10

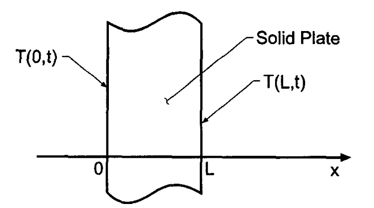
Parabolic Partial Differential Equations

Examples

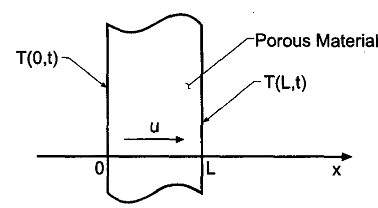
10.1 INTRODUCTION

Figure 10.1 illustrates two heat diffusion problems. The plate illustrated at the top of the figure has a thickness $L = 1.0$ cm and thermal diffusivity $\alpha = 0.01\ \text{cm}^2/\text{s}$. The internal temperature distribution is governed by the unsteady one-dimensional heat diffusion equation:

$$T_t = \alpha T_{xx} \tag{10.1}$$



$T_t = \alpha T_{xx}$, $T(x,0) = F(x)$, $T(x,t) = ?$



$T_t + uT_x = \alpha T_{xx}$, $T(x,0) = F(x)$, $T(x,t) = ?$

Figure 10.1 Unsteady heat diffusion problems.

The plate is heated to an initial temperature distribution, $T(x, 0)$, at which time the heat source is turned off. The initial temperature distribution in the plate is specified by

$$T(x, 0.0) = 200.0x \qquad 0.0 \le x \le 0.5 \tag{10.2a}$$

$$T(x, 0.0) = 200.0(1.0 - x) \qquad 0.5 \le x \le 1.0 \tag{10.2b}$$

where T is measured in degrees Celcius (C). This initial temperature distribution is illustrated by the top curve in Figure 10.2. The temperatures on the two faces of the plate are held at 0.0 C for all time. Thus,

$$T(0.0, t) = T(1.0, t) = 0.0 \tag{10.2c}$$

The temperature distribution within the plate, $T(x, t)$, is required.

The exact solution to this problem is obtained by assuming a product solution of the form $T(x, t) = X(x)\hat{T}(t)$, substituting this functional form into the PDE and separating variables, integrating the two resulting ordinary differential equations for $X(x)$ and $\hat{T}(t)$, applying the boundary conditions at $x = 0$ and $x = L$, and superimposing an infinite

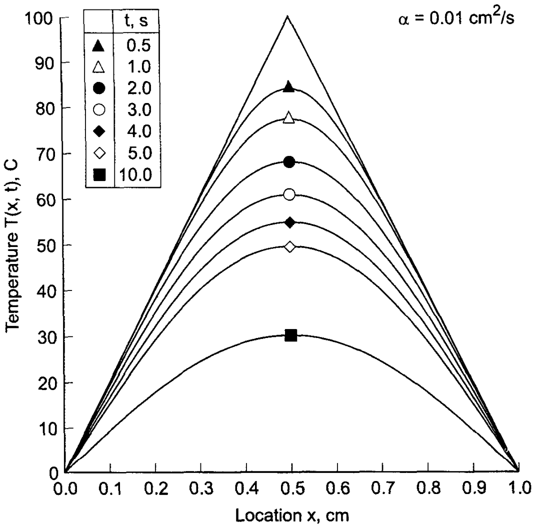


Figure 10.2 Exact solution of the heat diffusion problem.

number of harmonic functions (i.e., sines and cosines) in a Fourier series to satisfy the initial conditions $T(x, 0)$. The result is

$$\boxed{T(x,t) = \frac{800}{\pi^2}\sum_{m=0}^{\infty}\frac{(-1)^m}{(2m+1)^2}\sin[(2m+1)\pi x]e^{-(2m+1)^2\pi^2\alpha t}} \tag{10.3}$$

The exact solution at selected values of time is tabulated in Table 10.1 and illustrated in Figure 10.2. The solution is symmetrical about the midplane of the plate. The solution smoothly approaches the asymptotic steady state solution, $T(x, \infty) = 0.0$.

The second problem illustrated in Figure 10.1 is a combined convection-diffusion problem, which is governed by the convection-diffusion equation. This problem is similar to the first problem, with the added feature that the plate is porous and a cooling fluid flows through the plate. The exact solution and the numerical solution of this problem are presented in Section 10.10.

A wide variety of parabolic partial differential equations are encountered in engineering and science. Two of the more common ones are the *diffusion equation* and the *convection-diffusion equation*, presented below for the generic dependent variable $f(x, t)$:

$$\boxed{f_t = \alpha f_{xx}} \tag{10.4}$$

$$\boxed{f_t + u f_x = \alpha f_{xx}} \tag{10.5}$$

Table 10.1 Exact Solution of the Heat Diffusion Problem

	Temperature $T(x, t)$, C					
t, s	$x = 0.0$	$x = 0.1$	$x = 0.2$	$x = 0.3$	$x = 0.4$	$x = 0.5$
0.0	0.0	20.0000	40.0000	60.0000	80.0000	100.0000
0.5	0.0	19.9997	39.9847	59.6604	76.6674	84.0423
1.0	0.0	19.9610	39.6551	57.9898	72.0144	77.4324
2.0	0.0	19.3513	37.6601	53.3353	64.1763	68.0846
3.0	0.0	18.1235	34.8377	48.5749	57.7018	60.9128
4.0	0.0	16.6695	31.8585	44.1072	52.0966	54.8763
5.0	0.0	15.2059	28.9857	40.0015	47.1255	49.5912
10.0	0.0	9.3346	17.7561	24.4405	28.7327	30.2118
20.0	0.0	3.4794	6.6183	9.1093	10.7086	11.2597
50.0	0.0	0.1801	0.3427	0.4716	0.5544	0.5830
∞	0.0	0.0000	0.0000	0.0000	0.0000	0.0000

where α is the diffusivity and u is the convection velocity. The diffusion equation applies to problems in mass diffusion, heat diffusion (i.e., conduction), neutron diffusion, etc. The convection-diffusion equation applies to problems in which convection occurs in combination with diffusion, for example, fluid mechanics and heat transfer. The present chapter is devoted mainly to the numerical solution of the diffusion equation. All of the results also apply to the numerical solution of the convection-diffusion equation, which is considered briefly in Section 10.10.

The solution of Eqs. (10.4) and (10.5) is the function $f(x, t)$. This function must satisfy an initial condition at $t = 0, f(x, 0) = F(x)$. The time coordinate has an unspecified (i.e., open) final value. Since Eqs. (10.4) and (10.5) are second order in the spatial coordinate x, two boundary conditions are required. These may be of the Dirichlet type (i.e., specified values of f), the Neumann type (i.e., specified values of f_x), or the mixed type (i.e., specified combinations of f and f_x). The basic properties of finite difference methods for solving propagation problems governed by parabolic PDEs are presented in this chapter.

The organization of Chapter 10 is presented in Figure 10.3. Following the Introduction, the general features of parabolic partial differential equations are discussed. This discussion is followed by a discussion of the finite difference method. The solution of the diffusion equation by the forward-time centered-space (FTCS) method is then presented. This presentation is followed by a discussion of the concepts of consistency, order, stability, and convergence. Two additional explicit methods, the Richardson (leapfrog) method and the DuFort-Frankel method are then presented to illustrate an unstable method and an inconsistent method. Two implicit methods are then presented: the backward-time centered-space (BTCS) method and the Crank-Nicholson method. A procedure for implementing derivative boundary conditions is presented next. A discussion of nonlinear equations and multidimensional problems follows. A brief introduction to the solution of the convection-diffusion equation is then presented. This is followed by a discussion of the asymptotic steady-state solution of propagation problems as a procedure for solving mixed elliptic-parabolic and mixed elliptic-hyperbolic problems. A brief presentation of a program for solving the diffusion equation follows. A summary wraps up the chapter.

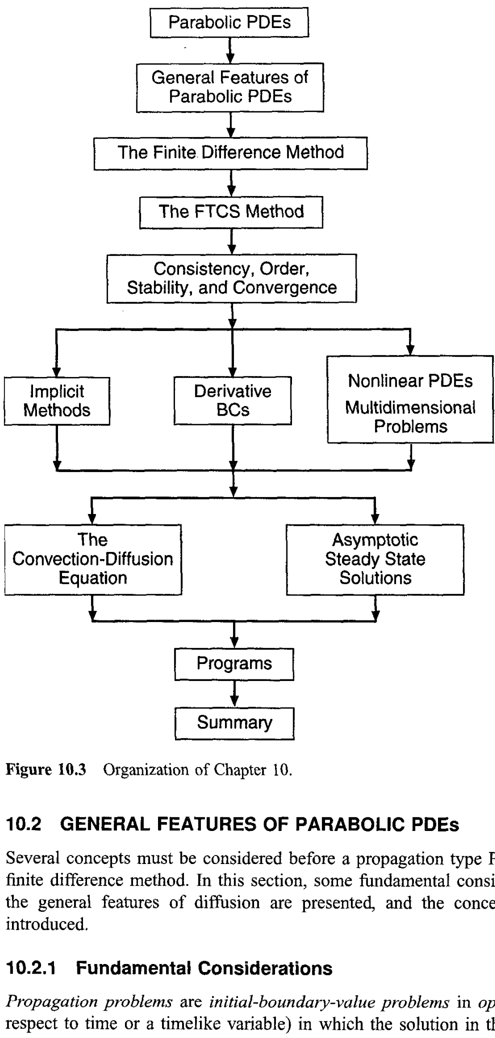


Figure 10.3 Organization of Chapter 10.

10.2 GENERAL FEATURES OF PARABOLIC PDEs

Several concepts must be considered before a propagation type PDE can be solved by a finite difference method. In this section, some fundamental considerations are discussed, the general features of diffusion are presented, and the concept of characteristics is introduced.

10.2.1 Fundamental Considerations

Propagation problems are *initial-boundary-value problems* in *open domains* (open with respect to time or a timelike variable) in which the solution in the domain of interest is marched forward from the initial state, guided and modified by the boundary conditions. Propagation problems are governed by parabolic or hyperbolic partial differential equations. The general features of parabolic and hyperbolic PDEs are discussed in Part III. Those features which are relevant to the finite difference solution of both parabolic and hyperbolic PDEs are presented in this section. Those features which are relevant only to the finite difference solution of hyperbolic PDEs are presented in Section 11.2.

The general features of *parabolic partial differential equations* (*PDEs*) are discussed in Section III.6. In that section it is shown that parabolic PDEs govern propagation problems, which are initial-boundary-value problems in open domains. Consequently, parabolic PDEs are solved numerically by marching methods. From the characteristic analysis presented in Section III.6, it is known that problems governed by parabolic PDEs have an *infinite physical information propagation speed.* As a result, the solution at a given point P at time level n depends on the solution at all other points in the solution domain at all times preceding and including time level n, and the solution at a given point P at time level n influences the solution at all other points in the solution domain at all times including and after time level n. Consequently, the physical information propagation speed $c = dx/dt$ is infinite. These general features of parabolic PDEs are illustrated in Figure 10.4.

10.2.2 General Features of Diffusion

Consider pure diffusion, which is governed by the diffusion equation:

$$\boxed{f_t = \alpha f_{xx}} \tag{10.6}$$

where α is the diffusion coefficient. Consider an initial property distribution, $f(x, 0) = \phi(x)$, given the general term of an exponential Fourier series:

$$\phi(x) = A_m e^{Ik_m x} \tag{10.7}$$

where $I = \sqrt{-1}$, $k_m = 2\pi m/2L$ is the wave number, and L is the width of the physical space. Assume that the exact solution of Eq. (10.6) is given by

$$f(x, t) = e^{-\alpha k_m^2 t}\phi(x) \tag{10.8}$$

Substituting Eq. (10.7) into Eq. (10.8) yields

$$f(x, t) = e^{-\alpha k_m^2 t} A_m e^{Ik_m x} \tag{10.9}$$

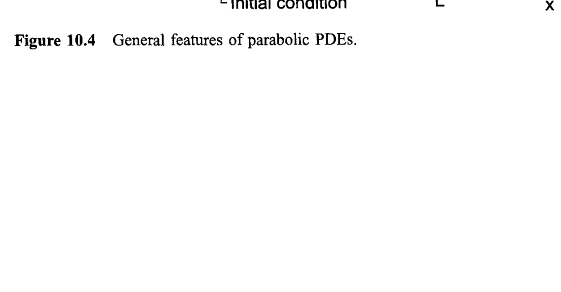


Figure 10.4 General features of parabolic PDEs.

Differentiating Eq. (10.9) with respect to t and x gives

$$f_t = -\alpha k_m^2 e^{-\alpha k_m^2 t} A_m e^{Ik_m x} = -\alpha k_m^2 e^{-\alpha k_m^2 t} \phi(x) \tag{10.10}$$

$$f_{xx} = e^{-\alpha k_m^2 t} A_m (Ik_m)^2 e^{Ik_m x} = -k_m^2 e^{-\alpha k_m^2 t} \phi(x) \tag{10.11}$$

Substituting Eqs. (10.10) and (10.11) into Eq. (10.6) demonstrates that Eq. (10.8) is the exact solution of the diffusion equation:

$$\boxed{f(x, t) = e^{-\alpha k_m^2 t} \phi(x)} \tag{10.12}$$

Equation (10.12) shows that the initial property distribution $\phi(x)$ simply decays with time at the exponential rate $\exp(-\alpha k_m^2 t)$. Thus, the rate of decay depends on the square of the wave number k_m. The initial property distribution does not propagate in space.

For an arbitrary initial property distribution represented by a Fourier series, Eq. (10.12) shows that each Fourier component simply decays exponentially with time, but that each component decays at a rate which depends on the square of its individual wave number k_m. Thus, the total property distribution changes shape. Consequently, pure diffusion causes the initial property distribution to decay and change shape, but the property distribution does not propagate in space.

10.2.3 Characteristic Concepts

The concept of characteristics of partial differential equations is introduced in Section III.3. In two-dimensional space, which is the case considered here (i.e., space x and time t), characteristics are paths (curved, in general) in the solution domain $D(x, t)$ along which physical information propagates. If a partial differential equation possesses real characteristics, then physical information propagates along the characteristic paths. The presence of characteristics has a significant effect on the solution of a partial differential equation (by both analytical and numerical methods).

Consider the unsteady one-dimensional diffusion equation $f_t = \alpha f_{xx}$. It is shown in Section III.6 that the characteristic paths for the unsteady one-dimensional diffusion equation are the lines of constant time. Thus, physical information propagates at an infinite rate throughout the entire physical solution domain. Every point influences all the other points, and every point depends on the solution at all the other points, including the boundary points. This behavior should be considered when solving parabolic PDEs by numerical methods.

10.3 THE FINITE DIFFERENCE METHOD

The objective of a finite difference method for solving a partial differential equation (PDE) is to transform a calculus problem into an algebra problem by

1. *Discretizing* the continuous physical domain into a discrete difference grid
2. *Approximating* the individual exact partial derivatives in the partial differential equation (PDE) by algebraic finite difference approximations (FDAs)
3. *Substituting* the FDAs into the PDE to obtain an algebraic finite difference equation (FDE)
4. *Solving* the resulting algebraic FDEs

There are several choices to be made when developing a finite difference solution to a partial differential equation. Foremost among these are the choice of the discrete finite difference grid used to discretize the continuous physical domain and the choice of the finite difference approximations used to represent the individual exact partial derivatives in the partial differential equation. Some fundamental considerations relevant to the finite difference approach are discussed in the next subsection. The general features of finite difference grids and finite difference approximations, which apply to both parabolic and hyperbolic PDEs, are discussed in the following subsections.

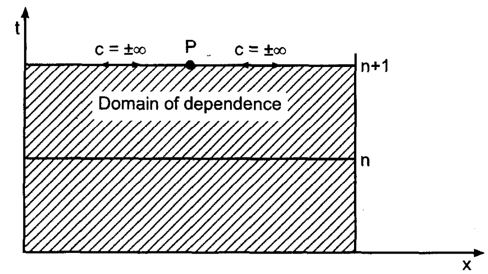


Figure 10.5 Physical domain of dependence of parabolic PDEs.

10.3.1 Fundamental Considerations

The objective of the numerical solution of a PDE is to march the solution at time level n forward in time to time level $n + 1$, as illustrated in Figure 10.5, where the physical domain of dependence of a parabolic PDE is illustrated. In view of the infinite physical information propagation speed $c = dx/dt$ associated with parabolic PDEs, the solution at point P at time level $n + 1$ depends on the solution at all of the other points at time level $n + 1$.

Finite difference methods in which the solution at point P at time level $n + 1$ depends only on the solution at neighboring points at time level n have a finite numerical information propagation speed $c_n = \Delta x/\Delta t$. Such finite difference methods are called *explicit methods* because the solution at each point is specified explicitly in terms of the known solution at neighboring points at time level n. This situation is illustrated in Figure 10.6, which resembles the physical domain of dependence of a hyperbolic PDE. The numerical information propagation speed $c_n = \Delta x/\Delta t$ is finite.

Finite difference methods in which the solution at point P at time level $n + 1$ depends on the solution at neighboring points at time level $n + 1$ as well as the solution at time level n have an infinite numerical information propagation speed $c_n = \Delta x/\Delta t$. Such methods couple the finite difference equations at time level $n + 1$ and result in a system of finite difference equations which must be solved at each time level. Such finite difference methods are called *implicit methods* because the solution at each point is specified implicitly in terms of the unknown solution at neighboring points at time level $n + 1$. This situation is illustrated in Figure 10.7, which resembles the physical domain of dependence of a parabolic PDE. The numerical information propagation speed, $c_n = \Delta x/\Delta t$, is infinite.

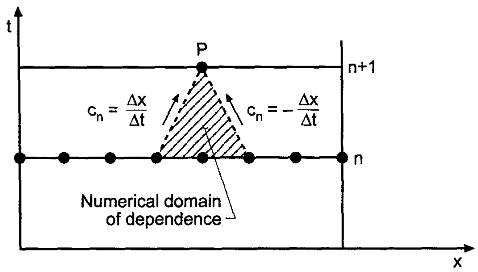


Figure 10.6 Numerical domain of dependence of explicit methods.

The similarities of and the differences between explicit and implicit numerical marching methods are illustrated in Figures 10.6 and 10.7. The major similarity is that both methods march the solution forward from one time level to the next time level. The major difference is that the numerical information propagation speed for explicit marching methods is finite, whereas the numerical information propagation speed for implicit marching methods is infinite.

