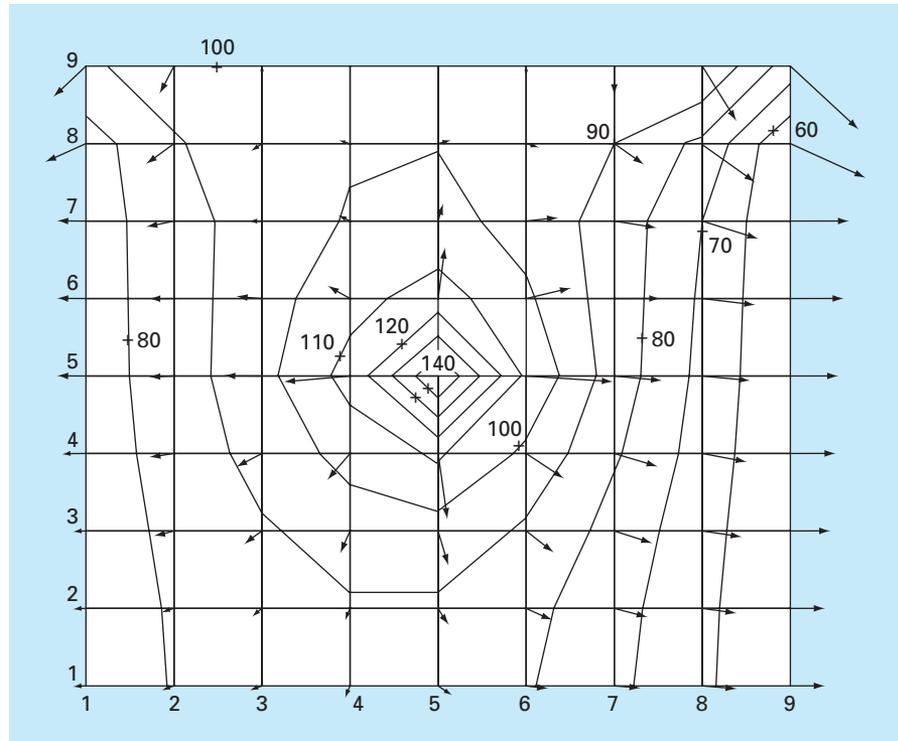


FIGURE 31.15

MATLAB-generated contour plots for the heated plate calculated with Excel (Fig. 31.14).

31.4.3 Mathcad

Mathcad has two functions that can solve Poisson's equation. You can use the **relax** function when you know the value of the unknown on all four sides of a square region. This function solves a system of linear algebraic equations using Gauss-Seidel iteration with overrelaxation to speed the rate of convergence. For the special case where there are internal sources or sinks, and the unknown function is zero on all four sides of the square, then you can use the **multigrid** function, which is usually faster than **relax**. Both of these functions return a square matrix where the location of the element in the matrix corresponds to its location within the square region. The value of the element approximates the value of the solution of Poisson's equation at this point.

Figure 31.16 shows an example where a square plate contains heat sources while the boundary is maintained at zero. The first step is to establish dimensions for the temperature grid and the heat source matrix. The temperature grid has dimensions $(R + 1) \times (R + 1)$ while the heat source matrix is $R \times R$. For example, a 3×3 temperature grid has $4 (2 \times 2)$ possible heat sources. In this case, we establish a 33×33 temperature grid and a 32×32 heat source matrix. The Mathcad command $M_{RR} := 0$ (with $R = 32$) establishes the dimensions of the source matrix and sets all the elements to zero. Next, the location and strength of two heat sources are established. Finally, S is the resulting temperature distribution as calculated by the **multigrid** function. The second argument of **multigrid**

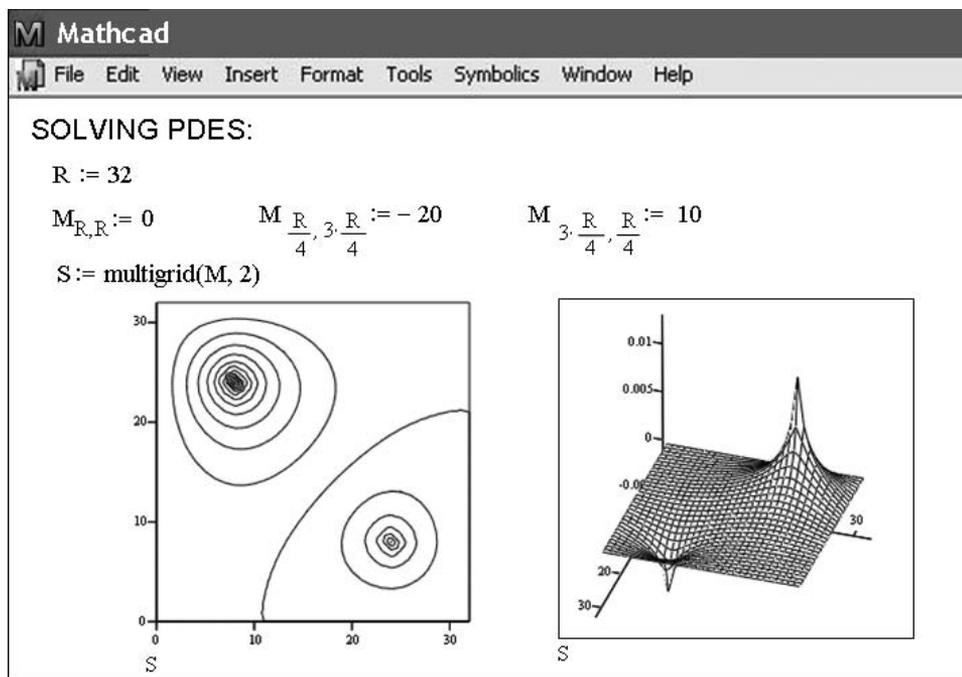


FIGURE 31.16

Mathcad screen to determine the solution of an elliptic PDE.

is a parameter that controls the numerical accuracy. As suggested by Mathcad help, a value of 2 generally gives a good approximation of the solution.

The temperature distribution can be displayed with surface, contour, or vector-field plots. These plots can be placed anywhere on the worksheet by clicking to the desired location. This places a red crosshair at that location. Then, use the Insert/Graph pull-down menu to place an empty plot on the worksheet with placeholders for the expressions to be graphed and for the ranges of variables. Simply type S in the placeholder on the z axis. Mathcad does the rest to produce the graphs shown in Fig. 31.16. Once the graph has been created, you can use the Format/Surface Plot and Format/Contour Plot pull-down menus to change the color or add titles, labels, and other features.

PROBLEMS

31.1 Repeat Example 31.1, but for $T(0, t) = 75$ and $T(10, t) = 150$ and a uniform heat source of 15.

31.2 Repeat Example 31.2, but for boundary conditions of $T(0, t) = 75$ and $T(10, t) = 150$ and a heat source of 15.

31.3 Apply the results of Prob. 31.2 to compute the temperature distribution for the entire rod using the finite-element approach.