Explicit methods are computationally faster than implicit methods because there is no system of finite difference equations to solve. Thus, explicit methods might appear to be superior to implicit methods. However, the finite numerical information propagation speed of explicit methods does not correctly model the infinite physical information propagation speed of parabolic PDEs, whereas the infinite numerical information propagation speed of implicit methods correctly models the infinite physical information propagation speed of parabolic PDEs. Thus, implicit methods appear to be well suited for solving parabolic PDEs, and explicit methods appear to be unsuitable for solving parabolic PDEs. In actuality, only an infinitesimal amount of physical information propagates at the infinite physical information propagation speed. The bulk of the physical information travels at a finite physical information propagation speed. Experience has shown that explicit methods as well as implicit methods can be employed to solve parabolic PDEs.

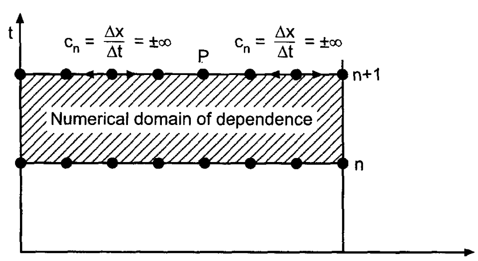


Figure 10.7 Numerical domain of dependence of implicit methods.

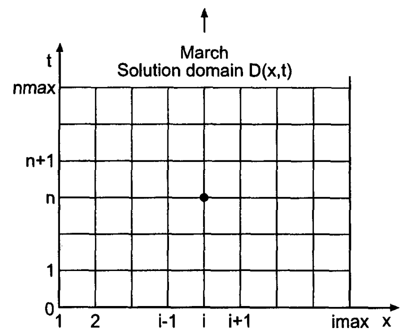


Figure 10.8 Solution domain, $D(x, t)$, and finite difference grid.

10.3.2 Finite Difference Grids

The solution domain $D(x, t)$ in xt space for an unsteady one-dimensional propagation problem is illustrated in Figure 10.8. The solution domain must be covered by a two-dimensional grid of lines, called the *finite difference grid*. The intersections of these *grid lines* are the *grid points* at which the finite difference solution of the partial differential equation is to be obtained. For the present, let the spatial grid lines be equally spaced lines perpendicular to the x axis having uniform spacing Δx. The temporal grid line spacing Δt may or may not be equally spaced. The resulting finite difference grid is also illustrated in Figure 10.8. The subscript i is used to denote the physical grid lines [i.e., $x_i = (i-1)\,\Delta x$], and the superscript n is used to denote the time grid lines (i.e., $t^n = n\,\Delta t$ if Δt is constant). Thus, grid point (i, n) corresponds to location (x_i, t^n) in the solution domain $D(x, t)$. The total number of x grid lines is denoted by imax, and the total number of time steps is denoted by nmax.

Two-dimensional physical spaces can be covered in a similar manner by a three-dimensional grid of planes perpendicular to the coordinate axes, where the subscripts i and j denote the physical grid planes perpendicular to the x and y axes, respectively, and the superscript n denotes time planes. Thus, grid point (i, j, n) corresponds to location (x_i, y_j, t^n) in the solution domain $D(x, y, t)$. Similarly, in three-dimensioinal physical space, grid point (i, j, k, n) corresponds to location (x_i, y_j, z_k, t^n) in the solution domain $D(x, y, z, t)$.

The dependent variable at a grid point is denoted by the same subscript-superscript notation that is used to denote the grid points themselves. Thus, the function $f(x, t)$ at grid point (i, n) is denoted by

$$f(x_i, t^n) = f_i^n \tag{10.13}$$

In a similar manner, derivatives are denoted by

$$\frac{\partial f(x_i, t^n)}{\partial t} = \left.\frac{\partial f}{\partial t}\right|_i^n = f_t|_i^n \quad \text{and} \quad \frac{\partial^2 f(x_i, t^n)}{\partial x^2} = \left.\frac{\partial^2 f}{\partial x^2}\right|_i^n = f_{xx}|_i^n \tag{10.14}$$

Similar results apply in two- and three-dimensional spaces.

10.3.3 Finite Difference Approximations

Now that the finite difference grid has been specified, *finite difference approximations* of the individual exact partial derivatives in the partial differential equation must be obtained. This is accomplished by writing Taylor series for the dependent variable at one or more grid points using a particular grid point as the base point and combining these Taylor series to solve for the desired partial derivatives. This is done in Chapter 5 for functions of one independent variable, where approximations of various types (i.e., forward, backward, and centered) of various orders (i.e., first order, second order, etc.) are developed for various derivatives (i.e., first derivative, second derivative, etc.). Those results are presented in Table 5.1.

In the development of finite difference approximations, a distinction must be made between the *exact solution* of a partial differential equation and the solution of the finite difference equation which is an *approximate solution* of the partial differential equation. For the remainder of this chapter, the exact solution of a PDE is denoted by an overbar over the symbol for the dependent variable, that is, $\bar{f}(x, t)$, and the approximate solution is denoted by the symbol for the dependent variable without an ovarbar, that is, $f(x, t)$. Thus,

$$\boxed{\begin{array}{l}\bar{f}(x,t) = \text{exact solution}\\ f(x,t) = \text{approximate solution}\end{array}}$$

Exact partial derivatives, such as $\bar{f}_t$ and $\bar{f}_{xx}$, which appear in the parabolic diffusion equation can be approximated at a grid point in terms of the values of $\bar{f}$ at that grid point and adjacent grid points in several ways. The exact time derivative $\bar{f}_t$ can be approximated at time level n by a first-order forward-time approximation or a second-order centered-time approximation. It can also be approximated at time level $n+1$ by a first-order backward-time approximation or at time level $n+1/2$ by a second-order centered-time approximation. The spatial derivative $\bar{f}_{xx}$ must be approximated at the same time level at which the time derivative $\bar{f}_t$ is evaluated.

The second-order spatial derivative $\bar{f}_{xx}$ is a model of physical diffusion. From characteristic concepts, it is known that the physical information propagation speed associated with second-order spatial derivatives is infinite, and that the solution at a point at a specified time level depends on and influences all of the other points in the solution domain at that time level. Consequently, second-order spatial derivatives, such as $\bar{f}_{xx}$, should be approximated by centered-space approximations at spatial location i. The centered-space approximations can be second-order, fourth-order, etc. Simplicity of the resulting finite difference equation usually dictates the use of second-order centered-space approximations for second-order spatial derivatives.

10.3.3.1 Time Derivatives

Consider the partial derivative $\bar{f}_t$. Writing the Taylor series for $\bar{f}_i^{n+1}$ using grid point (i, n) as the base point gives

$$\bar{f}_i^{n+1} = \bar{f}_i^n + \bar{f}_t|_i^n\,\Delta t + \tfrac{1}{2}\bar{f}_{tt}|_i^n\,\Delta t^2 + \cdots \tag{10.15}$$

where the convention $(\Delta t)^m \to \Delta t^m$ is employed for compactness. Solving Eq. (10.15) for $\bar{f}_t|_i^n$ yields

$$\bar{f}_t|_i^n = \frac{\bar{f}_i^{n+1} - \bar{f}_i^n}{\Delta t} - \tfrac{1}{2}\bar{f}_{tt}(\tau)\,\Delta t \tag{10.16}$$

where $t \le \tau \le t + \Delta t$. Truncating the remainder term yields the *first-order forward-time* approximation of $\bar{f}_t|_i^n$, denoted by $f_t|_i^n$:

$$\boxed{f_t|_i^n = \frac{f_i^{n+1} - f_i^n}{\Delta t}} \tag{10.17}$$

The remainder term which has been truncated in Eq. (10.17) is called the *truncation error* of the finite difference approximation of $\bar{f}_t|_i^n$. A *first-order backward-time* approximation and a *second-order centered time* approximation can be developed in a similar manner by choosing base points $n+1$ and $n+1/2$, respectively.

10.3.3.2 Space Derivatives

Consider the partial derivatives $\bar{f}_x$ and $\bar{f}_{xx}$. Writing Taylor series for $\bar{f}_{i+1}^n$ and $\bar{f}_{i-1}^n$ using grid point (i, n) as the base point gives

$$\bar{f}_{i+1}^n = \bar{f}_i^n + \bar{f}_x|_i^n\,\Delta x + \tfrac{1}{2}\bar{f}_{xx}|_i^n\,\Delta x^2 + \tfrac{1}{6}\bar{f}_{xxx}|_i^n\,\Delta x^3 + \tfrac{1}{24}\bar{f}_{xxxx}|_i^n\,\Delta x^4 + \cdots \tag{10.18}$$

$$\bar{f}_{i-1}^n = \bar{f}_i^n - \bar{f}_x|_i^n\,\Delta x + \tfrac{1}{2}\bar{f}_{xx}|_i^n\,\Delta x^2 - \tfrac{1}{6}\bar{f}_{xxx}|_i^n\,\Delta x^3 + \tfrac{1}{24}\bar{f}_{xxxx}|_i^n\,\Delta x^4 + \cdots \tag{10.19}$$

Subtracting Eq. (10.19) from Eq. (10.18) and solving for $\bar{f}_x|_i^n$ gives

$$\bar{f}_x|_i^n = \frac{\bar{f}_{i+1}^n - \bar{f}_{i-1}^n}{2\,\Delta x} - \tfrac{1}{3}\bar{f}_{xxx}(\xi)\,\Delta x^2 \tag{10.20}$$

where $x_{i-1} \le \xi \le x_{i+1}$. Truncating the remainder term yields the *second-order centered-space* approximation of $\bar{f}_x|_i^n$, denoted by $f_x|_i^n$:

$$\boxed{f_x|_i^n = \frac{f_{i+1}^n - f_{i-1}^n}{2\,\Delta t}} \tag{10.21}$$

Adding Eqs. (10.18) and (10.19) and solving for $\bar{f}_{xx}|_i^n$ gives

$$\bar{f}_{xx}|_i^n = \frac{\bar{f}_{i+1}^n - 2\bar{f}_i^n + \bar{f}_{i-1}^n}{\Delta x^2} - \tfrac{1}{12}\bar{f}_{xxxx}(\xi)\,\Delta x^2 \tag{10.22}$$

where $x_{i-1} \le \xi \le x_{i+1}$. Truncating the remainder term yields the *second-order centered-space* approximation of $\bar{f}_{xx}|_i^n$, denoted by $f_{xx}|_i^n$:

$$\boxed{f_{xx}|_i^n = \frac{f_{i+1}^n - 2f_i^n + f_{i-1}^n}{\Delta x^2}} \tag{10.23}$$

Second-order centered-difference finite difference approximations (FDAs) of $\bar{f}_x$ and $\bar{f}_{xx}$ at time level $n+1$ are obtained simply by replacing n by $n+1$ in Eqs. (10.21) and (10.23).

10.3.4 Finite Difference Equations

Finite difference equations are obtained by substituting the finite difference approximations of the individual exact partial derivatives into the PDE. Two types of FDEs can be developed, depending on the base point chosen for the FDAs. If grid point (i, n) is chosen as the base point of the FDAs, then f_i^{n+1} appears only in the finite difference approximation of $\bar{f}_t$. In that case, the FDE can be solved directly for f_i^{n+1}. Such FDEs are called *explicit* FDEs. However, if grid point $(i, n+1)$ is chosen as the base point of the FDAs, then f_i^{n+1} appears in the finite difference approximations of both $\bar{f}_t$ and $\bar{f}_{xx}$, and f_{i+1}^{n+1} and f_{i-1}^{n+1} appear in the finite difference approximation of $\bar{f}_{xx}$. In that case, f_i^{n+1} cannot be solved for directly, since f_i^{n+1} depends on f_{i+1}^{n+1} and f_{i-1}^{n+1}, which are also unknown. Such FDEs are called *implicit* FDEs.

Explicit FDEs are obviously easier to evaluate numerically than implicit FDEs. However, there are advantages and disadvantages of both explicit and implicit FDEs. Examples of both types of FDEs for solving the unsteady one-dimensional diffusion equation are developed in this chapter.

10.4 THE FORWARD-TIME CENTERED-SPACE (FTCS) METHOD

In this section the unsteady one-dimensional parabolic diffusion equation $\bar{f}_t = \alpha\bar{f}_{xx}$ is solved numerically by the *forward-time centered-space (FTCS) method*. In the FTCS method, the base point for the finite difference approximation (FDA) of the partial differential equation (PDE) is grid point (i, n). The finite difference equation (FDE) approximates the partial derivative $\bar{f}_t$ by the first-order forward-time approximation, Eq. (10.17), and the partial derivative $\bar{f}_{xx}$ by the second-order centered-space approximation, Eq. (10.23). The finite difference stencil is illustrated in Figure 10.9, where the grid points used to approximate $\bar{f}_t$ are denoted by the symbol $\times$ and the grid points used to approximate $\bar{f}_{xx}$ are denoted by the symbol $\cdot$. Thus,

$$\frac{f_i^{n+1} - f_i^n}{\Delta t} = \alpha \frac{f_{i+1}^n - 2f_i^n + f_{i-1}^n}{\Delta x^2} \tag{10.24}$$

Solving for f_i^{n+1} yields the desired FDE:

$$\boxed{f_i^{n+1} = f_i^n + d(f_{i+1}^n - 2f_i^n + f_{i-1}^n)} \tag{10.25}$$

where $d = \alpha\,\Delta t/\Delta x^2$ is called the *diffusion number*. Equation (10.25) is the FTCS approximation of the unsteady one-dimensional diffusion equation.

The general features of the FTCS approximation of the diffusion equation can be illustrated by applying it to solve the heat diffusion problem described in Section 10.1. Several solutions are presented in Example 10.1.

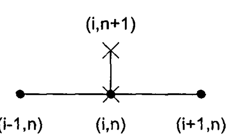


Figure 10.9 The FTCS method stencil.

Example 10.1. The FTCS method applied to the diffusion equation

Let's solve the heat diffusion problem presented in Section 10.1 by the FTCS method with $\Delta x = 0.1$ cm. Let $\Delta t = 0.1$ s, so $d = \alpha\,\Delta t/\Delta x^2 = (0.01)(0.1)/(0.1)^2 = 0.1$. The numerical solution $T(x, t)$ and errors, Error $= [T(x, t) - \bar{T}(x, t)]$, at selected times are presented in Table 10.2 and illustrated in Figure 10.10. Due to the symmetry of the solution, results are tabulated only for $x = 0.0$ to 0.5 cm. It is apparent that the numerical solution is a good approximation of the exact solution. The error at the midpoint (i.e., $x = 0.5$ cm) is the largest error at each time level. This is a direct result of the discontinuity in the slope of the initial temperature distribution at that point. However, the magnitude of this error decreases rapidly as the solution progresses, and the initial discontinuity in the slope is smoothed out. The errors at the remaining locations grow initially due to the accumulation of truncation errors, and reach a maximum value. As the solution progresses, however, the numerical solution approaches the exact asymptotic solution, $\bar{T}(x, \infty) = 0.0$, so the errors decrease and approach zero. The numerical results presented in Table 10.2 present a very favorable impression of the FTCS approximation of the diffusion equation.

The results obtained with $d = 0.1$ are quite good. However, a considerable amount of computational effort is required. The following question naturally arises: Can acceptable results be obtained with larger values of Δt, thus requiring less computational effort? To answer this question, let's rework the problem with $\Delta t = 0.5$ s (i.e., $d = 0.5$), which requires only one-fifth of the computational effort to reach a given time level. The results at selected times are illustrated in Figure 10.11. Although the solution is still

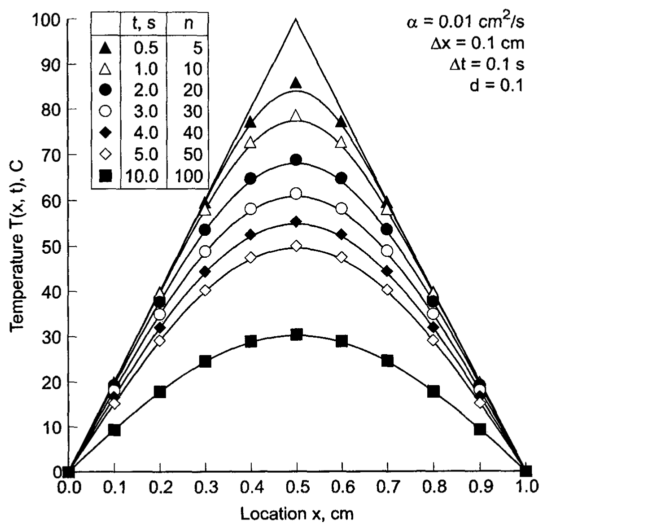


Figure 10.10 Solution by the FTCS method with $d = 0.1$.

Table 10.2 Solution by the FTCS Method for $d = 0.1$

	$T(x, t)$, C Error$(x, t) = [T(x, t) - \bar{T}(x, t)]$, C					
t, s	$x = 0.0$	$x = 0.1$	$x = 0.2$	$x = 0.3$	$x = 0.4$	$x = 0.5$
0.0	0.0	20.0000	40.0000	60.0000	80.0000	100.0000
1.0	0.0	19.9577	39.6777	58.2210	72.8113	78.6741
		−0.0033	0.0226	0.2312	0.7969	1.2417
2.0	0.0	19.3852	37.8084	53.7271	64.8650	68.9146
		0.0339	0.1483	0.3918	0.6887	0.8300
3.0	0.0	18.2119	35.0634	48.9901	58.2956	61.5811
		0.0884	0.2257	0.4152	0.5938	0.6683
4.0	0.0	16.7889	32.1161	44.5141	52.6255	55.4529
		0.1994	0.2576	0.4069	0.5289	0.5766
5.0	0.0	15.3371	29.2500	40.3905	47.6070	50.1072
		0.1312	0.2643	0.3890	0.4815	0.5160
10.0	0.0	9.4421	17.9610	24.7230	29.0653	30.5618
		0.1075	0.2049	0.2825	0.3326	0.3500

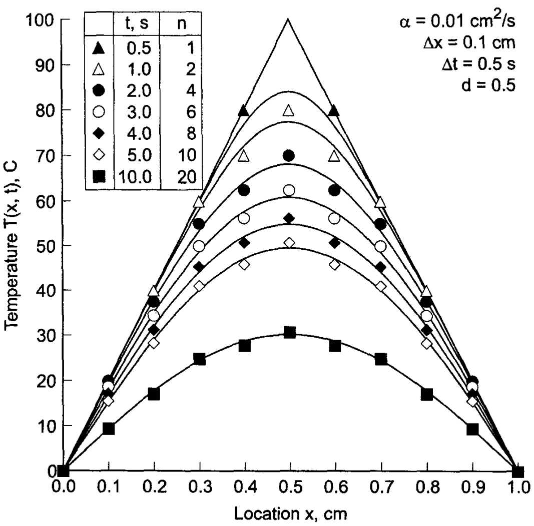


Figure 10.11 Solution by the FTCS method with $d = 0.5$.

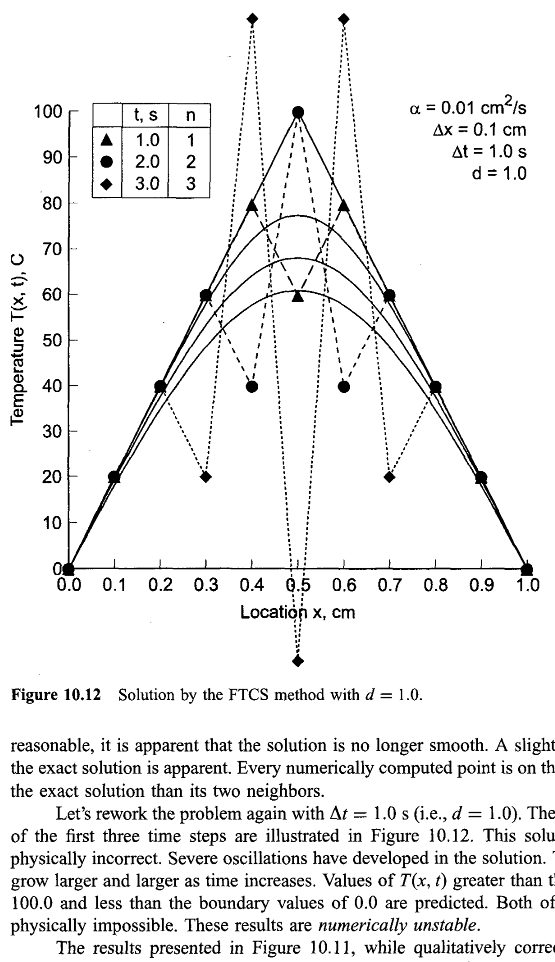


Figure 10.12 Solution by the FTCS method with $d = 1.0$.

reasonable, it is apparent that the solution is no longer smooth. A slight oscillation about the exact solution is apparent. Every numerically computed point is on the opposite side of the exact solution than its two neighbors.

Let's rework the problem again with $\Delta t = 1.0$ s (i.e., $d = 1.0$). The results after each of the first three time steps are illustrated in Figure 10.12. This solution is obviously physically incorrect. Severe oscillations have developed in the solution. These oscillations grow larger and larger as time increases. Values of $T(x, t)$ greater than the initial value of 100.0 and less than the boundary values of 0.0 are predicted. Both of these results are physically impossible. These results are *numerically unstable*.

The results presented in Figure 10.11, while qualitatively correct, appear on the verge of behaving like the results presented in Figure 10.12. The value $d = 0.5$ appears to be the boundary between physically meaningful results for d less than 0.5 and physically meaningless results for d greater than 0.5. To check out this supposition, let's rework the problem for two more values of d: $d = 04$ and $d = 0.6$. These results are illustrated in Figure 10.13 at $t = 6.0$ s. The numerical solution with $d = 0.4$ is obviously modeling physical reality, while the solution with $d = 0.6$ is not. These results support the

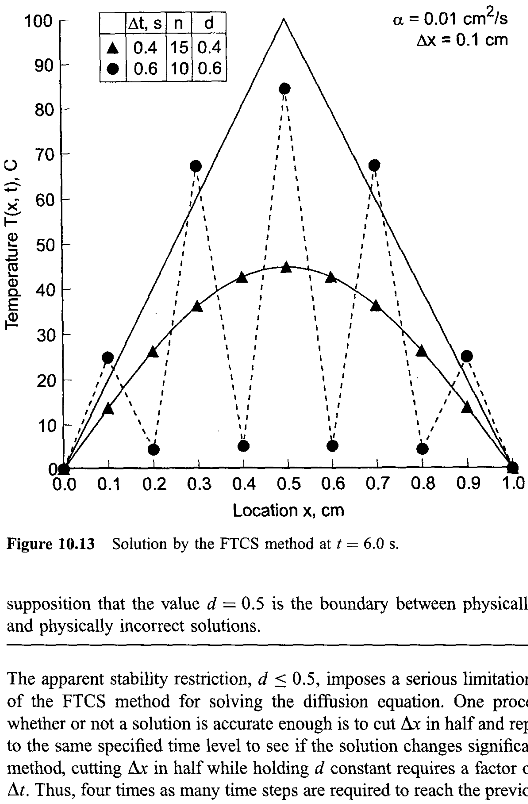


Figure 10.13 Solution by the FTCS method at $t = 6.0$ s.

supposition that the value $d = 0.5$ is the boundary between physically correct solutions and physically incorrect solutions.

The apparent stability restriction, $d \leq 0.5$, imposes a serious limitation on the usefulness of the FTCS method for solving the diffusion equation. One procedure for deciding whether or not a solution is accurate enough is to cut Δx in half and repeat the solution up to the same specified time level to see if the solution changes significantly. For the FTCS method, cutting Δx in half while holding d constant requires a factor of four decreases in Δt. Thus, four times as many time steps are required to reach the previously specified time level, and twice as much work is required for each time step since twice as many physical grid points are involved. Thus, the total computational effort increases by a factor of 8!

To further illustrate the FTCS approximation of the diffusion equation, consider a parametric study in which the temperature at $x = 0.4$ cm and $t = 5.0$ s, $T(0.4, 5.0)$, is calculated using values of $\Delta x = 0.1, 0.05, 0.025$ and 0.0125 cm for values of $d = 0.1$ and 0.5. The value of Δt for each solution is determined by the specified values of Δx and d. The exact solution is $\bar{T}(0.4, 5.0) = 47.1255$ C. The results are presented in Table 10.3. The truncation error of the FTCS method is $0(\Delta t) + 0(\Delta x^2)$. For a constant value of d, $\Delta t = d\,\Delta x^2/\alpha$. Thus, as Δx is successively halved, Δt is quartered. Consequently, both the $0(\Delta t)$ error term and the $0(\Delta x^2)$ error term, and thus the total error, should decrease by a factor of approximately 4 as Δx is halved for a constant value of d. This result is clearly evident from the results presented in Table 10.3.

Table 10.3 Parametric Study of $T(0.4, 5.0)$ by the FTCS Method

	$T(0.4, 5.0)$, C Error $(0.4, 5.0) = [T(0.4, 5.0) - \bar{T}(0.4, 5.0)]$, C			
Δx, cm	t, s	$d = 0.1$	t, s	$d = 0.5$
0.1	0.1	47.6070 0.4815	0.5	45.8984 −1.2271
0.05	0.025	47.2449 0.1194	0.125	47.4117 0.2862
0.025	0.00625	47.1553 0.0298	0.03125	47.1970 0.0715
0.0125	0.0015625	47.1329 0.0074	0.0078125	47.1434 0.0178

The forward-time centered-space (FTCS) method has a finite numerical information propagation speed $c_n = \Delta x/\Delta t$. Numerically, information propagates one physical grid increment in all directions during each time step. The diffusion equation has an infinite physical information propagation speed. Consequently, the FTCS method does not correctly model the physical information propagation speed of the diffusion equation. However, the bulk of the information propagates at a finite speed, and the FTCS method yields a reasonable approximation of the exact solution of the diffusion equation. For example, consider the results presented in this section. The solution at $t = 5.0$ s is presented in Table 10.4 for $d = 0.1$ and 0.5. The grid spacing, $\Delta x = 0.1$ cm, is the same for both solutions. The time step is determined from $\Delta t = d\,\Delta x^2/\alpha$. Thus, the numerical information propagation speed $c_n = \Delta x/\Delta t$ is given by

$$c_n = \frac{\Delta x}{\Delta t} = \frac{\Delta x}{d\,\Delta x^2/\alpha} = \frac{\alpha}{d\,\Delta x} = \frac{0.01}{d(0.1)} = \frac{0.1}{d}\ \text{cm/s} \tag{10.26}$$

Thus, $c_n = 1.0$ cm/s for $d = 0.1$ and $c_n = 0.2$ cm/s for $d = 0.5$. Consequently, the numerical information propagation speed varies by a factor of five for the results presented in Table 10.4. Those results show very little influence of this large change in the numerical information propagation speed, thus supporting the observation that the bulk of the physical information travels at a finite speed.

The explicit FTCS method can be applied to nonlinear PDEs simply by evaluating the nonlinear coefficients at base point (i, n). Systems of PDEs can be solved simply by

Table 10.4 Solution by the FTCS Method at $t = 5.0$ s

	$T(x, 5.0)$, C Error $= [T(x, 5.0) - \bar{T}(x, 5.0)]$, C					
d	$x = 0.0$	$x = 0.1$	$x = 0.2$	$x = 0.3$	$x = 0.4$	$x = 0.5$
0.1	0.0	15.3371 0.1312	29.2500 0.2643	40.3905 0.3890	47.6070 0.4815	50.1072 0.5160
0.5	0.0	15.6250 0.4191	28.3203 −0.6654	41.0156 1.0141	45.8984 −1.2271	50.7812 1.1900

solving the corresponding system of FDEs. Multidimensional problems can be solved simply by adding on the finite difference approximations of the y and z partial derivatives. Consequently, the FTCS method can be used to solve nonlinear PDEs, systems of PDEs, and multidimensional problems by a straightforward extension of the procedure presented in this section. The solution of nonlinear equations and multidimensional problems is discussed further in Section 10.9.

In summary, the forward-time centered-space (FTCS) approximation of the diffusion equation is explicit, single step, consistent, $0(\Delta t) + 0(\Delta x^2)$, conditionally stable, and convergent. It is somewhat inefficient because the time step varies as the square of the spatial grid size.

10.5 CONSISTENCY, ORDER, STABILITY, AND CONVERGENCE

There are four important properties of finite difference methods, for propagation problems governed by parabolic and hyperbolic PDEs, that must be considered before choosing a specific approach. They are:

1. Consistency
2. Order
3. Stability
4. Convergence

These concepts are defined and discussed in this section.

A finite difference equation is *consistent* with a partial differential equation if the difference between the FDE and the PDE (i.e., the truncation error) vanishes as the sizes of the grid spacings go to zero independently.

The *order* of a FDE is the rate at which the global error decreases as the grid sizes approach zero.

A finite difference equation is *stable* if it produces a bounded solution for a stable partial differential equation and is *unstable* if it produces an unbounded solution for a stable PDE.

A finite difference method is *convergent* if the solution of the finite difference equation (i.e., the numerical values) approaches the exact solution of the partial differential equation as the sizes of the grid spacings go to zero.

10.5.1 Consistency and Order

All finite difference equations must be analysed for consistency with the differential equation which they approximate. When the truncation errors of the finite difference approximations of the individual exact partial derivatives are known, proof of consistency is straightforward. When the truncation errors of the individual finite difference approximations are not known, the complete finite difference equation must be analyzed for consistency. That is accomplished by expressing each term in the finite difference equation [i.e., $f(x, t)$, not $\bar{f}(x, t)$] by a Taylor series with a particular base point. The resulting equation, which is called the *modified differential equation* (*MDE*), can be simplified to yield the exact form of the truncation error of the complete finite difference equation. Consistency can be investigated by letting the grid spacings go to zero. The order of the FDE is given by the lowest order terms in the MDE.

Warming and Hyett (1974) developed a convenient technique for analyzing the consistency of finite difference equations. The technique involves determining the actual partial differential equation that is solved by a finite difference equation. This actual partial differential equation is called the *modified differential equation* (*MDE*). Following Warming and Hyett, the MDE is determined by expressing each term in a finite difference equation in a Taylor series at some base point. Effectively, this changes the FDE back into a PDE.

Terms expressing in the MDE which do not appear in the original partial differential equation are truncation error terms. Analysis of the truncation error terms leads directly to the determination of *consistency* and *order*. A study of these terms can also yield insight into the stability of the finite difference equation. However, that approach to stability analysis is not presented in this book.

Example 10.2. Consistency and order analysis of the FTCS method.

As an example, consider the FTCS approximation of the diffusion equation $\bar{f}_t = \alpha\bar{f}_{xx}$ given by Eq. (10.25):

$$f_i^{n+1} = f_i^n + d(f_{i+1}^n - 2f_i^n + f_{i-1}^n) \tag{10.27}$$

where $d = \alpha\, \Delta t/\Delta x^2$ is the diffusion number. Let grid point (i, n) be the base point, and write Taylor series for all of the terms in Eq. (10.27). Thus,

$$f_i^{n+1} = f_i^n + f_t|_i^n\, \Delta t + \tfrac{1}{2} f_{tt}|_i^n\, \Delta t^2 + \tfrac{1}{6} f_{ttt}|_i^n\, \Delta t^3 + \cdots \tag{10.28}$$

$$\begin{aligned} f_{i\pm1}^n = f_i^n &\pm f_x|_i^n\, \Delta x + \tfrac{1}{2} f_{xx}|_i^n\, \Delta x^2 \pm \tfrac{1}{6} f_{xxx}|_i^n\, \Delta x^3 + \tfrac{1}{24} f_{xxxx}|_i^n\, \Delta x^4 \\ &\pm \tfrac{1}{120} f_{xxxxx}|_i^n\, \Delta x^5 + \tfrac{1}{720} f_{xxxxxx}|_i^n\, \Delta x^6 \pm \cdots \end{aligned} \tag{10.29}$$

Dropping the notation $|_i^n$ for clarity and substituting Eqs. (10.28) and (10.29) into Eq. (10.27) gives

$$\begin{aligned} &f + f_t\, \Delta t + \tfrac{1}{2} f_{tt}\, \Delta t^2 + \tfrac{1}{6} f_{ttt}\, \Delta t^3 + \cdots \\ &= f + \frac{\alpha\, \Delta t}{\Delta x^2}(2f + f_{xx}\, \Delta x^2 + \tfrac{1}{12} f_{xxxx}\, \Delta x^4 + \tfrac{1}{360} f_{xxxxxx}\, \Delta x^6 + \cdots - 2f) \end{aligned} \tag{10.30}$$

Cancelling zero-order terms (i.e., f), dividing through by Δt, and rearranging terms yields the MDE:

$$\boxed{\begin{aligned} f_t = \alpha f_{xx} &- \tfrac{1}{2} f_{tt}\, \Delta t - \tfrac{1}{6} f_{ttt}\, \Delta t^2 - \cdots \\ &+ \tfrac{1}{12}\alpha f_{xxxx}\, \Delta x^2 + \tfrac{1}{360}\alpha f_{xxxxxx}\, \Delta x^4 + \cdots \end{aligned}} \tag{10.31}$$

As $\Delta t \to 0$ and $\Delta x \to 0$, Eq. (10.31) approaches $f_t = \alpha f_{xx}$, which is the diffusion equation. Consequently, Eq. (10.27) is a consistent approximation of the diffusion equation. Equation (10.31) shows that the FDE is $0(\Delta t) + 0(\Delta x^2)$.

10.5.2 Stability

First, the general behavior of the exact solution of the PDE must be considered. If the partial differential equation itself is unstable, then the numerical solution also must be unstable. The concept of stability does not apply in that case. However, if the PDE itself is stable, then the numerical solution must be bounded. The concept of stability applies in that case.

Several methods have been devised to analyze the stability of a finite difference approximation of a PDE. Three methods for analysing the stability of FDEs are

1. The discrete perturbation method
2. The von Neumann method
3. The matrix method

The von Neumann method will be used to analyze stability for all of the finite difference equations developed in this book.

Stability analyses can be peformed only for linear PDEs. Consequently, nonlinear PDEs must be linearized locally, and the FDE which approximates the linearized PDE is analyzed for stability. Experience has shown that the stability criteria obtained for the FDE approximating the linearized PDE also apply to the FDE approximating the nonlinear PDE. Instances of suspected *nonlinear instabilities* have been reported in the literature, but it is not clear whether those phenomena are due to actual instabilities, inconsistent finite difference equations, excessively large grid spacings, inadequate treatment of boundary conditions, or simply incorrect computations. Consequently, in this book, the stability analysis of the finite difference equation which approximates a linearized PDE will be considered sufficient to determine the stability criteria for the FDE, even for nonlinear partial differential equations.

The von Neumann method of stability analysis will be used exclusively in this book. In the von Neumann method, the exact solution of the finite difference equation is obtained for the general Fourier component of a complex Fourier series representation of the initial property distribution. If the solution for the general Fourier component is bounded (either conditionally or unconditionally), then the finite difference equation is stable. If the solution for the general Fourier component is unbounded, then the finite difference equation is unstable.

Consider the FTCS approximation of the unsteady diffusion equation, Eq. (10.25):

$$f_i^{n+1} = f_i^n + d(f_{i+1}^n - 2f_i^n + f_{i-1}^n) \tag{10.32}$$

The exact solution of Eq. (10.32) for a single step can be expressed as

$$\boxed{f_i^{n+1} = Gf_i^n} \tag{10.33}$$

where G, which is called the *amplification factor*, is in general a complex constant. The solution of the FDE at time $T = N\,\Delta t$ is then

$$f_i^N = G^N f_i^0 \tag{10.34}$$

where $f_i^N = f(x_i, T)$ and $f_i^0 = f(x_i, 0)$. For f_i^N to remain bounded,

$$\boxed{|G| \le 1} \tag{10.35}$$

Stability analysis thus reduces to the determination of the single step exact solution of the finite difference equation, that is, the amplification factor G, and an investigation of the conditions necessary to ensure that $|G| \le 1$.

From Eq. (10.32), it is seen that f_i^{n+1} depends not only on f_i^n, but also on f_{i-1}^n and f_{i+1}^n. Consequently, f_{i-1}^n and f_{i+1}^n must be related to f_i^n, so that Eq. (10.32) can be solved explicitly for G. That is accomplished by expressing $f(x, t^n) = F(x)$ in a complex Fourier series. Each component of the Fourier series is propagated forward in time independently of all of the other Fourier components. The complete solution at any subsequent time is simply the sum of the individual Fourier components at that time.