31.4 Use Galerkin's method to develop an element equation for a steady-state version of the advection-diffusion equation described

in Prob. 30.7. Express the final result in the format of Eq. (31.26) so that each term has a physical interpretation.

31.5 A version of the Poisson equation that occurs in mechanics is the following model for the vertical deflection of a bar with a distributed load $P(x)$:

$$A_c E \frac{\partial^2 u}{\partial x^2} = P(x)$$

where A_c = cross-sectional area, E = Young's modulus, u = deflection, and x = distance measured along the bar's length. If the bar is rigidly fixed ($u = 0$) at both ends, use the finite-element method to model its deflections for $A_c = 0.1 \text{ m}^2$, $E = 200 \times 10^9 \text{ N/m}^2$, $L = 10 \text{ m}$, and $P(x) = 1000 \text{ N/m}$. Employ a value of $\Delta x = 2 \text{ m}$.

31.6 Develop a user-friendly program to model the steady-state distribution of temperature in a rod with a constant heat source using the finite-element method. Set up the program so that unequally spaced nodes may be used.

31.7 Use Excel to perform the same computation as in Fig. 31.14, but insulate the right-hand edge and add a heat sink of -150 at cell C7.

31.8 Use MATLAB or Mathcad to develop a contour plot with flux arrows for the Excel solution from Prob. 31.7.

31.9 Use Excel to model the temperature distribution of the slab shown in Fig. P31.9. The slab is 0.02 m thick and has a thermal conductivity of $3 \text{ W/(m} \cdot \text{ }^\circ\text{C)}$.

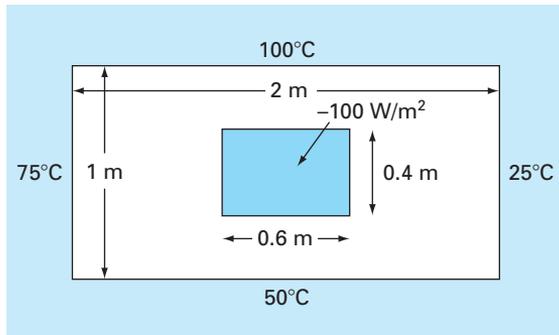


Figure P31.9

31.10 Use MATLAB or Mathcad to develop a contour plot with flux arrows for the Excel solution from Prob. 31.9.

31.11 Find the temperature distribution in a rod (Fig. P31.11) with internal heat generation using the finite-element method. Derive the element nodal equations using Fourier heat conduction

$$q_k = -kA \frac{dT}{\partial x}$$

and heat conservation relationships

$$\sum [q_k + f(x)] = 0$$

where q_k = heat flow (W), k = thermal conductivity ($\text{W/(m} \cdot \text{ }^\circ\text{C)}$), A = cross-sectional area (m^2), and $f(x)$ = heat source (W/cm). The rod has a value of $kA = 100 \text{ W m}^\circ\text{C}$. The rod is 50 cm long, the x -coordinate is zero at the left end, and positive to the right. Divide the rod into 5 elements (6 nodes, each 10 cm long). The left end of

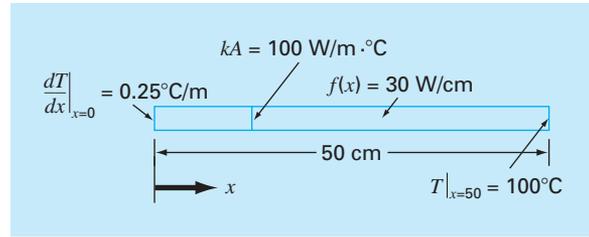


Figure P31.11

the rod has a fixed temperature gradient and the temperature is a variable. The right end has a fixed temperature and the gradient is a variable. The heat source $f(x)$ has a constant value. Thus, the conditions are

$$\left. \frac{dT}{\partial x} \right|_{x=0} = 0.25^\circ\text{C/m} \quad T|_{x=50} = 100^\circ\text{C} \quad f(x) = 30 \text{ W/cm}$$

Develop the nodal equations that must be solved for the temperatures and temperature gradients at each of the six nodes. Assemble the equations, insert the boundary conditions, and solve the resulting set for the unknowns.

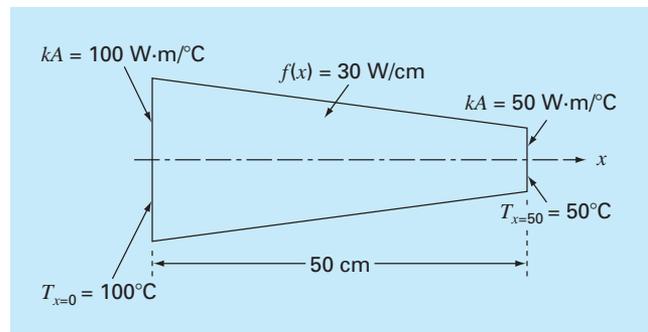
31.12 Find the temperature distribution in a rod (Fig. P31.12) with internal heat generation using the finite-element method. Derive the element nodal equations using Fourier heat conduction.

$$q_k = -kA \frac{dT}{\partial x}$$

and heat conservation relationships

$$\sum [q_k + f(x)] = 0$$

Figure P31.12



where q_k = heat flow (W), k = thermal conductivity (W/(m · °C)), A = cross-sectional area (m²), and $f(x)$ = heat source (W/cm). The rod is 50 cm long, the x -coordinate is zero at the left end, and positive to the right. The rod is also linearly tapered with a value of $kA = 100$ and 50 W m/°C at $x = 0$ and at $x = 50$, respectively. Divide the rod into 5 elements (6 nodes, each 10 cm long). Both ends of the rod have fixed temperatures. The heat source $f(x)$ has a constant value. Thus, the conditions are

$$T|_{x=0} = 100^\circ\text{C} \quad T|_{x=50} = 50^\circ\text{C} \quad f(x) = 30 \text{ W/cm}$$

The tapered areas must be treated as if they were constant over the length of an element. Therefore, average the kA values at each end of the node and take that average as a constant over the node. Develop the nodal equations that must be solved for the temperatures and temperature gradients at each of the six nodes. Assemble the equations, insert the boundary conditions, and solve the resulting set for the unknowns.

31.13 Use a software package to solve for the temperature distribution of the L-shaped plate in Fig. P29.18. Display your results as a contour plot with flux arrows.

Case Studies: Partial Differential Equations

The purpose of this chapter is to apply the methods from Part Eight to practical engineering problems. In *Sec. 32.1*, a parabolic PDE is used to compute the time-variable distribution of a chemical along the longitudinal axes of a rectangular reactor. This example illustrates how the instability of a solution can be due to the nature of the PDE rather than to properties of the numerical method.