The complex Fourier series for $f(x, t^n) = F(x)$ is given by

$$f(x, t^n) = F(x) = \sum_{m=-\infty}^{\infty} A_m e^{Ik_m x} = \sum_{m=-\infty}^{\infty} F_m \tag{10.36}$$

where the wave number k_m is defined as

$$k_m = 2m\pi/2L$$

Let $f_i^n = f(x_i, t^n)$ consist of the general term F_m. Thus,

$$f_i^n = F_m = A_m e^{Ik_m x_i} = A_m e^{Ik_m(i\,\Delta x)} = A_m e^{Ii(k_m\,\Delta x)} \tag{10.37}$$

Then $f_{i\pm1}^n = f(x_{i\pm1}, t^n)$ is given by

$$f_{i\pm1}^n = A_m e^{Ik_m(x_i \pm \Delta x)} = A_m e^{Ik_m(i\pm1)(\Delta x)} = A_m e^{Ii(k_m\,\Delta x)} e^{\pm I(k_m\,\Delta x)} = f_i^n e^{\pm I(k_m\,\Delta x)} \tag{10.38}$$

Equation (10.38) relates $f_{i\pm1}^n$ to f_i^n. A similar analysis of $f(x_i, t^{n+1})$ gives

$$f_{i\pm1}^{n+1} = f_i^{n+1} e^{\pm I(k_m\,\Delta x)} \tag{10.39}$$

Substituting these results into a FDE expresses the FDE in terms of f_i^n and f_i^{n+1} only, which enables the exact solution, Eq. (10.33), to be determined.

Equations (10.38) and (10.39) apply to the mth component of the complex Fourier series, Eq. (10.36). To ensure stability for a general property distribution, all components of Eq. (10.36) must be considered, and all values of Δx must be considered. This is accomplished by letting m vary from $-\infty$ to $+\infty$ and letting Δx vary from 0 to L. Thus, the product $k_m\,\Delta x$ varies from $-\infty$ to $+\infty$.

The complex exponentials in Eqs. (10.38) and (10.39), that is, $\exp[\pm I(k_m\,\Delta x)]$, represent sine and cosine functions, which have a period of 2π. Consequently, the values of these exponentials repeat themselves with a period of 2π. Thus, it is only necessary to investigate the behavior of the amplification factor G over the range $0 \le (k_m \Delta x) \le 2\pi$. In view of this behavior, the term $(k_m\,\Delta x)$ will be denoted simply as θ, and Eqs. (10.38) and (10.39) can be written as

$$\boxed{f_{i\pm1}^n = f_i^n e^{\pm I\theta} \quad \text{and} \quad f_{i\pm1}^{n+1} = f_i^{n+1} e^{\pm I\theta}} \tag{10.40}$$

Equation (10.40) can be expressed in terms of $\sin\theta$ and $\cos\theta$ using the relationships

$$\cos\theta = \frac{e^{I\theta} + e^{-I\theta}}{2} \quad \text{and} \quad \sin\theta = \frac{e^{I\theta} - e^{-I\theta}}{2I} \tag{10.41}$$

The steps for performing a von Neumann stability analysis of a finite difference equation (FDE) are summarized below.

1. Substitute the complex Fourier components for $f_{i\pm1}^n$ and $f_{i\pm1}^{n+1}$ into the FDE.
2. Express $\exp(\pm I\theta)$ in terms of $\sin\theta$ and $\cos\theta$ and determine the amplification factor, G.
3. Analyze G (i.e., $|G| \le 1$) to determine the stability criteria for the FDE.

Example 10.3. Stability analysis of the FTCS method

As an example of the von Neumann method of stability analysis, let's perform a stability analysis of the FTCS approximation of the diffusion equation, Eq. (10.25):

$$f_i^{n+1} = f_i^n + d(f_{i+1}^n - 2f_i^n + f_{i-1}^n) \tag{10.42}$$

The required Fourier components are given by Eq. (10.40). Substituting Eq. (10.40) into Eq. (10.42) gives

$$f_i^{n+1} = f_i^n + d(f_i^n e^{I\theta} - 2f_i^n + f_i^n e^{-I\theta}) \tag{10.43a}$$

which can be written as

$$f_i^{n+1} = f_i^n[1 + d(e^{I\theta} + e^{-I\theta} - 2)] = f_i^n\left[1 + 2d\left(\frac{e^{I\theta} + e^{-I\theta}}{2} - 1\right)\right] \tag{10.43b}$$

Introducing the relationship between the cosine and exponential functions, Eq. (10.41), yields

$$f_i^{n+1} = f_i^n[1 + 2d(\cos\theta - 1)] \tag{10.44}$$

Thus, the amplification factor G is defined as

$$\boxed{G = 1 + 2d(\cos\theta - 1)} \tag{10.45}$$

The amplification factor G is the single step exact solution of the finite difference equation for the general Fourier component, which must be less than unity in magnitude to ensure a bounded solution. For a specific wave number k_m and grid spacing Δx, Eq. (10.45) can be analysed to determine the range of values of the diffusion number d for which $|G| \le 1$. In the infinite Fourier series representation of the property distribution, k_m ranges from $-\infty$ to $+\infty$. The grid spacing Δx can range from zero to any finite value up to L, where L is the length of the physical space. Consequently, the product $(k_m \Delta x) = \theta$ ranges continuously from $-\infty$ to $+\infty$. To ensure that the FDE is stable for an arbitrary property distribution and arbitrary Δx, Eq. (10.45) must be analysed to determine the range of values of d for which $|G| \le 1$ as θ ranges continuously from $-\infty$ to $+\infty$.

Solving Eq. (10.45) for $|G| \le 1$ yields

$$-1 \le 1 + 2d(\cos\theta - 1) \le 1 \tag{10.46}$$

Note that $d = \alpha\, \Delta t/\Delta x^2$ is always positive. The upper limit is always satisfied for $d \ge 0$ because $(\cos\theta - 1)$ varies between -2 and 0 as θ ranges from $-\infty$ to $+\infty$. From the lower limit,

$$d \le \frac{1}{1 - \cos\theta} \tag{10.47}$$

The minimum value of d corresponds to the maximum value of $(1 - \cos\theta)$. As θ ranges from $-\infty$ to $+\infty$, $(1 - \cos\theta)$ varies between 0 and 2. Consequently, the minimum value of d is $\frac{1}{2}$. Thus, $|G| \le 1$ for all values of $\theta = k_m\, \Delta x$ if

$$\boxed{d \le \tfrac{1}{2}} \tag{10.48}$$

Consequently, the FTCS approximation of the diffusion equation is conditionally stable. This result explains the behavior of the FTCS method for $d = 0.6$ and $d = 1.0$ illustrated in Example 10.1.

The behavior of the amplification factor G also can be determined by graphical methods. Equation (10.45) can be written in the form

$$G = (1 - 2d) + 2d\cos\theta \tag{10.49}$$

In the complex plane, Eq. (10.49) represents an oscillation on the real axis, centered at $(1 - 2d + I0)$, with an amplitude of $2d$, as illustrated in Figure 10.14. The stability boundary, $|G| = 1$, is a circle of radius unity in the complex plane. For G to remain on or inside the unit circle, $-1 \le |G| \le 1$, as θ varies from $-\infty$ to $+\infty$, $2d \le 1$. The graphical approach is very useful when G is a complex function.

10.5.3 Convergence

The proof of convergence of a finite difference method is in the domain of the mathematician. We shall not attempt to prove convergence directly. However, the convergence of a finite difference method is related to the consistency and stability of the finite difference equation. The *Lax equivalence theorem* [Lax (1954) states:

> Given a properly posed linear initial-value problem and a finite difference approximation to it that is consistent, stability is the necessary and sufficient condition for convergence.

Thus, the question of convergence of a finite difference method is answered by a study of the consistency and stability of the finite difference equation. If the finite difference equation is consistent and stable, then the finite difference method is convergent.

The Lax equivalence theorem applies to well-posed, linear, initial-value problems. Many problems in engineering and science are not linear, and nearly all problems involve boundary conditions in addition to the initial conditions. There is no equivalence theorem for such problems. Nonlinear PDEs must be linearized locally, and the FDE that approximates the linearized PDE is analysed for stability. Experience has shown that the stability criteria obtained for the FDE which approximates the linearized PDE also apply to

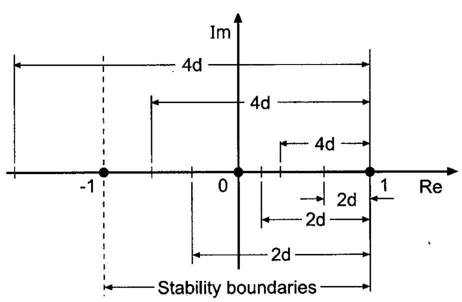


Figure 10.14 Locus of the amplification factor G for the FTCS method.

the FDE which approximates the nonlinear PDE, and that FDEs that are consistent and whose linearized equivalent is stable generally converge, even for nonlinear initial-boundary-value problems.

10.5.3 Summary

The concepts of consistency, stability, and convergence must be considered when choosing a finite difference approximation of a partial differential equation. Consistency is demonstrated by developing the modified differential equation (MDE) and letting the grid increments go to zero. The MDE also yields the order of the FDE. Stability is ascertained by developing the amplification factor, G, and determining the conditions required to ensure that $|G| \leq 1$.

Convergence is assured by the Lax equivalence theorem if the finite difference equation is consistent and stable.

10.6 THE RICHARDSON AND DUFORT-FRANKEL METHODS

The forward-time centered-space (FTCS) approximation of the diffusion equation $\bar{f}_t = \alpha \bar{f}_{xx}$ presented in Section 10.4 has several desirable features. It is an explicit, two-level, single-step method. The finite difference approximation of the spatial derivative is second order. However, the finite difference approximation of the time derivative is only first order. An obvious improvement would be to use a second-order finite difference approximation of the time derivative. The Richardson (leapfrog) and DuFort-Frankel methods are two such methods.

10.6.1 The Richardson (Leapfrog) Method

Richardson (1910) proposed approximating the diffusion equation $\bar{f}_t = \alpha \bar{f}_{xx}$ by replacing the partial derivative $\bar{f}_t$ by the three-level second-order centered-difference approximation based on time levels $n-1$, n, and $n+1$, and replacing the partial derivative $\bar{f}_{xx}$ by the second-order centered-difference approximation, Eq. (10.23). The corresponding finite difference stencil is presented in Figure 10.15. The Taylor series for f_i^{n+1} and f_i^{n-1} with base point (i, n) are given by

$$\bar{f}_i^{n+1} = \bar{f}_i^n + \bar{f}_t|_i^n\, \Delta t + \tfrac{1}{2}\bar{f}_{tt}|_i^n\, \Delta t^2 + \tfrac{1}{6}\bar{f}_{ttt}|_i^n\, \Delta t^3 + \tfrac{1}{24}\bar{f}_{tttt}|_i^n\, \Delta t^4 + \cdots \tag{10.50}$$

$$\bar{f}_i^{n-1} = \bar{f}_i^n - \bar{f}_t|_i^n\, \Delta t + \tfrac{1}{2}\bar{f}_{tt}|_i^n\, \Delta t^2 - \tfrac{1}{6}\bar{f}_{ttt}|_i^n\, \Delta t^3 + \tfrac{1}{24}\bar{f}_{tttt}|_i^n\, \Delta t^4 + \cdots \tag{10.51}$$

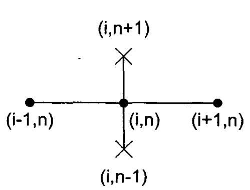


Figure 10.15 The Richardson (leapfrog) method stencil.

Adding Eqs. (10.50) and (10.51) gives

$$\bar{f}_i^{n+1} + \bar{f}_i^{n-1} = 2\bar{f}_i^n + \bar{f}_{tt}|_i^n\, \Delta t^2 + \tfrac{1}{12}\bar{f}_{tttt}|_i^n\, \Delta t^4 + \cdots \tag{10.52}$$

Solving Eq. (10.52) for $\bar{f}_t|_i^n$ gives

$$\bar{f}_t|_i^n = \frac{\bar{f}_i^{n+1} - \bar{f}_i^{n-1}}{2\,\Delta t} - \tfrac{1}{12}\bar{f}_{ttt}(\tau)\, \Delta t^2 \tag{10.53}$$

where $t^{n-1} \leq \tau \leq t^{n+1}$. Truncating the remainder term yields the second-order centered-time approximation:

$$\boxed{f_t|_i^n = \frac{f_i^{n+1} - f_i^{n-1}}{2\,\Delta t}} \tag{10.54}$$

Substituting Eqs. (10.54) and (10.23) into the diffusion equation gives

$$\frac{f_i^{n+1} - f_i^{n-1}}{2\,\Delta t} = \alpha \frac{f_{i+1}^n - 2f_i^n + f_{i-1}^n}{\Delta x^2} \tag{10.55}$$

Solving Eq. (10.55) for f_i^{n+1} yields

$$\boxed{f_i^{n+1} = f_i^{n-1} + 2d(f_{i+1} - 2f_i^n + f_{i-1}^n)} \tag{10.56}$$

where $d = \alpha\, \Delta t/\Delta x^2$ is the diffusion number.

The Richardson method appears to be a significant improvement over the FTCS method because of the increased accuracy of the finite difference approximation of $\bar{f}_t$. However, Eq. (10.56) is unconditionally unstable. Performing a von Neumann stability analysis of Eq. (10.56) (where $f_i^n = Gf_i^{n-1}$) yields

$$G = \frac{1}{G} + 4d(\cos\theta - 1) \tag{10.57}$$

which yields

$$G^2 + bG - 1 = 0 \tag{10.58}$$

where $b = -4d(\cos\theta - 1) = 8d\sin^2(\theta/2)$. Solving Eq. (10.58) by the quadratic formula yields

$$G = \frac{-b \pm \sqrt{b^2 + 4}}{2} \tag{10.59}$$

When $b = 0$, $|G| = 1$. For all other values of b, $|G| > 1$. Consequently, the Richardson (leapfrog) method is unconditionally unstable when applied to the diffusion equation.

Since the Richardson method is unconditionally unstable when applied to the diffusion equation, it cannot be used to solve that equation, or any other parabolic PDE. This conclusion applies only to parabolic differential equations. The combination of a three-level centered-time approximation of $\bar{f}_t$ combined with a centered-space approximation of a spatial derivative may be stable when applied to hyperbolic partial differential equations. For example, when applied to the hyperbolic convection equation, where it is known simply as the leapfrog method, a conditionally stable finite difference method is obtained. However, when applied to the convection-diffusion equation, an unconditionally unstable finite difference equation is again obtained. Such occurrences of diametrically opposing results require the numerical analyst to be constantly alert when applying finite difference approximations to solve partial differential equations.

10.6.2 The DuFort-Frankel Method

DuFort and Frankel (1953) proposed a modification to the Richardson method for the diffusion equation $\bar{f}_t = \alpha \bar{f}_{xx}$ which removes the unconditional instability. In fact, the resulting FDE is unconditionally stable. The central grid point value f_i^n in the second-order centered-difference approximation of $\bar{f}_{xx}|_i^n$ is replaced by the average of f_i at time levels $n+1$ and $n-1$, that is, $f_i^n = (f_i^{n+1} + f_i^{n-1})/2$. Thus, Eq. (10.55) becomes

$$\frac{f_i^{n+1} - f_i^{n-1}}{2\,\Delta t} = \alpha \frac{f_{i+1}^n - (f_i^{n+1} + f_i^{n-1}) + f_{i-1}^n}{\Delta x^2} \tag{10.60}$$

At this point, it is not obvious how the truncation error is affected by this replacement. The value f_i^{n+1} appears on both sides of Eq. (10.60). However, it appears linearly, so Eq. (10.60) can be solved explicitly for f_i^{n+1}. Thus,

$$\boxed{(1+2d)f_i^{n+1} = (1-2d)f_i^{n-1} + 2d(f_{i+1}^n + f_{i-1}^n)} \tag{10.61}$$

where $d = \alpha\,\Delta t/\Delta x^2$ is the diffusion number.

The modified differential equation (MDE) corresponding to Eq. (10.61) is

$$\begin{aligned} f_t = \alpha f_{xx} - \tfrac{1}{6} f_{ttt}\,\Delta t^2 - \cdots - \alpha f_{tt}\frac{\Delta t^2}{\Delta x^2} - \tfrac{1}{12}\alpha f_{tttt}\frac{\Delta t^4}{\Delta x^2} + \cdots \\ + \tfrac{1}{12}\alpha f_{xxxx}\,\Delta x^2 + \tfrac{1}{360}\alpha f_{xxxxxx}\,\Delta x^4 + \cdots \end{aligned} \tag{10.62}$$

As $\Delta t \to 0$ and $\Delta x \to 0$, the terms involving the ratio $(\Delta t/\Delta x)$ do not go to zero. In fact, they become indeterminate. Consequently, Eq. (10.62) is not a consistent approximation of the diffusion equation. A von Neumann stability analysis does show that $|G| \le 1$ for all values of d. Thus, Eq. (10.61) is unconditionally stable. However, due to the inconsistency illustrated in Eq. (10.62), the DuFort-Frankel method is not an acceptable method for solving the parabolic diffusion equation, or any other parabolic PDE. Consequently, it will not be considered further.

10.7 IMPLICIT METHODS

The forward-time centered-space (FTCS) method is an example of an explicit finite difference method. In explicit methods, the finite difference approximations of the individual exact partial derivatives in the partial differential equation are evaluated at the known time level n. Consequently, the solution at a point at the solution time level $n+1$ can be expressed explicitly in terms of the known solution at time level n. Explicit finite difference methods have many desirable features. However, they share one undesirable feature: they are only conditionally stable, or as in the case of the DuFort-Frankel method, they are not consistent with the partial differential equation. Consequently, the allowable time step is generally quite small, and the amount of computational effort required to obtain the solution of some problems is quite large. A procedure for avoiding the time step limitation is obviously desirable. Implicit finite difference methods provide such a procedure.

In implicit methods, the finite difference approximations of the individual exact partial derivatives in the partial differential equation are evaluated at the solution time level $n+1$. Fortuitously, implicit difference methods are unconditionally stable. There is no limit on the allowable time step required to achieve a numerically stable solution. There is, of course, some practical limit on the time step required to maintain the truncation errors within reasonable limits, but this is not a stability consideration; it is an accuracy consideration. Implicit methods do have some disadvantages, however. The foremost disadvantage is that the solution at a point in the solution time level $n+1$ depends on the solution at neighboring points in the solution time level, which are also unknown. Consequently, the solution is implied in terms of unknown function values, and a system of finite difference equations must be solved to obtain the solution at each time level. Additional complexities arise when the partial differential equations are nonlinear. In that case, a system of nonlinear finite difference equations results, which must be solved by some manner of linearization and/or iteration.