Sections 32.2 and 32.3 involve applications of the Poisson and Laplace equations to civil and electrical engineering problems, respectively. Among other things, this will allow you to see similarities as well as differences between field problems in these areas of engineering. In addition, they can be contrasted with the heated-plate problem that has served as our prototype system in this part of the book. *Section 32.2* deals with the deflection of a square plate, whereas *Sec. 32.3* is devoted to computing the voltage distribution and charge flux for a two-dimensional surface with a curved edge.

Section 32.4 presents a finite-element analysis as applied to a series of springs. This application is closer in spirit to finite-element applications in mechanics and structures than was the temperature field problem used to illustrate the approach in Chap. 31.

32.1 ONE-DIMENSIONAL MASS BALANCE OF A REACTOR (CHEMICAL/BIO ENGINEERING)

Background. Chemical engineers make extensive use of idealized reactors in their design work. In Secs. 12.1 and 28.1, we focused on single or coupled well-mixed reactors. These are examples of *lumped-parameter systems* (recall Sec. PT3.1.2).

FIGURE 32.1

An elongated reactor with a single entry and exit point. A mass balance is developed around a finite segment along the tank's longitudinal axis in order to derive a differential equation for the concentration.

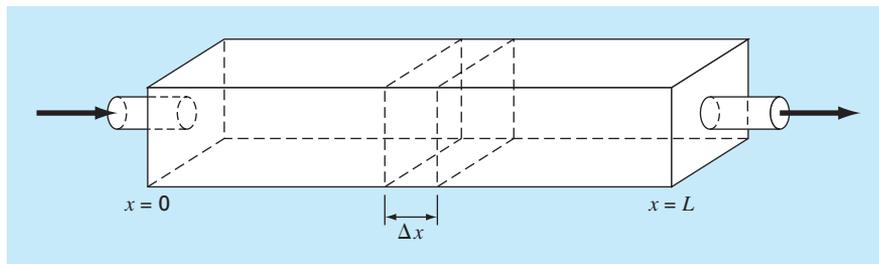


Figure 32.1 depicts an elongated reactor with a single entry and exit point. This reactor can be characterized as a *distributed-parameter system*. If it is assumed that the chemical being modeled is subject to first-order decay¹ and the tank is well-mixed vertically and laterally, a mass balance can be performed on a finite segment of length Δx , as in

$$\begin{aligned}
 V \frac{\Delta c}{\Delta t} = & \underbrace{Qc(x)}_{\text{Flow in}} - \underbrace{Q \left[c(x) + \frac{\partial c(x)}{\partial x} \Delta x \right]}_{\text{Flow out}} - \underbrace{DA_c \frac{\partial c(x)}{\partial x}}_{\text{Dispersion in}} \\
 & + \underbrace{DA_c \left[\frac{\partial c(x)}{\partial x} + \frac{\partial}{\partial x} \frac{\partial c(x)}{\partial x} \Delta x \right]}_{\text{Dispersion out}} - \underbrace{kVc}_{\text{Decay reaction}}
 \end{aligned} \tag{32.1}$$

where V = volume (m^3), Q = flow rate (m^3/h), c is concentration (moles/m^3), D is a dispersion coefficient (m^2/h), A_c is the tank's cross-sectional area (m^2), and k is the first-order decay coefficient (h^{-1}). Note that the dispersion terms are based on *Fick's first law*,

$$\text{Flux} = -D \frac{\partial c}{\partial x} \tag{32.2}$$

which is directly analogous to Fourier's law for heat conduction [recall Eq. (29.4)]. It specifies that turbulent mixing tends to move mass from regions of high to low concentration. The parameter D , therefore, reflects the magnitude of turbulent mixing.

If Δx and Δt are allowed to approach zero, Eq. (32.1) becomes

$$\frac{\partial c}{\partial t} = D \frac{\partial^2 c}{\partial x^2} - U \frac{\partial c}{\partial x} - kc \tag{32.3}$$

where $U = Q/A_c$ is the velocity of the water flowing through the tank. The mass balance for Fig. 32.1 is, therefore, now expressed as a parabolic partial differential equation. Equation (32.3) is sometimes referred to as the *advection-dispersion equation* with first-order reaction. At steady state, it is reduced to a second-order ODE,

$$0 = D \frac{d^2 c}{dx^2} - U \frac{dc}{dx} - kc \tag{32.4}$$

Prior to $t = 0$, the tank is filled with water that is devoid of the chemical. At $t = 0$, the chemical is injected into the reactor's inflow at a constant level of c_{in} . Thus, the following boundary conditions hold:

$$Qc_{\text{in}} = Qc_0 - DA_c \frac{\partial c_0}{\partial x}$$

and

$$c'(L, t) = 0$$

The second condition specifies that the chemical leaves the reactor purely as a function of flow through the outlet pipe. That is, it is assumed that dispersion in the reactor does not

¹That is, the chemical decays at a rate that is linearly proportional to how much chemical is present.

affect the exit rate. Under these conditions, use numerical methods to solve Eq. (32.4) for the steady-state levels in the reactor. Note that this is an ODE boundary-value problem. Then solve Eq. (32.3) to characterize the transient response—that is, how the levels change in time as the system approaches the steady state. This application involves a PDE.

Solution. A steady-state solution can be developed by substituting centered finite differences for the first and the second derivatives in Eq. (32.4) to give

$$0 = D \frac{c_{i+1} - 2c_i + c_{i-1}}{\Delta x^2} - U \frac{c_{i+1} - c_{i-1}}{2 \Delta x} - kc_i$$

Collecting terms gives

$$-\left(\frac{D}{U \Delta x} + \frac{1}{2}\right)c_{i-1} + \left(\frac{2D}{U \Delta x} + \frac{k \Delta x}{U}\right)c_0 - \left(\frac{D}{U \Delta x} - \frac{1}{2}\right)c_{i+1} = 0 \quad (32.5)$$

This equation can be written for each of the system's nodes. At the reactor's ends, this process introduces nodes that lie outside the system. For example, at the inlet node ($i = 0$),

$$-\left(\frac{D}{U \Delta x} + \frac{1}{2}\right)c_{-1} + \left(\frac{2D}{U \Delta x} + \frac{k \Delta x}{U}\right)c_0 - \left(\frac{D}{U \Delta x} - \frac{1}{2}\right)c_1 = 0 \quad (32.6)$$

The c_{-1} can be removed by invoking the first boundary condition. At the inlet, the following mass balance must hold:

$$Qc_{\text{in}} = Qc_0 - DA_c \frac{\partial c_0}{\partial x}$$

where $c_0 =$ concentration at $x = 0$. Thus, this boundary condition specifies that the amount of chemical carried into the tank by advection through the pipe must be equal to the amount carried away from the inlet by both advection and turbulent dispersion in the tank. A finite divided difference can be substituted for the derivative

$$Qc_{\text{in}} = Qc_0 - DA_c \frac{c_1 - c_{-1}}{2 \Delta x}$$

which can be solved for

$$c_{-1} = c_1 + \frac{2 \Delta x U}{D} c_{\text{in}} - \frac{2 \Delta x U}{D} c_0$$

which can be substituted into Eq. (32.6) to give

$$\left(\frac{2D}{U \Delta x} + \frac{k \Delta x}{U} + 2 + \frac{\Delta x U}{D}\right)c_0 - \left(\frac{2D}{U \Delta x}\right)c_1 = \left(2 + \frac{\Delta x U}{D}\right)c_{\text{in}} \quad (32.7)$$