In spite of their disadvantages, the advantage of unconditional stability makes implicit finite difference methods attractive. Consequently, two implicit finite difference methods are presented in this section: the backward-time centered-space (BTCS) method and the Crank-Nicolson (1947) method.

10.7.1 The Backward-Time Centered-Space (BTCS) Method

In this subsection the unsteady one-dimensional diffusion equation, $\bar{f}_t = \alpha \bar{f}_{xx}$, is solved by the *backward-time centered-space (BTCS) method.* This method is also called the *fully implicit method.* The finite difference equation which approximates the partial differential equation is obtained by replacing the exact partial derivative $\bar{f}_t$ by the first-order backward-time approximation, which is developed below, and the exact partial derivative $\bar{f}_{xx}$ by the second-order centered-space approximation, Eq. (10.23), evaluated at time level $n+1$. The finite difference stencil is illustrated in Figure 10.16. The Taylor series for $\bar{f}_i^n$ with base point $(i, n+1)$ is given by

$$\bar{f}_i^n = \bar{f}_i^{n+1} + \bar{f}_t|_i^{n+1}(-\Delta t) + \tfrac{1}{2}\bar{f}_{tt}|_i^{n+1}(-\Delta t)^2 + \cdots \tag{10.63}$$

Solving Eq. (10.63) for $\bar{f}_t|_i^{n+1}$ gives

$$\bar{f}_t|_i^{n+1} = \frac{\bar{f}_i^{n+1} - \bar{f}_i^n}{\Delta t} + \tfrac{1}{2}\bar{f}_{tt}(\tau)\,\Delta t \tag{10.64}$$

Truncating the remainder term yields the first-order backward-time approximation:

$$\boxed{f_t|_i^{n+1} = \frac{f_i^{n+1} - f_i^n}{\Delta t}} \tag{10.65}$$

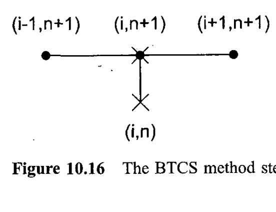


Figure 10.16 The BTCS method stencil.

Substituting Eqs. (10.65) and (10.23) into the diffusion equation, $\bar{f}_t = \alpha \bar{f}_{xx}$, yields

$$\frac{f_i^{n+1} - f_i^n}{\Delta t} = \alpha \frac{f_{i+1}^{n+1} - 2f_i^{n+1} + f_{i-1}^{n+1}}{\Delta x^2} \tag{10.66}$$

Rearranging Eq. (10.66) yields the implicit BTCS FDE:

$$\boxed{-df_{i-1}^{n+1} + (1+2)f_i^{n+1} - df_{i+1}^{n+1} = f_i^n} \tag{10.67}$$

where $d = \alpha\, \Delta t / \Delta x^2$ is the diffusion number.

Equation (10.67) cannot be solved explicitly for f_i^{n+1} because the two unknown neighboring values f_{i-1}^{n+1} and f_{i+1}^{n+1} also appear in the equation. The value of f_i^{n+1} is implied in Eq. (10.67), however. Finite difference equations in which the unknown solution value f_i^{n+1} is implied in terms of its unknown neighbors rather than being explicitly given in terms of known initial values are called implicit FDEs.

The modified differential equation (MDE) corresponding to Eq. (10.67) is

$$f_t = \alpha f_{xx} + \tfrac{1}{2} f_{tt}\, \Delta t - \tfrac{1}{6} f_{ttt}\, \Delta t^2 + \cdots + \tfrac{1}{12} \alpha f_{xxxx}\, \Delta x^2 + \tfrac{1}{360} \alpha f_{xxxxxx}\, \Delta x^4 + \cdots \tag{10.68}$$

As $\Delta t \to 0$ and $\Delta x \to 0$, all of the truncation error terms go to zero, and Eq. (10.68) approaches $f_t = \alpha f_{xx}$. Consequently, Eq. (10.67) is consistent with the diffusion equation. The truncation error is $0(\Delta t) + 0(\Delta x^2)$. From a von Neumann stability analysis, the amplification factor G is

$$G = \frac{1}{1 + 2d(1 - \cos\theta)} \tag{10.69}$$

The term $(1 - \cos\theta)$ is greater than or equal to zero for all values of $\theta = (k_m\, \Delta x)$. Consequently, the denominator of Eq. (10.69) is always ≥ 1. Thus, $|G| \leq 1$ for all positive values of d, and Eq. (10.67) is unconditionally stable. The BTCS approximation of the diffusion equation is consistent and unconditionally stable. Consequently, by the Lax Equivalence Theorem, the BTCS method is convergent.

Consider now the solution of the unsteady one-dimensional diffusion equation by the BTCS method. The finite difference grid for advancing the solution from time level n to time level $n + 1$ is illustrated in Figure 10.17. For Dirichlet boundary conditions (i.e., the value of the function is specified at the boundaries), the finite difference equation must be applied only at the interior points, points 2 to imax − 1. At grid point 1, $f_1^{n+1} = \bar{f}(0, t)$, and at grid point imax, $f_{\text{imax}}^{n+1} = \bar{f}(L, t)$. The following set of simultaneous linear equations is obtained:

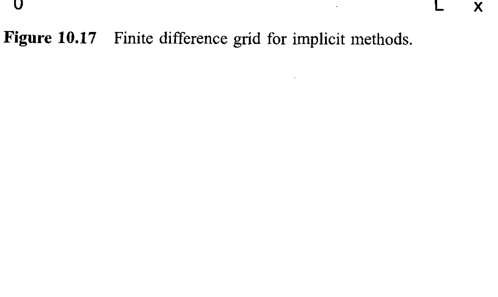


Figure 10.17 Finite difference grid for implicit methods.

$$\begin{aligned}
(1 + 2d)f_2^{n+1} - df_3^{n+1} &= f_2^n + d\bar{f}(0, t) = b_2 \\
-df_2^{n+1} + (1 + 2d)f_3^{n+1} - df_4^{n+1} &= f_3^n = b_3 \\
-df_3^{n+1} + (1 + 2d)f_4^{n+1} - df_5^{n+1} &= f_4^n = b_4 \\
\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots & \\
-df_{\text{imax}-2}^{n+1} + (1 + 2d)f_{\text{imax}-1}^{n+1} &= f_{\text{imax}-1}^n + d\bar{f}(L, t) = b_{\text{imax}-1}
\end{aligned} \tag{10.70}$$

Equation (10.70) comprises a tridiagonal system of linear algebraic equations. That system of equations may be written as

$$\mathbf{A}\mathbf{f}^{n+1} = \mathbf{b} \tag{10.71}$$

where $\mathbf{A}$ is the (imax − 2 × imax − 2) coefficient matrix, $\mathbf{f}^{n+1}$ is the (imax − 2 × 1) solution column vector, and $\mathbf{b}$ is the (imax − 2 × 1) column vector of nonhomogeneous terms. Equation (10.71) can be solved very efficiently by the Thomas algorithm presented in Section 1.5. Since the coefficient matrix $\mathbf{A}$ does not change from one time level to the next, LU factorization can be employed with the Thomas algorithm to reduce the computational effort even further.

The FTCS method and the BTCS method are both first order in time and second order in space. So what advantage, if any, does the BTCS method have over the FTCS method? The BTCS method is unconditionally stable. The time step can be much larger than the time step for the FTCS method. Consequently, the solution at a given time level can be reached with much less computational effort by taking time steps much larger than those allowed for the FTCS method. In fact, the time step is limited only by accuracy requirements.

Example 10.4. The BTCS method applied to the diffusion equation.

Let's solve the heat diffusion problem described in Section 10.1 by the BTCS method with $\Delta x = 0.1$ cm. For the first case, let $\Delta t = 0.5$ s, so $d = \alpha\, \Delta t / \Delta x^2 = 0.5$. The results at selected time levels are presented in Figure 10.18. It is obvious that the numerical solution is a good approximation of the exact solution. The general features of the numerical solution presented in Figure 10.18 are qualitatively similar to the numerical solution obtained by the FTCS method for $\Delta t = 0.5$ s and d = 0.5, which is presented in Figure 10.11. Although the results obtained by the BTCS method are smoother, there is no major difference. Consequently, there is no significant advantage of the BTCS method over the FTCS method for $d = 0.5$.

The numerical solutions at $t = 10.0$ s, obtained with $\Delta t = 1.0$, 2.5, 5.0, and 10.0 s, for which $d = 1.0$, 2.5, 5.0, and 10.0, respectively, are presented in Figure 10.19. These results clearly demonstrate the unconditional stability of the BTCS method. However, the numerical solution lags the exact solution seriously for the larger values of d. The advantage of the BTCS method over explicit methods is now apparent. If the decreased accuracy associated with the larger time steps is acceptable, then the solution can be

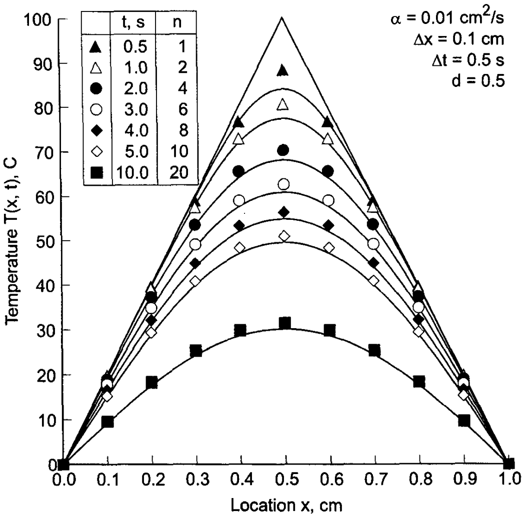


Figure 10.18 Solution by the BTCS method with $d = 0.5$.

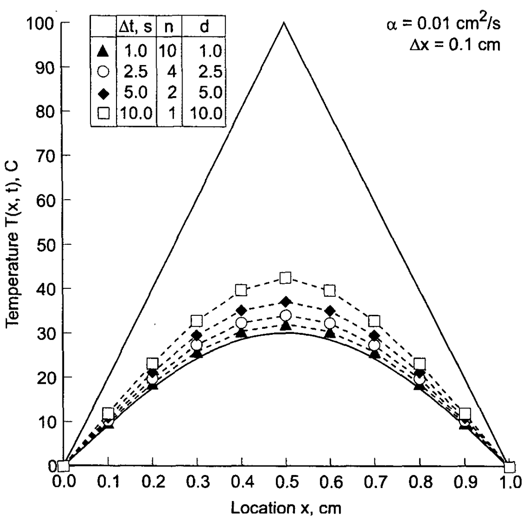


Figure 10.19 Solution by the BTCS method at $t = 10.0$ s.

obtained with less computational effort with the BTCS method than with the FTCS method. However, the results presented in Figure 10.19 suggest that large values of the diffusion number, d, lead to serious decreases in the accuracy of the solution.

The final results presented for the BTCS method are a parametric study in which the value of $T(0.4, 5.0)$ is calculated using values of $\Delta x = 0.1$, 0.05, 0.025, and 0.0125 cm, for values of $d = 0.5$, 1.0, 2.0, and 5.0. The value of Δt for each solution is determined by the specified values of Δx and d. The exact solution is $\bar{T}(0.5, 5.0) = 47.1255$ C. The results are presented in Table 10.5. The truncation error of the BTCS method is $0(\Delta t) + 0(\Delta x^2)$. For a constant value of d, $\Delta t = d\,\Delta x^2/\alpha$. Thus, as Δx is successively halved, Δt is quartered. Consequently, both the $0(\Delta t)$ error term and the $0(\Delta x^2)$ error term, and thus the total error, should decrease by a factor of approximately 4 as Δx is halved for a constant value of d. This result is clearly evident from the results presented in Table 10.5.

The backward-time centered-space (BTCS) method has an infinite numerical information propagation speed. Numerically, information propagates throughout the entire physical space during each time step. The diffusion equation has an infinite physical information propagation speed. Consequently, the BTCS method correctly models this feature of the diffusion equation.

When a PDE is nonlinear, the corresponding FDE is nonlinear. Consequently, a system of nonlinear FDEs must be solved. For one-dimensional problems, this situation is the same as described in Section 8.7 for ordinary differential equations, and the solution procedures described there also apply here. When systems of nonlinear PDEs are considered, a corresponding system of nonlinear FDEs is obtained at each solution point, and the combined equations at all of the solution points yield block tridiagonal systems of FDEs. For multidimensional physical spaces, banded matrices result. Such problems are frequently solved by alternating-direction-implicit (ADI) methods or approximate-factorization-implicit (AFI) methods, as described by Peaceman and Rachford (1955) and Douglas (1962). The solution of nonlinear equations and multidimensional problems are discussed in Section 10.9. The solution of a coupled system of several nonlinear multidimensional PDEs by an implicit finite difference method is indeed a formidable task.

Table 10.5 Parametric Study of $T(0.4, 5.0)$ by the BTCS Method

	$T(0.4, 5.0)$, C Error $(0.4, 5.0) = [T(0.4, 5.0) - \bar{T}(0.4, 5.0)]$, C			
Δx, cm	$d = 0.5$	$d = 1.0$	$d = 2.5$	$d = 5.0$
0.1	48.3810	49.0088	50.7417	53.1162
	1.2555	1.8830	3.6162	5.9907
0.05	47.4361	47.5948	48.0665	48.8320
	0.3106	0.4693	0.9410	1.7065
0.025	47.2029	47.2426	47.3614	47.5587
	0.0774	0.1171	0.2359	0.4332
0.0125	47.1448	47.1548	47.1845	47.2340
	0.0193	0.0293	0.0590	0.1085

In summary, the backward-time centered-space approximation of the diffusion equation is implicit, single step, consistent, $0(\Delta t) + 0(\Delta x^2)$, unconditionally stable, and convergent. Consequently, the time step is chosen based on accuracy requirements, not stability requirements. The BTCS method can be used to solve nonlinear PDEs, systems of PDEs, and multidimensional problems. However, in those cases, the solution procedure becomes quite complicated.

10.7.2 The Crank-Nicolson Method

The backward-time centered-space (BTCS) approximation of the diffusion equation $\bar{f}_t = \alpha \bar{f}_{xx}$, presented in the previous subsection, has a major advantage over explicit methods: It is unconditionally stable. It is an implicit single step method. The finite difference approximation of the spatial derivative is second order. However, the finite difference approximation of the time derivative is only first order. Using a second-order finite difference approximation of the time derivative would be an obvious improvement.

Crank and Nicolson (1947) proposed approximating the partial derivative $\bar{f}_t$ at grid point $(i, n + 1/2)$ by the second-order centered-time approximation obtained by combining Taylor series for $\bar{f}_i^{n+1}$ and $\bar{f}_k^n$. Thus,

$$\bar{f}_i^{n+1} = \bar{f}_i^{n+1/2} + \bar{f}_t|_i^{n+1/2}\left(\frac{\Delta t}{2}\right) + \tfrac{1}{2}\bar{f}_{tt}|_i^{n+1/2}\left(\frac{\Delta t}{2}\right)^2 + \tfrac{1}{6}\bar{f}_{ttt}|_i^{n+1/2}\left(\frac{\Delta t}{2}\right)^3 + \cdots \tag{10.72}$$

$$\bar{f}_i^{n} = \bar{f}_i^{n+1/2} - \bar{f}_t|_i^{n+1/2}\left(\frac{\Delta t}{2}\right) + \tfrac{1}{2}\bar{f}_{tt}|_i^{n+1/2}\left(\frac{\Delta t}{2}\right)^2 - \tfrac{1}{6}\bar{f}_{ttt}|_i^{n+1/2}\left(\frac{\Delta t}{2}\right)^3 + \cdots \tag{10.73}$$

Subtracting these two equations and solving for $\bar{f}_t|_i^{n+1/2}$ gives

$$\bar{f}_t|_i^{n+1/2} = \frac{\bar{f}_i^{n+1} - \bar{f}_i^n}{\Delta t} - \tfrac{1}{24}\bar{f}_{ttt}(\tau)\,\Delta t^2 \tag{10.74}$$

where $t^n \le \tau \le t^{n+1}$. Truncating the remainder term in Eq. (10.74) yields the second-order centered-time approximation of $\bar{f}_t$:

$$f_t|_i^{n+1/2} = \frac{f_i^{n+1} - f_i^n}{\Delta t} \tag{10.75}$$

The partial derivative $\bar{f}_{xx}$ at grid point $(i, n + \frac{1}{2})$ is approximated by

$$\bar{f}_{xx}|_i^{n+1/2} = \tfrac{1}{2}(\bar{f}_{xx}|_i^{n+1} + \bar{f}_{xx}|_i^n) \tag{10.76}$$

The order of the FDE obtained using Eqs. (10.75) and (10.76) is expected to be $0(\Delta t^2) + 0(\Delta x^2)$, but that must be proven from the MDE. The partial derivative $\bar{f}_{xx}$ at time levels n and $n + 1$ are approximated by the second-order centered-difference approximation, Eq. (10.23), applied at time levels n and $n + 1$, respectively. The finite difference stencil is illustrated in Figure 10.20. The resulting finite difference approximation of the one-dimensional diffusion equation is

$$\frac{f_i^{n+1} - f_i^n}{\Delta t} = \alpha\frac{1}{2}\left(\frac{f_{i+1}^{n+1} - 2f_i^{n+1} + f_{i-1}^{n+1}}{\Delta x^2} + \frac{f_{i+1}^n - 2f_i^n + f_{i-1}^n}{\Delta x^2}\right) \tag{10.77}$$

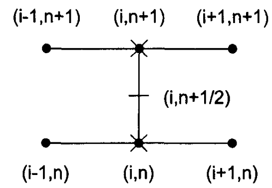


Figure 10.20 The Crank-Nicolson method stencil.

Rearranging Eq. (10.77) yields the Crank-Nicolson finite difference equation:

$$\boxed{\begin{aligned} &-df_{i-1}^{n+1} + 2(1+d)f_i^{n+1} - df_{i+1}^{n+1} \\ &= df_{i-1}^n + 2(1-d)f_i^n + df_{i+1}^n \end{aligned}} \tag{10.78}$$

where $d = \alpha\,\Delta t/\Delta x^2$ is the diffusion number.