A similar exercise can be performed for the outlet, where the original difference equation is

$$-\left(\frac{D}{U \Delta x} + \frac{1}{2}\right)c_{n-1} + \left(\frac{2D}{U \Delta x} + \frac{k \Delta x}{U}\right)c_n - \left(\frac{D}{U \Delta x} - \frac{1}{2}\right)c_{n+1} = 0 \quad (32.8)$$

The boundary condition at the outlet is

$$Qc_n - DA_c \frac{dc_n}{dx} = Qc_n$$

As with the inlet, a divided difference can be used to approximate the derivative.

$$Qc_n - DA_c \frac{c_{n+1} - c_{n-1}}{2\Delta x} = Qc_n \quad (32.9)$$

Inspection of this equation leads us to conclude that $c_{n+1} = c_{n-1}$. In other words, the slope at the outlet must be zero for Eq. (32.9) to hold. Substituting this result into Eq. (32.8) and simplifying gives

$$-\left(\frac{2D}{U\Delta x}\right)c_{n-1} + \left(\frac{2D}{U\Delta x} + \frac{k\Delta x}{U}\right)c_n = 0 \quad (32.10)$$

Equations (32.5), (32.7), and (32.10) now form a system of n tridiagonal equations with n unknowns. For example, if $D = 2$, $U = 1$, $\Delta x = 2.5$, $k = 0.2$, and $c_{\text{in}} = 100$, the system is

$$\begin{bmatrix} 5.35 & -1.6 & & & \\ -1.3 & 2.1 & -0.3 & & \\ & -1.3 & 2.1 & -0.3 & \\ & & -1.3 & 2.1 & -0.3 \\ & & & -1.6 & 2.1 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix} = \begin{bmatrix} 325 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

which can be solved for

$$\begin{aligned} c_0 &= 76.44 & c_1 &= 52.47 & c_2 &= 36.06 \\ c_3 &= 25.05 & c_4 &= 19.09 \end{aligned}$$

These results are plotted in Fig. 32.2. As expected, the concentration decreases due to the decay reaction as the chemical flows through the tank. In addition to the above computation, Fig. 32.2 shows another case with $D = 4$. Notice how increasing the turbulent mixing tends to flatten the curve.

FIGURE 32.2

Concentration versus distance along the longitudinal axis of a rectangular reactor for a chemical that decays with first-order kinetics.

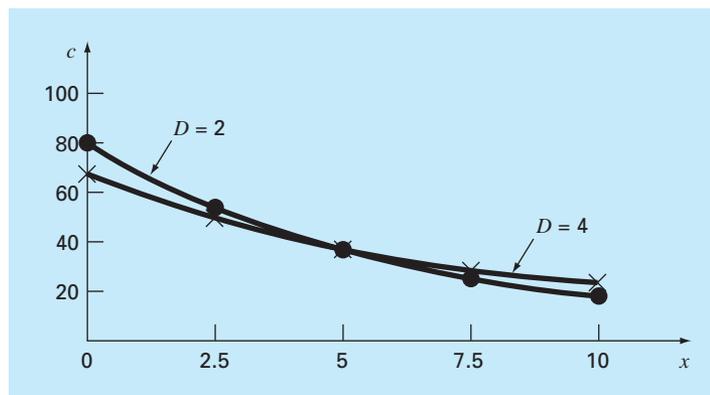
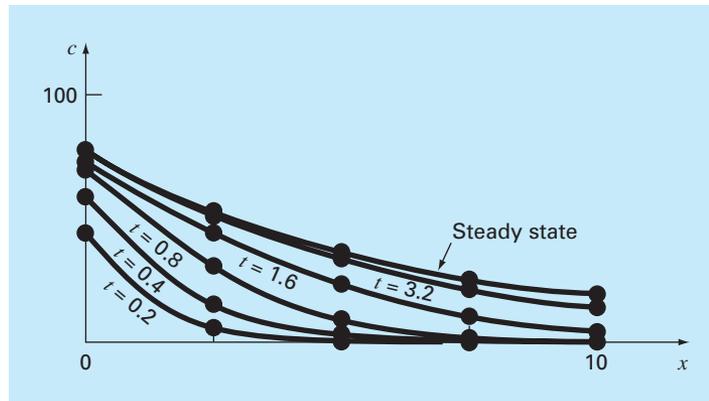


FIGURE 32.3

Concentration versus distance at different times during the buildup of chemical in a reactor.



In contrast, if dispersion is decreased, the curve would become steeper as mixing became less important relative to advection and decay. It should be noted that if dispersion is decreased too much, the computation will become subject to numerical errors. This type of error is referred to as *static instability* to contrast it with the *dynamic instability* due to too large a time step during a dynamic computation. The criterion to avoid this static instability is

$$\Delta x \leq \frac{2D}{U}$$

Thus, the criterion becomes more stringent (lower Δx) for cases where advection dominates over dispersion.

Aside from steady-state computations, numerical methods can be used to generate time-variable solutions of Eq. (32.3). Figure 32.3 shows results for $D = 2$, $U = 1$, $\Delta x = 2.5$, $k = 0.2$, and $c_{in} = 100$, where the concentration in the tank is 0 at time zero. As expected, the immediate impact is near the inlet. With time, the solution eventually approaches the steady-state level.

It should be noted that in such dynamic calculations, the time step is constrained by a stability criterion expressed as (Chapra, 1997)

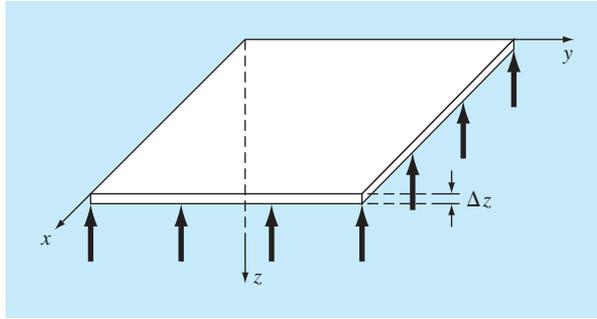
$$\Delta t \leq \frac{(\Delta x)^2}{2D + k(\Delta x)^2}$$

Thus, the reaction term acts to make the time step smaller.

32.2 DEFLECTIONS OF A PLATE (CIVIL/ENVIRONMENTAL ENGINEERING)

Background. A square plate with simply supported edges is subject to an areal load q (Fig. 32.4). The deflection in the z dimension can be determined by solving the elliptic PDE (see Carnahan, Luther, and Wilkes, 1969)

$$\frac{\partial^4 z}{\partial x^4} + 2 \frac{\partial^4 z}{\partial x^2 \partial y^2} + \frac{\partial^4 z}{\partial y^4} = \frac{q}{D} \quad (32.11)$$

**FIGURE 32.4**

A simply supported square plate subject to an areal load.

subject to the boundary conditions that, at the edges, the deflection and slope normal to the boundary are zero. The parameter D is the flexural rigidity,

$$D = \frac{E \Delta z^3}{12(1 - \sigma^2)} \quad (32.12)$$

where E = the modulus of elasticity, Δz = the plate's thickness, and σ = Poisson's ratio.

If a new variable is defined as

$$u = \frac{\partial^2 z}{\partial x^2} + \frac{\partial^2 z}{\partial y^2}$$

Eq. (32.11) can be reexpressed as

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = \frac{q}{D} \quad (32.13)$$

Therefore, the problem reduces to successively solving two Poisson equations. First, Eq. (32.13) can be solved for u subject to the boundary condition that $u = 0$ at the edges. Then, the results can be employed in conjunction with

$$\frac{\partial^2 z}{\partial x^2} + \frac{\partial^2 z}{\partial y^2} = u \quad (32.14)$$

to solve for z subject to the condition that $z = 0$ at the edges.

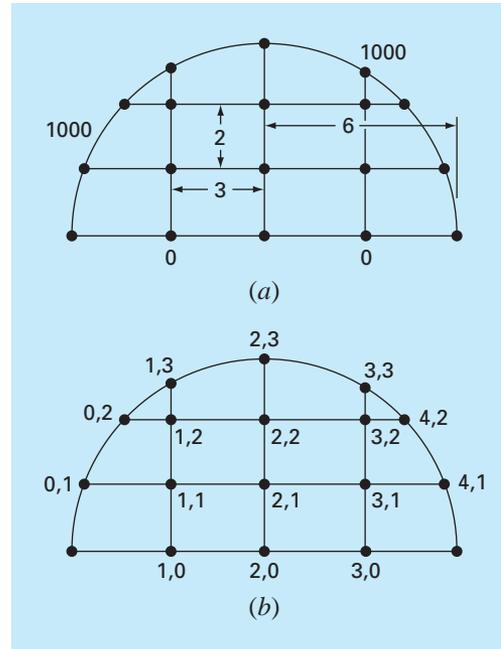
Develop a computer program to determine the deflections for a square plate subject to a constant areal load. Test the program for a plate with 2-m-long edges, $q = 33.6 \text{ kN/m}^2$, $\sigma = 0.3$, $\Delta z = 10^{-2} \text{ m}$, and $E = 2 \times 10^{11} \text{ Pa}$. Employ $\Delta x = \Delta y = 0.5 \text{ m}$ for your test run.

Solution. Finite-divided differences can be substituted into Eq. (32.13) to give

$$\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta y^2} = \frac{q}{D} \quad (32.15)$$

Equation (32.12) can be used to compute $D = 1.832 \times 10^4 \text{ N/m}$. This result, along with the other system parameters, can be substituted into Eq. (32.15) to give

$$u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{i,j} = 0.458$$

**FIGURE 32.5**

(a) A two-dimensional system with a voltage of 1000 along the circular boundary and a voltage of 0 along the base. (b) The nodal numbering scheme.

Finally, for regions containing no free charge (that is $\rho_v = 0$), Eq. (32.16) reduces to a Laplace equation,

$$\frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2} = 0 \quad (32.17)$$

Employ numerical methods to solve Eq. (32.17) for the situation depicted in Fig. 32.5. Compute both the values for V and for D if $\epsilon = 2$.

Solution. Using the approach outlined in Sec. 29.3.2, Eq. (29.24) can be written for node (1, 1) as

$$\frac{2}{\Delta x^2} \left[\frac{V_{1,1} - V_{0,1}}{\alpha_1(\alpha_1 + \alpha_2)} + \frac{V_{1,1} - V_{2,1}}{\alpha_2(\alpha_1 + \alpha_2)} \right] + \frac{2}{\Delta y^2} \left[\frac{V_{1,1} - V_{0,1}}{\beta_1(\beta_1 + \beta_2)} + \frac{V_{1,1} - V_{2,1}}{\beta_2(\beta_1 + \beta_2)} \right] = 0$$

According to the geometry depicted in Fig. 32.5, $\Delta x = 3$, $\Delta y = 2$, $\beta_1 = \beta_2 = \alpha_2 = 1$, and $\alpha_1 = 0.94281$. Substituting these values yields

$$0.12132V_{1,1} - 121.32 + 0.11438V_{1,1} - 0.11438V_{2,1} + 0.25V_{1,1} \\ + 0.25V_{1,1} - 0.25V_{1,2} = 0$$

Collecting terms gives

$$0.73570V_{1,1} - 0.11438V_{2,1} - 0.25V_{1,2} = 121.32$$

A similar approach can be applied to the remaining interior nodes. The resulting simultaneous equations can be expressed in matrix form as

$$\begin{bmatrix} 0.73570 & -0.11438 & & -0.25000 & & \\ -0.11111 & 0.72222 & -0.11111 & & -0.25000 & \\ & -0.11438 & 0.73570 & & & -0.25000 \\ -0.31288 & & & 1.28888 & -0.14907 & \\ & -0.25000 & & -0.11111 & 0.72222 & -0.11111 \\ & & -0.31288 & & -0.14907 & 1.28888 \end{bmatrix} \times \begin{Bmatrix} V_{1,1} \\ V_{2,1} \\ V_{3,1} \\ V_{1,2} \\ V_{2,2} \\ V_{3,2} \end{Bmatrix} = \begin{Bmatrix} 121.32 \\ 0 \\ 121.32 \\ 826.92 \\ 250 \\ 826.92 \end{Bmatrix}$$

which can be solved for

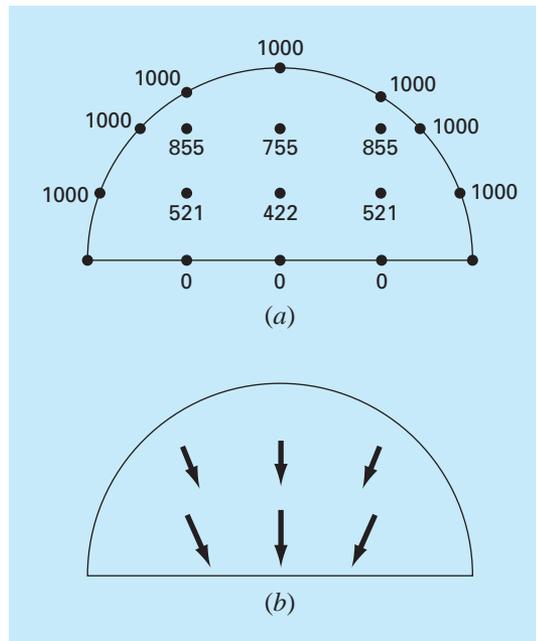
$$\begin{aligned} V_{1,1} &= 521.19 & V_{2,1} &= 421.85 & V_{3,1} &= 521.19 \\ V_{1,2} &= 855.47 & V_{2,2} &= 755.40 & V_{3,2} &= 855.47 \end{aligned}$$

These results are depicted in Fig. 32.6a.