The modified differential equation (MDE) obtained by writing Taylor series for $f(x, t)$ about point $(i, n + \frac{1}{2})$ is

$$\begin{aligned} f_t = {} & \alpha f_{xx} - \tfrac{1}{24}f_{ttt}\,\Delta t^2 + \cdots + \tfrac{1}{8}\alpha f_{xxtt}\,\Delta t^2 \\ & + \tfrac{1}{12}\alpha f_{xxxx}\Delta x^2 + \tfrac{1}{360}\alpha f_{xxxxxx}\Delta x^4 + \cdots \end{aligned} \tag{10.79}$$

As $\Delta t \to 0$ and $\Delta x \to 0$, all of the truncation error terms go to zero, and Eq. (10.79) approaches $f_t = \alpha f_{xx}$. Consequently, Eq. (10.78) is consistent with the diffusion equation. The leading truncation error terms are $0(\Delta t^2)$ and $0(\Delta x^2)$. From a von Neumann stability analysis, the amplification factor G is

$$G = \frac{1 - d(1 - \cos\theta)}{1 + d(1 - \cos\theta)} \tag{10.80}$$

The term $(1 - \cos\theta) \ge 0$ for all values of $\theta = (k_m\,\Delta x)$. Consequently $|G| \le 1$ for all positive values of d, and Eq. (10.78) is unconditionally stable. The Crank-Nicolson approximation of the diffusion equation is consistent and unconditionally stable. Consequently, by the Lax equivalence theorem, the Crank-Nicolson approximation of the diffusion equation is convergent.

Now consider the solution of the unsteady one-dimensional diffusion equation by the Crank-Nicolson method. The finite difference grid for advancing the solution from time level n to time level $n + 1$ is illustrated in Figure 10.17. For Dirichlet boundary conditions (i.e., the value of the function is specified at the boundaries), the finite difference equation must be applied only at the interior points, points 2 to imax − 1. At grid point 1, $f_1^{n+1} = \bar{f}(0, t)$, and at grid point imax, $f_{\text{imax}}^{n+1} = \bar{f}(L, t)$. The following set of simultaneous linear equations is obtained:

$$\begin{aligned} 2(1+d)f_2^{n+1} - df_3^{n+1} &= df_1^n + 2(1-d)f_2^n + df_3^n + d\bar{f}(0,t) = b_2 \\ -df_2^{n+1} + 2(1+d)f_3^{n+1} - df_4^{n+1} &= df_2^n + 2(1-d)f_3^n + df_4^n = b_3 \\ -df_3^{n+1} + 2(1+d)f_4^{n+1} - df_5^{n+1} &= df_3^n + 2(1-d)f_4^n + df_5^n = b_4 \\ &\cdots\cdots\cdots \\ -df_{\text{imax}-2}^{n+1} + 2(1+d)f_{\text{imax}-1}^{n+1} &= df_{\text{imax}-2}^n + 2(1-d)f_{\text{imax}-1}^n + df_{\text{imax}}^n + d\bar{f}(L,t) = b_{\text{imax}-1} \end{aligned} \tag{10.81}$$

Equation (10.81) comprises a tridiagonal system of linear algebraic equations, which is very similar to the system of equations developed in Section 10.7.1. for the backward-time centered-space (BTCS) method. Consequently, the present system of equations can be solved by the Thomas algorithm, as discussed in that section.

Like the backward-time centered space (BTCS) method, the Crank-Nicolson method is unconditionally stable. Consequently, the solution at a given time level can be reached with much less computational effort by taking large time steps. The time step is limited only by accuracy requirements.

Example 10.5. The Crank-Nicolson method applied to the diffusion equation

Let's solve the heat diffusion problem described in Section 10.1 by the Crank-Nicolson method with $\Delta x = 0.1$ cm. Let $\Delta t = 0.5$ s, so $d = 0.5$. The numerical solution is presented in Figure 10.21. As expected, the results are more accurate than the corresponding results presented in Figure 10.18 for the BTCS method.

The numerical solution at $t = 10.0$ s, obtained with $\Delta t = 1.0$, 2.5, 5.0, and 10.0 s, for which $d = 1.0$, 2.5, 5.0, and 10.0, respectively, is presented in Figure 10.22. These results clearly demonstrate the unconditional stability of the Crank-Nicolson method. However, an overshoot and oscillation exists in the numerical solution for all values of d considered in Figure 10.22. These oscillations are not due to an instability. They are an inherent feature of the Crank-Nicolson method when the diffusion number becomes large. The source of these oscillations can be determined by examining the eigenvalues of the

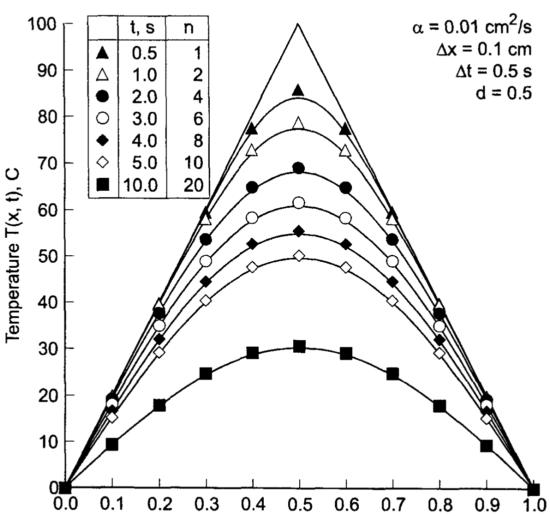


Figure 10.21 Solution by the Crank-Nicolson method with $d = 0.5$.

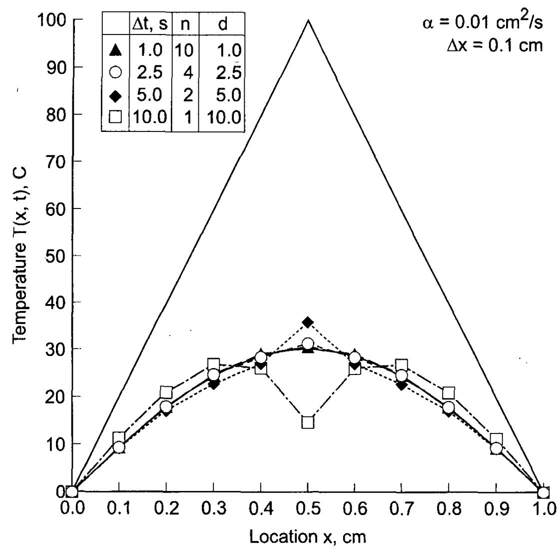


Figure 10.22 Solution by the Crank-Nicolson method at $t = 10.0$ s.

coefficient matrix for the complete system of linear equations, Eq. (10.81). The results presented in Figure 10.22 suggest that values of the diffusion number d much greater than 1.0 lead to a serious loss of accuracy in the transient solution.

The final results presented for the Crank-Nicolson method are a parametric study in which the value of $T(0.4, 5.0)$ is calculated using values of $\Delta x = 0.1$, 0.05, 0.025, and 0.0125 cm, for values of $d = 0.5$, 1.0, 2.0, and 5.0. The value of Δt for each solution is determined by the specified values of Δx and d. The exact solution is $\bar{T}(0.4, 5, 0) = 46.1255$ C. Results are presented in Table 10.6. The truncation error of the Crank-Nicolson method is $0(\Delta t^2) + 0(\Delta x^2)$. For a given value of d, $\Delta t = d\, \Delta x^2/\alpha$. Thus, as Δx is successively halved, Δt is quartered. Consequently, the $0(\Delta t^2)$ term should decrease by a factor of approximately 16 and the $0(\Delta x^2)$ term should decrease by a factor of approximately 4 as Δx is halved for a constant value of d. The results presented in Table 10.6 show that the total error decreases by a factor of approximately 4, indicating that the $0(\Delta x^2)$ term is the dominant error term.

The Crank-Nicolson method has an infinite numerical information propagation speed. Numerically, information propagates throughout the entire physical space during each time step. The diffusion equation has an infinite physical information propagation speed. Consequently, the Crank-Nicolson method correctly models this feature of the diffusion equation.

Table 10.6 Parametric Study of $T(0.4, 5.0)$ by the Crank-Nicolson Method

	$T(0.4, 5.0)$, C Error $(0.4, 5.0) = [T(0.4, 5.0) - \bar{T}(0.4, 5.0)]$, C			
Δx, cm	$d = 0.5$	$d = 1.0$	$d = 2.5$	$d = 5.0$
0.1	47.7269	47.7048	46.1511	47.9236
	0.6014	0.5793	−0.9744	0.7981
0.05	47.2762	47.2744	47.2831	47.0156
	0.1507	0.1489	0.1576	−0.1099
0.025	47.1632	47.1631	47.1623	47.1584
	0.0377	0.0376	0.0368	0.0329
0.0125	47.1349	47.1349	47.1349	47.1347
	0.0094	0.0094	0.0094	0.0092

The implicit Crank-Nicolson method can be used to solve nonlinear PDEs, systems of PDEs, and multidimensional PDEs. The techniques and problems are the same as those discussed in the previous section for the BTCS method and in Section 10.9.

In summary, the Crank-Nicolson approximation of the diffusion equation is implicit, single step, consistent, $0(\Delta t^2) + 0(\Delta x^2)$, unconditionally stable, and convergent. Consequently, the time step size is chosen based on accuracy requirements, not stability requirements.

10.8 DERIVATIVE BOUNDARY CONDITIONS

All of the finite difference solutions of the unsteady one-dimensional diffusion equation presented thus far in this chapter have been for Dirichlet boundary conditions, that is, the values of the function are specified on the boundaries. In this section, a procedure for implementing derivative, or Neumann, boundary conditions is presented.

The general features of a derivative boundary condition can be illustrated by considering a modification of the heat diffusion problem presented in Section 10.1, in which the thickness of the plate is $L = 0.5$ cm and the boundary condition on the right side of the plate is

$$\bar{T}_x(0.5, t) = 0.0 \tag{10.82}$$

The initial condition, $\bar{T}(x, 0.0)$, and the boundary condition of the left side, $\bar{T}(0.0, t)$, are the same as in the original problem. This problem is identical to the original problem due to the symmetry of the initial condition and the boundary conditions. The exact solution is given by Eq. (10.3), tabulated in Table 10.1, and illustrated in Figure 10.23. The solution smoothly approaches the asymptotic steady state solution, $\bar{T}(x, \infty) = 0.0$.

In this section, we will solve this problem numerically using the forward-time centered-space (FTCS) method at the interior points. The implementation of a derivative boundary condition does not depend on whether the problem is an equilibrium problem or a propagation problem, nor does the number of space dimensions alter the procedure. Consequently, the procedure presented in Section 8.5 for implementing a derivative boundary condition for one-dimensional equilibrium problems can be applied directly to

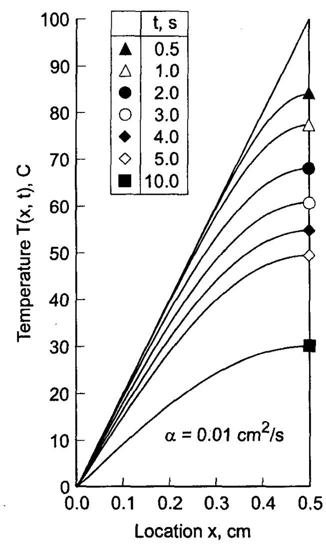


Figure 10.23 Exact solution with a derivative BC.

one-dimensional propagation problems. The finite difference grid for implementing a right-hand side derivative boundary condition is illustrated in Figure 10.24.

Let's apply the FTCS finite difference equation (FDE) at grid point I on the right-hand boundary, as illustrated in Figure 10.24. The FDE is Eq. (10.25):

$$f_I^{n+1} = f_I^n + d(f_{I-1}^n - 2f_I^n + f_{I+1}^n) \tag{10.83}$$

Grid point $I + 1$ is outside of the solution domain, so f_{I+1}^n is not defined. However, a value for f_{I+1}^n can be determined from the boundary condition on the right-hand boundary $\bar{f}_x|_I^n$ = known.

The finite difference approximation employed in Eq. (10.83) for the space derivative $\bar{f}_{xx}$ is second order. It's desirable to match this truncation error by using a second-order

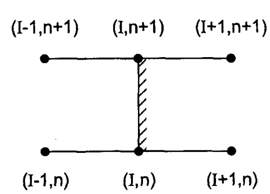


Figure 10.24 Finite difference stencil for right-hand side derivative BC.

finite difference approximation for the derivative boundary condition $\bar{f}_x|_I^n$ = known. Applying Eq. (10.21) at grid point I gives

$$\bar{f}_x|_I^n = \frac{\bar{f}_{I+1}^n - \bar{f}_{I-1}^n}{2\,\Delta x} + 0(\Delta x^2) \tag{10.84}$$

Truncating the remainder term and solving Eq. (10.84) for f_{I+1}^n gives

$$f_{I+1}^n = f_{I-1}^n + 2\bar{f}_x|_I^n\,\Delta x \tag{10.85}$$

Substituting Eq. (10.85) into Eq. (10.83) yields

$$f_I^{n+1} = f_I^n + d[f_{I-1}^n - 2f_I^n + (f_{I-1}^n + 2\bar{f}_x|_I^n\,\Delta x)] \tag{10.86}$$

Rearranging Eq. (10.86) gives the FDE applicable at the right-hand side boundary:

$$\boxed{f_I^{n+1} = f_I^n + 2d(f_{I-1}^n - f_I^n + \bar{f}_x|_I^n\,\Delta x)} \tag{10.87}$$

Equation (10.87) must be examined for consistency and stability. Consider the present example where $\bar{f}_x|_I^n = 0$. The modified differential equation (MDE) corresponding to Eq. (10.87) is

$$f_t = \alpha f_{xx} - \tfrac{1}{2} f_{tt}\,\Delta t - \tfrac{1}{3} f_{xxx}\Delta x + \tfrac{1}{12} f_{xxxx}\Delta x^2 + \cdots \tag{10.88}$$

As $\Delta x \to 0$ and $\Delta t \to 0$, all of the truncation error terms go to zero, and Eq. (10.88) approaches $f_t = \alpha f_{xx}$. Consequently, Eq. (10.87) is consistent with the diffusion equation. A matrix method stability analysis yields the stability criterion $d \le \frac{1}{2}$.

Example 10.6. Derivative boundary condition for the diffusion equation.

Let's work the example problem with $\Delta x = 0.1$ cm and $\Delta t = 0.1$ s, so $d = 0.1$, using Eq. (10.25) at the interior points and Eq. (10.87) at the right-hand boundary. The results are presented in Figure 10.25. The numerical solution is a good approximation of the exact solution. These results are identical to the results presented in Figure 10.10 for this problem which has a symmetrical initial condition and symmetrical boundary conditions.

10.9 NONLINEAR EQUATIONS AND MULTIDIMENSIONAL PROBLEMS

All of the finite difference equations and examples presented so far in this chapter are for the linear unsteady one-dimensional diffusion equation. Some of the problems which arise for nonlinear partial differential equations and multidimensional problems are discussed briefly in this section.

10.9.1 Nonlinear Equations

Consider the nonlinear unsteady one-dimensional convection-diffusion equation:

$$\boxed{\bar{f}_t + u(\bar{f})\,\bar{f}_x = \alpha(\bar{f})\,\bar{f}_{xx}} \tag{10.89}$$

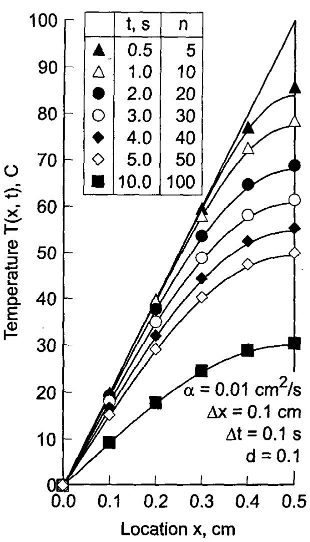


Figure 10.25 Solution with a derivative BC.

where the convection velocity u and the diffusion coefficient α depend on $\bar{f}$. The FTCS approximation of Eq. (10.89) is

$$\frac{f_i^{n+1} - f_i^n}{\Delta t} + u_i^n \frac{f_{i+1}^n - f_{i-1}^n}{2\,\Delta x} = \alpha_i^n \frac{f_{i+1}^n - 2f_i^n + f_{i-1}^n}{\Delta x^2} \tag{10.90}$$

The nonlinear coefficients are simply evaluated at the base point (i, n) where f_i^n, and hence u_i^n and α_i^n, are known. The FDE can be solved directly for f_i^{n+1}. The nonlinear coefficients cause no numerical complexities. This type of result is typical of all explicit finite difference approximations.

The BTCS approximations of Eq. (10.89) is

$$\frac{f_i^{n+1} - f_i^n}{\Delta t} + u_i^{n+1} \frac{f_{i+1}^{n+1} - f_{i-1}^{n+1}}{2\,\Delta x} = \alpha_i^{n+1} \frac{f_{i+1}^{n+1} - 2f_i^{n+1} + f_{i-1}^{n+1}}{\Delta x^2} \tag{10.91}$$

The nonlinear coefficients present a serious numerical problem. The nonlinear coefficients u_i^{n+1} and α_i^{n+1} depend on f_i^{n+1}, which is unknown. Equation (10.91), when applied at every point in the finite difference grid, yields a system of coupled nonlinear finite difference equations. The system of coupled nonlinear FDEs can be solved by simply lagging the nonlinear coefficients (i.e., letting $u_i^{n+1} = u_i^n$ and $\alpha_i^{n+1} = \alpha_i^n$), by iteration, by Newton's method, or by time linearization. Iteration and Newton's method are discussed in Section 8.7 for nonlinear one-dimensional boundary-value problems. Time linearization is presented in Section 7.11 for nonlinear one-dimensional initial-value problems. The Taylor series for $\bar{f}_i^{n+1}$ with base point (i, n) is

$$\bar{f}_i^{n+1} = \bar{f}_i^n + \bar{f}_t|_i^n \, \Delta t + 0(\Delta t^2) \tag{10.92}$$

The derivative $\bar{f}_t|_i^n$ is obtained from the PDE, which is evaluated at grid point (i, n) using the same spatial finite difference approximations used to derive the implicit FDE. Values of u_i^{n+1} and α_i^{n+1} can be evaluated for the value f_i^{n+1} obtained from Eq. (10.92). Time linearization requires a considerable amount of additional work. It also introduces additional truncation errors that depend on Δt, which reduces the accuracy and generally restricts the time step, thus reducing the advantage of unconditional stability associated with implicit finite difference equations.