To compute the flux (recall Sec. 29.2.3), Eqs. (29.14) and (29.15) must be modified to account for the irregular boundaries. For the present example, the modifications

FIGURE 32.6

The results of solving the Laplace equation with correction factors for the irregular boundaries. (a) Potential and (b) flux.



result in

$$D_x = -\varepsilon \frac{V_{i+1,j} - V_{i-1,j}}{(\alpha_1 + \alpha_2) \Delta x}$$

and

$$D_y = -\varepsilon \frac{V_{i,j+1} - V_{i,j-1}}{(\beta_1 + \beta_2) \Delta y}$$

For node (1, 1), these formulas can be used to compute the x and y components of the flux

$$D_x = -2 \frac{421.85 - 1000}{(0.94281 + 1)3} = 198.4$$

and

$$D_y = -2 \frac{855.47 - 0}{(1 + 1)2} = -427.7$$

which in turn can be used to calculate the electric flux density vector

$$D = \sqrt{198.4^2 + (-427.7)^2} = 471.5$$

with a direction of

$$\theta = \tan^{-1} \left(\frac{-427.7}{198.4} \right) = -65.1^\circ$$

The results for the other nodes are

Node	D_x	D_y	D	θ
2, 1	0.0	-377.7	377.7	-90
3, 1	-198.4	-427.7	471.5	245.1
1, 2	109.4	-299.6	281.9	-69.1
2, 2	0.0	-289.1	289.1	-90.1
3, 2	-109.4	-299.6	318.6	249.9

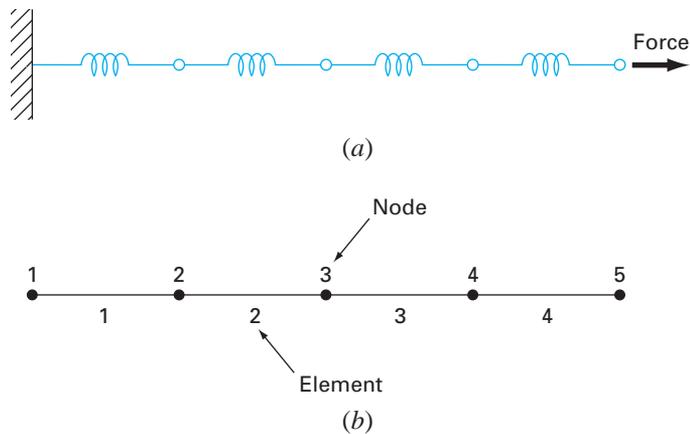
The fluxes are displayed in Fig. 32.6b.

32.4 FINITE-ELEMENT SOLUTION OF A SERIES OF SPRINGS (MECHANICAL/AEROSPACE ENGINEERING)

Background. Figure 32.7 shows a series of interconnected springs. One end is fixed to a wall, whereas the other is subject to a constant force F . Using the step-by-step procedure outlined in Chap. 31, a finite-element approach can be employed to determine the displacements of the springs.

Solution.

Discretization. The way to partition this system is obviously to treat each spring as an element. Thus, the system consists of four elements and five nodes (Fig. 32.7b).

**FIGURE 32.7**

(a) A series of interconnected springs. One end is fixed to a wall, whereas the other is subject to a constant force F . (b) The finite-element representation. Each spring represents an element. Therefore, the system consists of four elements and five nodes.

**FIGURE 32.8**

A free-body diagram of a spring system.

Element equations. Because this system is so simple, its element equations can be written directly without recourse to mathematical approximations. This is an example of the direct approach for deriving elements.

Figure 32.8 shows an individual element. The relationship between force F and displacement x can be represented mathematically by Hooke's law:

$$F = kx$$

where k = the spring constant, which can be interpreted as the force required to cause a unit displacement. If a force F_1 is applied at node 1, the following force balance must hold:

$$F = k(x_1 - x_2)$$

where x_1 = displacement of node 1 from its equilibrium position and x_2 = displacement of node 2 from its equilibrium position. Thus, $x_2 - x_1$ represents how much the spring is elongated or compressed relative to equilibrium (Fig. 32.8).

This equation can also be written as

$$F_1 = kx_1 - kx_2$$

For a stationary system, a force balance also necessitates that $F_1 = -F_2$ and, therefore,

$$F_2 = -kx_1 + kx_2$$

and $\{x'\}$ and $\{F'\}$ are the expanded displacement and force vectors, respectively. Notice that, as the equations were assembled, the internal forces cancel. Thus, the final result for $\{F'\}$ has zeros for all but the first and last nodes.

Before proceeding to the next step, we must comment on the structure of the assemblage property matrix [Eq. (32.19)]. Notice that the matrix is tridiagonal. This is a direct result of the particular global numbering scheme that was chosen (Table 31.1) prior to assemblage. Although it is not very important in the present context, the attainment of such a banded, sparse system can be a decided advantage for more complicated problem settings. This is due to the efficient schemes that are available for solving such systems.

Boundary Conditions. The present system is subject to a single boundary condition, $x_1 = 0$. Introduction of this condition and applying the global renumbering scheme reduces the system to ($k's = 1$)

$$\begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & -1 & 2 & -1 & \\ & & -1 & 2 & -1 \\ & & & -1 & 1 \end{bmatrix} \begin{Bmatrix} x_2 \\ x_3 \\ x_4 \\ x_5 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ F \end{Bmatrix}$$

The system is now in the form of Eq. (31.11) and is ready to be solved.

Although reduction of the equations is certainly a valid approach for incorporating boundary conditions, it is usually preferable to leave the number of equations intact when performing the solution on the computer. Whatever the method, once the boundary conditions are incorporated, we can proceed to the next step—the solution.

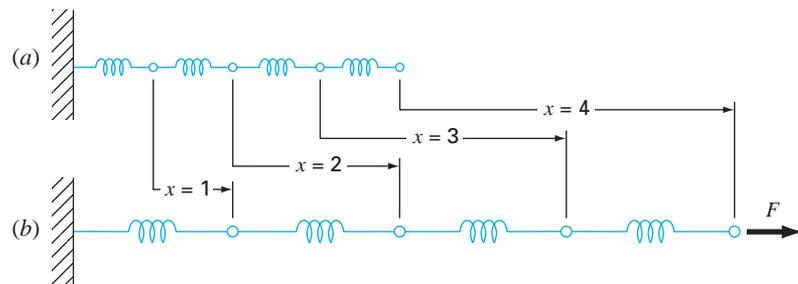
Generating Solution. Using one of the approaches from Part Three, such as the efficient tridiagonal solution technique delineated in Chap. 11, the system can be solved for (with all $k's = 1$ and $F = 1$)

$$x_2 = 1 \quad x_3 = 2 \quad x_4 = 3 \quad x_5 = 4$$

Postprocessing. The results can now be displayed graphically. As in Fig. 32.9, the results are as expected. Each spring is elongated a unit displacement.

FIGURE 32.9

(a) The original spring system. (b) The system after the application of a constant force. The displacements are indicated in the space between the two systems.



PROBLEMS

Chemical/Bio Engineering

32.1 Perform the same computation as in Sec. 32.1, but use $\Delta x = 1.25$.

32.2 Develop a finite-element solution for the steady-state system of Sec. 32.1.

32.3 Compute mass fluxes for the steady-state solution of Sec. 32.1 using Fick's first law.

32.4 Compute the steady-state distribution of concentration for the tank shown in Fig. P32.4. The PDE governing this system is

$$D \left(\frac{\partial^2 c}{\partial x^2} + \frac{\partial^2 c}{\partial y^2} \right) - kc = 0$$

and the boundary conditions are as shown. Employ a value of 0.5 for D and 0.1 for k .

32.5 Two plates are 10 cm apart, as shown in Fig. P32.5. Initially, both plates and the fluid are still. At $t = 0$, the top plate is moved at a constant velocity of 8 cm/s. The equations governing the motions of the fluids are

$$\frac{\partial v_{\text{oil}}}{\partial t} = \mu_{\text{oil}} \frac{\partial^2 v_{\text{oil}}}{\partial x^2} \quad \text{and} \quad \frac{\partial v_{\text{water}}}{\partial t} = \mu_{\text{water}} \frac{\partial^2 v_{\text{water}}}{\partial x^2}$$

and the following relationships hold true at the oil-water interface:

$$v_{\text{oil}} = v_{\text{water}} \quad \text{and} \quad \mu_{\text{oil}} \frac{\partial v_{\text{oil}}}{\partial x} = \mu_{\text{water}} \frac{\partial v_{\text{water}}}{\partial x}$$

What is the velocity of the two fluid layers at $t = 0.5, 1, \text{ and } 1.5$ s at distances $x = 2, 4, 6, \text{ and } 8$ cm from the bottom plate? Note that μ_{water} and $\mu_{\text{oil}} = 1$ and 3 cp, respectively.

32.6 The displacement of a uniform membrane subject to a tension and a uniform pressure can be described by the Poisson equation

$$\frac{\partial^2 z}{\partial x^2} + \frac{\partial^2 z}{\partial y^2} = -\frac{P}{T}$$

Solve for the displacement of a 1-cm-square membrane that has $P/T = 0.6/\text{cm}$ and is fastened so that it has zero displacement along its four boundaries. Employ $\Delta x = \Delta y = 0.1$ cm. Display your results as a contour plot.

Civil/Environmental Engineering

32.7 Perform the same computation as in Sec. 32.2, but use $\Delta x = \Delta y = 0.4$ m.

32.8 The flow through porous media can be described by the Laplace equation

$$\frac{\partial^2 h}{\partial x^2} + \frac{\partial^2 h}{\partial y^2} = 0$$

where h is head. Use numerical methods to determine the distribution of head for the system shown in Fig. P32.8.

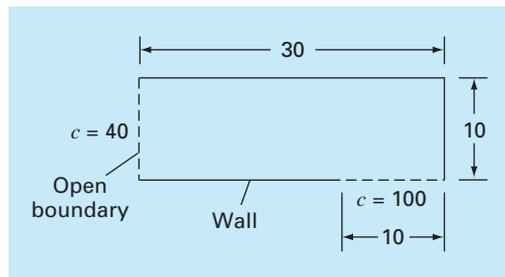


Figure P32.4

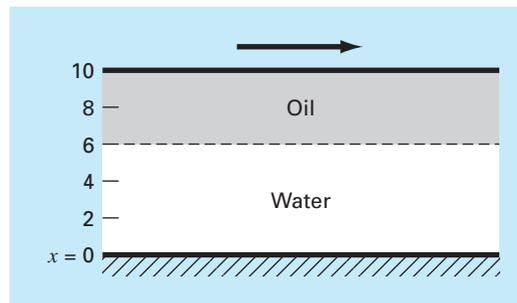


Figure P32.5

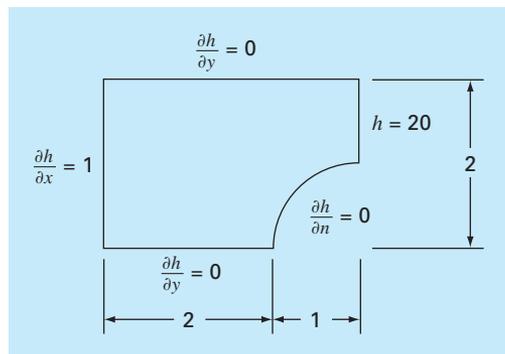


Figure P32.8

32.9 The velocity of water flow through the porous media can be related to head by D’Arcy’s law

$$q_n = -K \frac{dh}{dn}$$

where K is the hydraulic conductivity and q_n is discharge velocity in the n direction. If $K = 5 \times 10^{-4}$ cm/s, compute the water velocities for Prob. 32.8.

Electrical Engineering

32.10 Perform the same computation as in Sec. 32.3 but for the system depicted in Fig. P32.10.

32.11 Perform the same computation as in Sec. 32.3 but for the system depicted in Fig. P32.11.

Figure P32.10

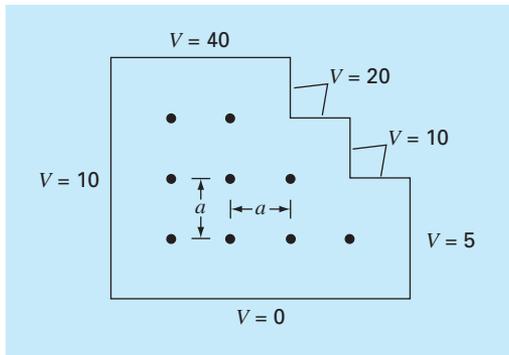
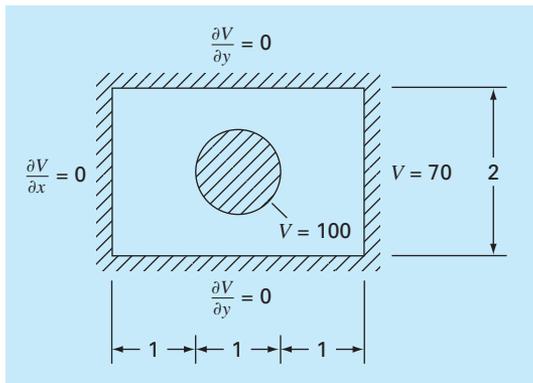


Figure P32.11



32.12 Use Poisson’s equation to compute the electric potential over a unit square (1×1) plate with zero voltage at the edges and point charge sources of $\rho_v/\epsilon(0.5, 0.5) = 1$ and $\rho_v/\epsilon(-0.5, -0.5) = -1$. Employ $\Delta x = \Delta y = 0.1$ and display your results as a contour plot.