10.9.2 Multidimensional Problems

All of the finite difference equations and examples presented so far in this chapter are for the linear unsteady one-dimensional diffusion equation. Some of the problems which arise for multidimensional problems are discussed in this section.

Consider the linear unsteady two-dimensional diffusion equation:

$$\boxed{\bar{f}_t = \alpha(\bar{f}_{xx} + \bar{f}_{yy})} \tag{10.93}$$

The FTCS approximation of Eq. (10.93) is

$$\frac{f_{i,j}^{n+1} - f_{i,j}^n}{\Delta t} = \alpha\left(\frac{f_{i+1,j}^n - 2f_{i,j}^n + f_{i-1,j}^n}{\Delta x^2} + \frac{f_{i,j+1}^n - 2f_{i,j}^n + f_{i,j-1}^n}{\Delta y^2}\right) \tag{10.94}$$

Equation (10.94) can be solved directly for $f_{i,j}^{n+1}$. No additional numerical complexities arise because of the second spatial derivative. For the three-dimensional diffusion equation, the additional derivative $\bar{f}_{zz}$ is present in the PDE. Its finite different approximations is simply added to Eq. (10.94) without further complications. This type of result is typical of all explicit finite difference approximations.

The BTCS approximation of Eq. (10.93) is

$$\frac{f_{i,j}^{n+1} - f_{i,j}^n}{\Delta t} = \alpha\left(\frac{f_{i+1,j}^{n+1} - 2f_{i,j}^{n+1} + f_{i-1,j}^{n+1}}{\Delta x^2} + \frac{f_{i,j+1}^{n+1} - 2f_{i,j}^{n+1} + f_{i,j-1}^{n+1}}{\Delta y^2}\right) \tag{10.95}$$

Applying Eq. (10.95) at every point in a two-dimensional finite difference grid yields a banded pentadiagonal matrix, which requires a large amount of computational effort. Successive-over-relaxation (SOR) methods can be applied for two-dimensional problems, but even that approach becomes almost prohibitive for three-dimensional problems. Alternating-direction-implicit (ADI) methods [Peaceman and Rachford (1955) and Douglas (1962)] and approximate-factorization-implicit (AFI) methods can be used to reduce the banded matrices to two (or three for three-dimensional problems) systems of tridiagonal matrices, which can be solved successively by the Thomas algorithm (see Section 1.5).

10.9.2.1 Alternating-Direction-Implicit (ADI) Method

The *alternating-direction-implicit* (*ADI*) approach consists of solving the PDE in two steps. In the first time step, the spatial derivatives in one direction, say y, are evaluated at

the known time level n and the other spatial derivatives, say x, are evaluated at the unknown time level $n+1$. On the next time step, the process is reversed. Consider the two-dimensional diffusion equation, Eq. (10.93). For the first step, the semidiscrete (i.e., time equation, discretization only) finite difference approximation yields

$$\frac{f_{i,j}^{n+1} - f_{i,j}^n}{\Delta t} = \alpha f_{xx}|_{i,j}^{n+1} + \alpha f_{yy}|_{i,j}^n \tag{10.96}$$

For the second step,

$$\frac{f_{i,j}^{n+2} - f_{i,j}^{n+1}}{\Delta t} = \alpha f_{xx}|_{i,j}^{n+1} + \alpha f_{yy}|_{i,j}^{n+2} \tag{10.97}$$

If the spatial derivatives in Eqs. (10.96) and (10.97) are replaced by second-order centered-difference approximations, Eqs. (10.96) and (10.97) both yield a tridiagonal system of FDEs, which can be solved by the Thomas algorithm (see Section 1.5). Ferziger (1981) shows that the alternating-direction-implicit method is consistent, $0(\Delta t^2) + (0(\Delta x^2) + 0(\Delta y^2)$, and unconditionally stable.

Alternating-direction-implicit (ADI) procedures can also be applied to three-dimensional problems, in which case a third permutation of Eqs. (10.96) and (10.97) involving the z-direction derivatives is required. A direct extension of the procedure presented above does not work. A modification that does work in three dimensions is presented by Douglas (1962) and Douglas and Gunn (1964).

The ADI method must be treated carefully at the $n+1$ time step at the boundaries. No problem arises for constant BCs. However, for time dependent BCs, Eq. (10.96), which is $0(\Delta t)$, yields less accurate solutions than Eq. (10.97), which is $0(\Delta t^2)$. When accurate BCs are specified, the errors in the solution at the boundaries at time steps $n+1$ and $n+2$ are different orders, which introduces additional errors into the solution. Ferziger (1981) discusses techniques for minimizing this problem.

10.9.2.2 Approximate-Factorization-Implicit (AFI) Method

The *approximate-factorization-implicit* (*AFI*) approach can be illustrated for the BTCS approximation of the two-dimensional diffusion equation, Eq. (10.93), by expressing it in the semidiscrete operator form

$$\frac{f_i^{n+1} - f_i^n}{\Delta t} = \alpha\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right) f_{i,j}^{n+1} \tag{10.98}$$

Collecting term yields the two-dimensional operator

$$\left[1 - \alpha\,\Delta t\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)\right] f_{i,j}^{n+1} = f_{i,j}^n \tag{10.99}$$

Equation (10.99) can be approximated by the product of two one-dimensional operators:

$$\left(1 - \alpha\,\Delta t\frac{\partial^2}{\partial x^2}\right)\left(1 - \alpha\Delta t\frac{\partial^2}{\partial y^2}\right) f_{i,j}^{n+1} = f_{i,j}^n \tag{10.100}$$

Equation (10.100) can be solved in two steps:

$$\left(1 - \alpha\,\Delta t \frac{\partial^2}{\partial y^2}\right) f^*_{i,j} = f^n_{i,j} \tag{10.101}$$

$$\left(1 - \alpha \Delta t \frac{\partial^2}{\partial x^2}\right) f^{n+1}_{i,j} = f^*_{i,j} \tag{10.102}$$

If the spatial derivatives in Eqs. (10.101) and (10.102) are replaced by three-point second-order centered-difference approximations, Eqs. (10.101) and (10.102), both yield a tridiagonal sytstem of FDEs, which can be solved by the Thomas algorithm (see Section 1.5).

Multiplying the two operators in Eq. (10.100) yields the single operator

$$\left[1 - \alpha \Delta t\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right) + \alpha^2\,\Delta t^2 \frac{\partial^2}{\partial x^2}\frac{\partial^2}{\partial y^2}\right] f^{n+1}_{i,j} = f^n_{i,j} \tag{10.103}$$

The $0(\Delta t^2)$ term in Eq. (10.103) is not present in the original finite difference equation, Eq. (10.99). Thus, the factorization has introduced a local $0(\Delta t^2)$ error term into the solution. For this reason this approach is called an *approximate factorization*.The local error of the BTCS approximation is $0(\Delta t^2)$, so the approximate factorizaton preserves the order of the BTCS approximation.

Approximate factorization can be applied to three-dimensional problems, in which case a third one-dimensional operator is added to Eq. (10.100) with a corresponding third step in Eqs. (10.101) and (10.102).

10.10 THE CONVECTION-DIFFUSION EQUATION

The solution of the parabolic unsteady diffusion equation has been discussed in Sections 10.2 to 10.9. The solution of the parabolic convection-diffusion equation is discussed in this section.

10.10.1 Introduction

Consider the unsteady one-dimensional parabolic convection-diffusion equation for the generic dependent variable $\bar{f}(x, t)$:

$$\boxed{\bar{f}_t + u\bar{f}_x = \alpha \bar{f}_{xx}} \tag{10.104}$$

where u is the convection velocity and α is the diffusion coefficient. Since the classification of a PDE is determined by the coefficients of its highest-order derivatives, the presence of the first-order convection term $u\bar{f}_x$ in the convection-diffusion equation does not affect its classification. The diffusion equation and the convection-diffusion equation are both parabolic PDEs. However, the presence of the first-order convection term has a major influence on the numerical solution procedure.

Most of the concepts, techniques, and conclusions presented in Sections 10.2 to 10.9 for solving the diffusion equation are directly applicable, sometimes with very minor modifications, for solving the convection-diffusion equation. The finite difference grids and finite difference approximations presented in Section 10.3 also apply to the convection-diffusion equation. The concepts of consistency, order, stability, and convergence presented in Section 10.5 also apply to the convection-diffusion equation. The present section is devoted to the numerical solution of the convection-diffusion equation, Eq. (10.104).

The solution to Eq. (10.104) is the function $\bar{f}(x, t)$. This function must satisfy an initial condition at $t = 0, \bar{f}(x, 0) = F(x)$. The time coordinate has an unspecified (i.e, open) final value. Equation (10.104) is second order in the space coordinate x. Consequently, two boundary conditions (BCs) are required. These BCs may be of the Dirichlet type (i.e., specified values of $\bar{f}$), the Neumann type (i.e., specified values of $\bar{f}_x$), or the mixed type (i.e., specified combinations of $\bar{f}$ and $\bar{f}_x$). The space coordinate x must be a closed physical domain.

The convection-diffusion equation applies to problems in mass transport, momentum transport, energy transport, etc. Most people have some physical feeling for heat transfer due to its presence in our everyday life. Consequently, the convection-diffusion equation governing heat transfer in a porous plate is considered in this chapter to demonstrate numerical methods for solving the convection-diffusion equation. That equation is presented in Section III.8, Eq. (III.101), which is repeated below:

$$T_t + \mathbf{V} \cdot \nabla T = \alpha\, \nabla^2 T \tag{10.105}$$

where T is the temperature, $\mathbf{V}$ is the vector convection velocity, and α is the thermal diffusivity. For unsteady one-dimensional heat transfer, Eq. (10.105) becomes

$$T_t + uT_x = \alpha T_{xx} \tag{10.106}$$

The following problem is considered in this chapter to illustrate the behavior of finite difference methods for solving the convection-diffusion equation. A porous plate of thickness $L = 1.0$ cm is cooled by a fluid flowing through the porous material, as illustrated in Figure 10.26. The thermal conductivity of the porous material is small compared to the thermal conductivity of the fluid, so that heat conduction through the porous material itself is negligible compared to heat transfer through the fluid by convection and diffusion (i.e., conduction). The temperatures on the two faces of the plate are

$$T(0.0, t) = 0.0\ \mathrm{C} \quad \text{and} \quad T(L, t) = 100.0\ \mathrm{C} \tag{10.107}$$

The initial fluid velocity is zero, so the initial temperature distribution is the pure diffusion distribution:

$$T(x, 0.0) = 100.0x/L \qquad 0.0 \le x \le L \tag{10.108}$$

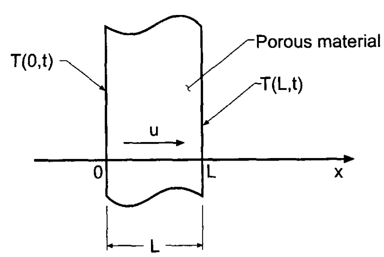


Figure 10.26 Heat convection-diffusion in a porous plate.

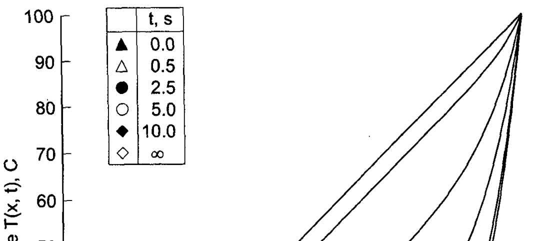


Figure 10.27 Exact solution of the heat convection-diffusion problem for $P = 10$.

This initial temperature distribution is illustrated by the top curve in Figure 10.27. The diffusion coefficient of the fluid is $\alpha = 0.01$ cm^2/s. At time $t = 0$, the fluid in the plate is instantaneously given a constant velocity $u = 0.1$ cm/s to the right. The temperature distribution $T(x, t)$ in the fluid is required.

The exact solution to this problem is obtained by replacing the original problem with two auxiliary problems. Both problems satisfy Eq. (10.106). For the first problem, $T(x, 0.0) = 0.0$, $T(0.0, t) = 0.0$, and $T(L, t) = 100.0$. For the second problem, $T(x, 0.0)$ is given by Eq. (10.108), $T(0.0, t) = 0.0$, and $T(L, t) = 0.0$. Since Eq. (10.106) is linear, the solution to the original problem is the sum of the solutions to the two auxiliary problems. The exact solution to each of the two auxiliary problems is obtained by assuming a product solution of the form $T(x, t) = X(x)\hat{T}(t)$, separating variables, integrating the resulting two ordinary differential equations, applying the boundary conditions at $x = 0$ and $x = L$, and superimposing an infinite number of harmonic functions (i.e., sines and cosines) in a Fourier series to satisfy the initial condition. The final result is

$$\boxed{T(x, t) = 100\left[\frac{\exp(Px/L) - 1}{\exp(P) - 1} + \frac{4\pi \exp(Px/2L)\sinh(P/2)}{\exp(P) - 1}\sum_{m=1}^{\infty} A_m + 2\pi \exp(Px/2L)\sum_{m=1}^{\infty} B_m\right]} \tag{10.109}$$

where P is the Peclet number

$$\boxed{P = \frac{uL}{\alpha}} \tag{10.110}$$

the coefficients A_m and B_m are given by

$$A_m = (-1)^m m\beta_m^{-1}\sin(m\pi x/L)e^{-\lambda_m t} \tag{10.111}$$

$$B_m = \left[(-1)^{m+1} m\beta_m^{-1}\left(1 + \frac{P}{\beta_m}\right)e^{-P/2} + \frac{mP}{\beta_m^2}\right]\sin\left(\frac{m\pi x}{L}\right)e^{-\lambda_m t} \tag{10.112}$$

and

$$\beta_m = \left(\frac{P}{2}\right)^2 + (m\pi)^2 \tag{10.113}$$

$$\lambda_m = \frac{u^2}{4\alpha} + \frac{m^2\pi^2\alpha}{L} = \frac{\alpha\beta_m}{L^2} \tag{10.114}$$

The exact transient solution at selected values of time t for $L = 1.0$ cm, $u = 0.1$ cm/s, and $\alpha = 0.01$ cm^2/s, for which the Peclet number $P = uL/\alpha = 10$, is tabulated in Table 10.7 and illustrated in Figure 10.27. As expected, heat flows out of the faces of the plate to the surroundings by diffusion, and heat is convected out of the plate by convection. The temperature distribution smoothly approaches the asymptotic steady-state solution

$$\boxed{T(x, \infty) = 100\frac{\exp(Px/L) - 1}{\exp(P) - 1}} \tag{10.115}$$

From Table 10.7, it can be seen that the transient solution has reached the asymptotic steady state by $t = 50.0$ s.

Table 10.7 Exact Solution of the Heat Convection-Diffusion Problem for $P = 10$

	Temperature $T(x, t)$, C							
t, s	$x = 0.0$	$x = 0.2$	$x = 0.4$	$x = 0.6$	$x = 0.7$	$x = 0.8$	$x = 0.9$	$x = 1.0$
0.0	0.00	20.00	40.00	60.00	70.00	80.00	90.00	100.00
0.5	0.00	15.14	35.00	55.00	65.00	75.02	85.43	100.00
1.0	0.00	11.36	30.05	50.00	60.02	70.18	81.57	100.00
1.5	0.00	8.67	25.39	45.02	55.07	65.50	77.99	100.00
2.0	0.00	6.72	21.25	40.14	50.18	60.91	74.55	100.00
2.5	0.00	5.29	17.71	35.46	45.41	56.43	71.20	100.00
5.0	0.00	1.80	7.13	17.74	25.64	36.78	56.13	100.00
10.0	0.00	0.30	1.38	4.81	9.09	18.39	40.96	100.00
50.00	0.00	0.03	0.24	1.83	4.97	13.53	36.79	100.00
∞	0.00	0.03	0.24	1.83	4.97	13.53	36.79	100.00

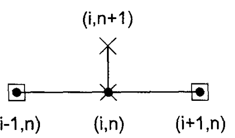


Figure 10.28 The FTCS stencil method.

10.10.2 The Forward-Time Centered-Space Method

In this section the convection-diffusion equation $\bar{f}_t + u\bar{f}_x = \alpha\bar{f}_{xx}$ is solved numerically by the forward-time centered-space (FTCS) method. The base point for the finite difference approximations (FDAs) of the individual exact partial derivatives is grid point (i, n). The partial derivative $\bar{f}_t$ is approximated by the first-order forward-difference FDA, Eq. (10.17), the partial derivative $\bar{f}_x$ is approximated by the second-order centered-difference FDA, Eq. (10.21), and the partial derivative $\bar{f}_{xx}$ is approximated by the second-order centered-difference FDA, Eq. (10.23). The corresponding finite difference stencil is illustrated in Figure 10.28. The resulting finite difference equation (FDE) is

$$\boxed{f_i^{n+1} = f_i^n - \frac{c}{2}(f_{i+1}^n - f_{i-1}^n) + d(f_{i+1}^n - 2f_i^n + f_{i-1}^n)} \tag{10.116}$$

where $c = u\,\Delta t/\Delta x$ is the convection number and $d = \alpha\,\Delta t/\Delta x^2$ is the diffusion number.