Mechanical/Aerospace Engineering

32.13 Perform the same computation as in Sec. 32.4, but change the force to 1.5 and the spring constants to

Spring	1	2	3	4
k	0.75	1.5	0.5	2

32.14 Perform the same computation as in Sec. 32.4, but use a force of 2 and five springs with

Spring	1	2	3	4	5
k	0.25	0.5	1.5	0.75	1

32.15 An insulated composite rod is formed of two parts arranged end to end, and both halves are of equal length. Part a has thermal conductivity k_a , for $0 \leq x \leq 1/2$, and part b has thermal conductivity k_b , for $1/2 \leq x \leq 1$. The nondimensional transient heat conduction equations that describe the temperature u over the length x of the composite rod are

$$\frac{\partial^2 u}{\partial x^2} = \frac{\partial u}{\partial t} \quad 0 \leq x \leq 1/2$$

$$r \frac{\partial^2 u}{\partial x^2} = \frac{\partial u}{\partial t} \quad 1/2 \leq x \leq 1$$

where u = temperature, x = axial coordinate, t = time, and $r = k_a/k_b$. The boundary and initial conditions are

Boundary conditions	$u(0, t) = 1$	$u(1, t) = 1$
	$\left(\frac{\partial u}{\partial x}\right)_a = \left(\frac{\partial u}{\partial x}\right)_b$	$x = 1/2$
Initial conditions	$u(x, 0) = 0$	$0 < x < 1$

Solve this set of equations for the temperature distribution as a function of time. Use second-order accurate finite-difference analogues for the derivatives with a Crank-Nicolson formulation to integrate in time. Write a computer program for the solution, and select values of Δx and Δt for good accuracy. Plot the temperature u versus length x for various values of time t . Generate a separate curve for the following values of the parameter $r = 1, 0.1, 0.01, 0.001$, and 0.

32.16 Solve the nondimensional transient heat conduction equation in two dimensions, which represents the transient temperature distribution in an insulated plate. The governing equation is

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = \frac{\partial u}{\partial t}$$

where u = temperature, x and y are spatial coordinates, and t = time. The boundary and initial conditions are

Boundary conditions	$u(x, 0, t) = 0$	$u(x, 1, t) = 1$	
	$u(0, y, t) = 0$	$u(1, y, t) = 1$	
Initial condition	$u(x, y, 0) = 0$	$0 \leq x < 1$	$0 \leq y < 1$

Solve using the alternating direction-implicit technique. Write a computer program to implement the solution. Plot the results using a three-dimensional plotting routine where the horizontal plan contains the x and y axes and the z axis is the dependent variable u . Construct several plots at various times, including the following: **(a)** the initial conditions; **(b)** one intermediate time, approximately halfway to steady state; and **(c)** the steady-state condition.

EPILOGUE: PART EIGHT

PT8.3 TRADE-OFFS

The primary trade-offs associated with numerical methods for the solution of partial differential equations involve choosing between *finite-difference* and *finite-element* approaches. The finite-difference methods are conceptually easier to understand. In addition, they are easy to program for systems that can be approximated with uniform grids. However, they are difficult to apply to systems with complicated geometries.

Finite-difference approaches can be divided into categories depending on the type of PDE that is being solved. *Elliptic PDEs* can be approximated by a set of linear algebraic equations. Consequently, the *Liebmann method* (which, in fact, is Gauss-Seidel) can be employed to obtain a solution iteratively.

One-dimensional parabolic PDEs can be solved in two fundamentally different ways: explicit or implicit approaches. The *explicit method* steps out in time in a fashion that is similar to Euler's technique for solving ODEs. It has the advantage that it is simple to program but has the shortcoming of a very stringent stability criterion. In contrast, stable implicit methods are available. These typically involve solving simultaneous tridiagonal algebraic equations at each time step. One of these approaches, the *Crank-Nicolson method*, is both accurate and stable and, therefore, is widely used for one-dimensional linear parabolic problems.

Two-dimensional parabolic PDEs can also be modeled explicitly. However, their stability constraints are even more severe than for the one-dimensional case. Special implicit approaches, which are generally referred to as splitting methods, have been developed to circumvent this shortcoming. These approaches are both efficient and stable. One of the most common is the *ADI*, or *alternating-direction implicit*, method.

All the above *finite-difference* approaches become unwieldy when applied to systems involving nonuniform shapes and heterogeneous conditions. Finite-element methods are available that handle such systems in a superior fashion.

Although the *finite-element method* is based on some fairly straightforward ideas, the mechanics of generating a good finite-element code for two- and three-dimensional problems is not a trivial exercise. In addition, it can be computationally expensive for large problems. However, it is vastly superior to finite-difference approaches for systems involving complicated shapes. Consequently, its expense and conceptual "overhead" are often justified because of the detail of the final solution.

PT8.4 IMPORTANT RELATIONSHIPS AND FORMULAS

Table PT8.3 summarizes important information that was presented regarding the finite-difference methods in Part Eight. This table can be consulted to quickly access important relationships and formulas.

TABLE PT8.3 Summary of finite-difference methods.

	Computational Molecule	Equation
Elliptic PDEs Liebmann's method		$T_{i,j} = \frac{T_{i+1,j} + T_{i-1,j} + T_{i,j+1} + T_{i,j-1}}{4}$
Parabolic PDEs (one-dimensional) Explicit method		$T_i^{l+1} = T_i^l + \lambda(T_{i+1}^l - 2T_i^l + T_{i-1}^l)$
Implicit method		$-\lambda T_{i-1}^{l+1} + (1 + 2\lambda)T_i^{l+1} - \lambda T_{i+1}^{l+1} = T_i^l$
Crank-Nicolson method		$-\lambda T_{i-1}^{l+1} + 2(1 + \lambda)T_i^{l+1} - \lambda T_{i+1}^{l+1} \\ = \lambda T_{i-1}^l + 2(1 - \lambda)T_i^l + \lambda T_{i+1}^l$

PT8.5 ADVANCED METHODS AND ADDITIONAL REFERENCES

Carnahan, Luther, and Wilkes (1969); Rice (1983); Ferziger (1981); and Lapidus and Pinder (1982) provide useful surveys of methods and software for solving PDEs. You can also consult Ames (1977), Gladwell and Wait (1979), Vichnevetsky (1981, 1982), and Zienkiewicz (1971) for more in-depth treatments. Additional information on the finite-element method can be found in Allaire (1985), Huebner and Thornton (1982), Stasa (1985), and Baker (1983). Aside from elliptic and hyperbolic PDEs, numerical methods are also available to solve hyperbolic equations. Nice introductions and summaries of some of these methods can be found in Lapidus and Pinder (1981), Ferziger (1981), Forsythe and Wasow (1960), and Hoffman (1992).