The modified differential equation (MDE) corresponding to Eq. (10.116) is

$$f_t + uf_x = \alpha f_{xx} - \tfrac{1}{2} f_{tt}\,\Delta t - \tfrac{1}{6} f_{ttt}\,\Delta t^2 - \cdots - \tfrac{1}{6} uf_{xxx}\Delta x^2 - \cdots$$
$$+ \tfrac{1}{12}\alpha f_{xxxx}\Delta x^4 + \cdots \tag{10.117}$$

As $\Delta t \to 0$ and $\Delta x \to 0$, Eq. (10.117) approaches $f_t + uf_x = \alpha f_{xx}$. Consequently, Eq. (10.116) is a consistent approximation of the convection-diffusion equation, Eq. (10.104). The FDE is $0(\Delta t) + 0(\Delta x^2)$. The amplification factor G corresponding to Eq. (10.116) is

$$G = (1 - 2d) + 2d\cos\theta - Ic\sin\theta \tag{10.118}$$

For $-\infty \le \theta \le \infty$, Eq. (10.118) represents an ellipse in the complex plane, as illustrated in Figure 10.29. The center of the ellipse is at $(1 - 2d + I0)$ and the axes are $2d$ and c. For stability, $|G| \le 1$, which requires that the ellipse lie on or within the unit circle $|G| = 1$. From Figure 10.29, two stability criteria are obvious. The real and imaginary axes of the ellipse must both be less than or equal to unity. From curves a and b,

$$c \le 1 \quad \text{and} \quad 2d \le 1 \tag{10.119}$$

In addition, at point $(1 + I0)$ the curvature of the ellipse must be greater than the curvature of the unit circle, or the ellipse will not remain within the unit circle even though $c < 1$ and $d < 1/2$, as illustrated by curve c. This condition is satisfied if

$$c^2 \le 2d \tag{10.120}$$

which, with $2d \le 1$, includes the condition $c \le 1$. Thus, the stability criteria for the FTCS approximation of the convection-diffusion equation are

$$\boxed{c^2 \le 2d \le 1} \tag{10.121}$$

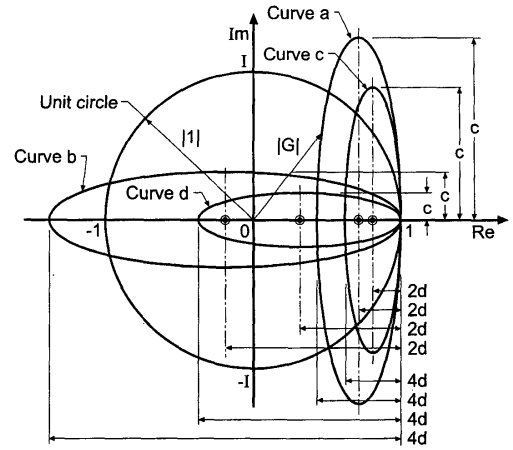


Figure 10.29 Locus of the amplification factor G for the FTCS method.

Consequently, the method is conditionally stable. Curve d in Figure 10.29 illustrates a stable condition. The FTCS approximation of the convection-diffusion equation is consistent and conditionally stable. Consequently, by the Lax equivalence theorem, it is a convergent approximation of that equation.

Example 10.7. The FTCS method applied to the convection-diffusion equation

Let's solve the heat convection-diffusion problem using Eq. (10.116) with $\Delta x = 0.1$ cm. The exact solution at selected times is presented in Table 10.7. The numerical solution for $\Delta t = 0.5$ s, for which $c = u\,\Delta t/\Delta x = (0.1)(0.5)/(0.1) = 0.5$ and $d = \alpha\,\Delta t/\Delta x^2 = (0.01)(0.5)/(0.1)^2 = 0.5$ is presented in Figure 10.30, it is apparent that the numerical solution is a reasonable approximation of the exact solution. Compare these results with the solution of the diffusion equation presented in Example 10.1 and illustrated in Figure 10.11. It is apparent that the solution of the convection-diffusion equation has larger errors than the solution of the diffusion equation. These larger errors are a direct consequence of the presence of the convection term $u\bar{f}_x$. As the solution progresses in time, the numerical solution smoothly approaches the steady-state solution.

At $t = 50$ s, the exact asymptotic steady state solution has been reached. The numerical solution at $t = 50.0$ s is a reasonable approximation of the steady state solution.

In summary, the FTCS approximation of the convection-diffusion equation is explicit, single step, consistent, $0(\Delta t) + 0(\Delta x^2)$, conditionally stable, and convergent. The FTCS approximation of the convection-diffusion equation yields reasonably accurate transient solutions.

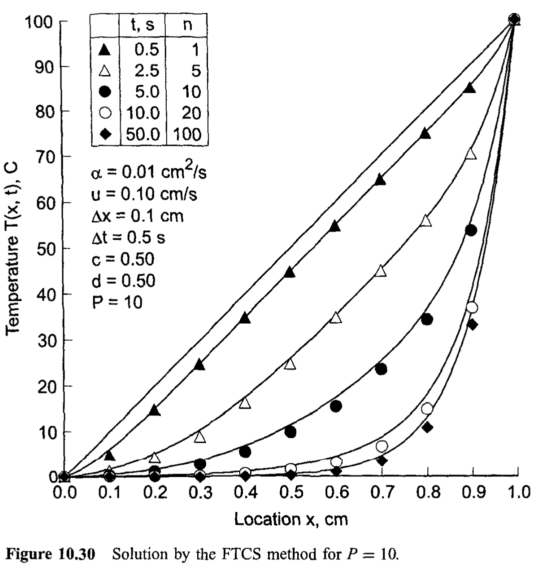


Figure 10.30 Solution by the FTCS method for $P = 10$.

10.10.3 The Backward-Time Centered-Space Method

The BTCS method is applied to the convection-diffusion equation $\bar{f}_t + u\bar{f}_x = \alpha\bar{f}_{xx}$ in this section. The base point for the finite difference approximation of the individual exact partial derivatives is grid point $(i, n+1)$. The partial derivative $\bar{f}_t$ is approximated by the first-order backward-difference FDA, Eq. (10.65), the partial derivative $\bar{f}_x$ is approximated by the second-order centered-difference FDA, Eq. (10.21), and the partial derivative $\bar{f}_{xx}$ is approximated by the second-order centered-difference FDA, Eq. (10.23), both evaluated at time level $n+1$. The corresponding finite difference stencil is illustrated in Figure 10.31. The resulting finite difference equation (FDE) is

$$\frac{f_i^{n+1} - f_i^n}{\Delta t} + u\frac{f_{i+1}^{n+1} - f_{i-1}^{n+1}}{2\,\Delta x} = \alpha\frac{f_{i+1}^{n+1} - 2f_i^{n+1} + f_{i-1}^{n+1}}{\Delta x^2} \tag{10.122}$$

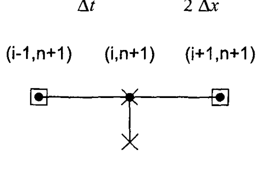


Figure 10.31 The BTCS method stencil.

Rearranging Eq. (10.122) yields

$$\boxed{-\left(\frac{c}{2}+d\right)f_{i-1}^{n+1} + (1+2d)f_i^{n+1} + \left(\frac{c}{2}-d\right)f_{i+1}^{n+1} = f_i^n} \tag{10.123}$$

where $c = u\,\Delta t/\Delta x$ is the convection number and $d = \alpha\Delta t/\Delta x^2$ is the diffusion number. Equation (10.123) cannot be solved explicitly for f_i^{n+1} because the two unknown neighboring values f_{i-1}^{n+1} and f_{i+1}^{n+1} also appear in the equation. Consequently, an implicit system of finite difference equations results.

The modified differential equation (MDE) corresponding to Eq. (10.123) is

$$f_t + uf_x = \alpha f_{xx} + \tfrac{1}{2} f_{tt}\,\Delta t - \tfrac{1}{6} f_{ttt}\,\Delta t^2 + \cdots - \tfrac{1}{6} uf_{xxx}\,\Delta x^2 + \cdots + \tfrac{1}{12}\alpha f_{xxxx}\,\Delta x^2 + \cdots \tag{10.124}$$

As $\Delta t \to 0$ and $\Delta x \to 0$, Eq. (10.124) approaches $f_t + uf_x = \alpha f_{xx}$. Consequently, Eq. (10.123) is a consistent approximation of the convection-diffusion equation, Eq. (10.104). From a von Neumann stability analysis, the amplification factor G is

$$G = \frac{1}{1 + 2d(1-\cos\theta) + Ic\sin\theta} \tag{10.125}$$

The term $(1-\cos\theta)$ is ≥ 0 for all values of $\theta = (k_m\,\Delta x)$. Consequently, the denominator of Eq. (10.125) is always ≥ 1, $|G| \leq 1$ for all values of c and d, and Eq. (10.123) is unconditionally stable. The BTCS approximation of the convection-diffusion equation is consistent and unconditionally stable. Consequently, by the Lax Equivalence Theorem it is a convergent approximation of that equation.

Consider now the solution of the convection-diffusion equation by the BTCS method. As discussed in Section 10.7 for the diffusion equation, a tridiagonal system of equations results when Eq. (10.123) is applied at every grid point. That system of equations can be solved by the Thomas algorithm (see Section 1.5). For the linear convection-diffusion equation, LU factorization can be used.

Example 10.8. The BTCS method applied to the convection-diffusion equation

Let's solve the heat convection-diffusion problem using Eq. (10.123) with $\Delta x = 0.1$ cm and $\Delta t = 0.5$ s. The transient solution for $\Delta t = 0.5$ s, for which $c = d = 0.5$ s, is presented in Figure 10.32. These results are a reasonable approximation of the exact transient solution. At $t = 50.0$ s, the exact asymptotic steady state solution has been reached. The numerical solution at $t = 50.0$ s is a reasonable approximation of the steady state solution.

The implicit BTCS method becomes considerably more complicated when applied to nonlinear PDEs, systems of PDEs, and multidimensional problems. A brief discussion of these problems is presented in Section 10.9.

In summary, the BTCS approximation of the convection-diffusion equation is implicit, single step, consistent, $0(\Delta t) + 0(\Delta x^2)$, unconditionally stable, and convergent. The implicit nature of the method yields a set of finite difference equations which must be solved simultaneously. For one-dimensional problems, that can be accomplished by the Thomas algorithm. The BTCS approximation of the convection-diffusion equation yields reasonably accurate transient solutions for modest values of the convection and diffusion numbers.

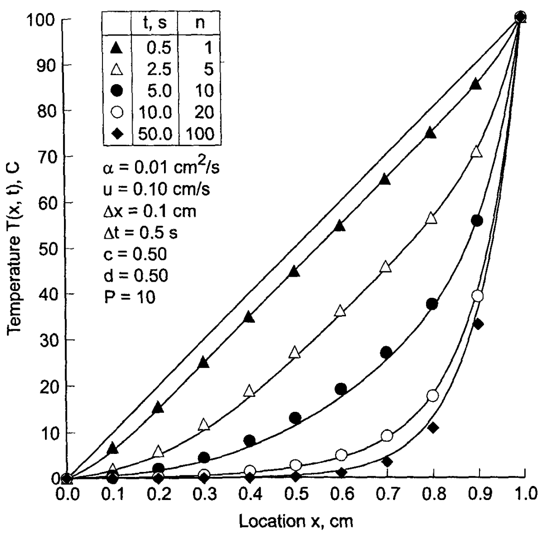


Figure 10.32 Solution by the BTCS method for $P = 10$.

10.11 ASYMPTOTIC STEADY STATE SOLUTION TO PROPAGATION PROBLEMS

Marching methods are employed for solving unsteady propagation problems, which are governed by parabolic and hyperbolic partial differential equations. The emphasis in those problems is on the transient solution itself.

Marching methods also can be used to solve steady equilibrium problems and steady mixed (i.e., elliptic-parabolic or elliptic-hyperbolic) problems as the asymptotic solution in time of an appropriate unsteady propagation problem. Steady equilibrium problems are governed by elliptic PDEs. Steady mixed problems are governed by PDEs that change classification from elliptic to parabolic or elliptic to hyperbolic in some portion of the solution domain, or by systems of PDEs which are a mixed set of elliptic and parabolic or elliptic and hyperbolic PDEs. Mixed problems present serious numerical difficulties due to the different types of solution domains (closed domains for equilibrium problems and open domains for propagation problems) and different types of auxiliary conditions (boundary conditions for equilibrium problems and boundary conditions and initial conditions for propagation problems). Consequently, it may be easier to obtain the solution of a steady mixed problem by reposing the problem as an unsteady parabolic or hyperbolic problem and using marching methods to obtain the asymptotic steady state solution. That approach to solving steady state problems is discussed in this section.

The appropriate unsteady propagation problem must be governed by a parabolic or hyperbolic PDE having the same spatial derivatives as the steady equilibrium problem or

the steady mixed problem and the same boundary conditions. As an example, consider the steady convection-diffusion equation:

$$\boxed{u\hat{f}_x = \alpha\hat{f}_{xx}} \tag{10.126}$$

The solution to Eq. (10.126) is the function $\hat{f}(x)$, which must satisfy two boundary conditions. The boundary conditions may be of the Dirichlet type (i.e., specified values of $\hat{f}$), the Neumann type (i.e., specified values of $\hat{f}_x$), or the mixed type (i.e., specified combinations of $\hat{f}$ and $\hat{f}_x$).

An appropriate unsteady propagation problem for solving Eq. (10.126) as the asymptotic solution in time is the unsteady convection-diffusion equation:

$$\bar{f}_t + u\bar{f}_x = \alpha\bar{f}_{xx} \tag{10.127}$$

The solution to Eq. (10.127) is the function $\bar{f}(x, t)$, which must satisfy an initial condition, $\bar{f}(x, 0) = F(x)$, and two boundary conditions. If the boundary conditions for $\bar{f}(x, t)$ are the same as the boundary conditions for $\hat{f}(x)$, then

$$\boxed{\hat{f}(x) = \lim_{t\to\infty} \bar{f}(x, t) = \bar{f}(x, \infty)} \tag{10.128}$$

As long as the asymptotic solution converges, the particular choice for the initial condition, $\bar{f}(x, 0) = F(x)$, should not affect the steady state solution. However, the steady state solution may be reached in fewer time steps if the initial condition is a reasonable approximation of the general features of the steady state solution.

The steady state solution of the transient heat convection-diffusion problem presented in Section 10.10 is considered in this section to illustrate the solution of steady equilibrium problems as the asymptotic solution of unsteady propagation problems. The exact solution to that problem is

$$\hat{T}(x) = 100\frac{e^{(Px/L)} - 1}{e^P - 1} \tag{10.129}$$

where $P = (uL/\alpha)$ is the Peclet number. The solution for $P = 10$ is tabulated in Table 10.7 as the last row of data in the table corresponding to $t = \infty$.

As shown in Figures 10.30 and 10.32, the solution of the steady state convection-diffusion equation can be obtained as the asymptotic solution in time of the unsteady convection-diffusion equation. The solution by the FTCS method required 100 time steps. The solution by the BTCS method also required 100 time steps. However, the BTCS method is unconditionally stable, so much larger time steps can be taken if the accuracy of the transient solution is not of interest. The results of this approach are illustrated in Example 10.9.

Example 10.9. Asymptotic steady state solution of the convection-diffusion equation

Let's solve the unsteady heat convection-diffusion problem for the asymptotic steady state solution with $\Delta x = 0.1$ cm by the BTCS method. Figure 10.33 presents seven solutions of the heat convection-diffusion equation, each one for a single time step with a different

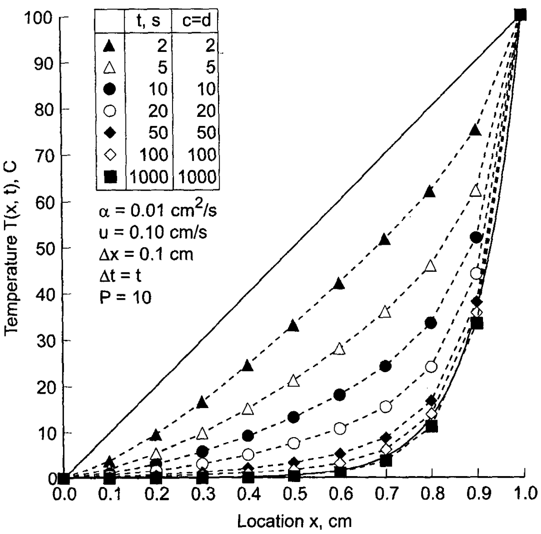


Figure 10.33 Single-step solutions by the BTCS method for $P = 10$.

value of Δt. As Δt is increased from 2.0 s to 1000.0 s, the single-step solution approaches the steady state solution more and more closely. In fact, the solution for $\Delta t = 1000.0$ s is essentially identical to the steady state solution and was obtained in a single step.

In summary, steady equilibrium problems, mixed elliptic/parabolic problems, and mixed elliptic/hyperbolic problems can be solved as the asymptotic steady state solution of an appropriate unsteady propagation problem. For linear problems, the asymptotic steady state solution can be obtained in one or two steps by the BTCS method, which is the recommended method for such problems. For nonlinear problems, the BTCS method becomes quite time consuming, since several time steps must be taken to reach the asymptotic steady-state solution. The asymptotic steady state approach is a powerful procedure for solving difficult equilibrium problems and mixed equilibrium/propagation problems.

10.12 PROGRAMS

Three FORTRAN subroutines for solving the diffusion equation are presented in this section:

1. The forward-time centered-space (FTCS) method
2. The backward-time centered-space (BTCS) method
3. The Crank-Nicolson (CN) method

The basic computational algorithms are presented as completely self-contained subroutines suitable for use in other programs. Input data and output statements are contained in a main (or driver) program written specifically to illustrate the use of each subroutine.

10.12.1 The Forward-Time Centered-Space (FTCS) Method

The diffusion equation is given by Eq. (10.4):

$$f_t = \alpha f_{xx} \tag{10.130}$$

When Dirichlet (i.e., specified f) boundary conditions are imposed, those values must be specified at the boundary points. That type of boundary condition is considered in this section. The first-order forward-time second-order centered-space (FTCS) approximation of Eq. (10.130) is given by Eq. (10.25):

$$f_i^{n+1} = f_i^n - d(f_{i+1}^n + 2.0f_i^n + f_{i-1}^n) \tag{10.131}$$

A FORTRAN subroutine, *subroutine ftcs*, for solving Eq. (131) is presented in Program 10.1. *Program main* defines the data set and prints it, calls *subroutine ftcs* to implement the solution, and prints the solution.

Program 10.1. The FTCS method for the diffusion equation program

```
      program main
c     main program to illustrate diffusion equation solvers
c     nxdim x-direction array dimension, nxdim = 11 in this program
c     ntdim t-direction array dimension, ntdim = 101 in this progra
c     imax  number of grid points in the x direction
c     nmax  number of time steps
c     iw    intermediate results output flag: 0 none, 1 all
c     ix    output increment: 1 every grid point, n every nth point
c     it    output increment: 1 every time step, n every nth step
c     f     solution array, f(i,n)
c     dx    x-direction grid increment
c     dt    time step
c     alpha diffusion coefficient
      dimension f(11,101)
      data nxdim,ntdim,imax,nmax,iw,ix,it/11,101,11,101, 0, 1, 10/
      data (f(i,1),i=1,11) / 0.0, 20.0, 40.0, 60.0, 80.0, 100.0,
     1 80.0, 60.0, 40.0, 20.0, 0.0 /
      data dx,dt,alpha,n,t / 0.1, 0.1, 0.01, 1, 0.0 /
      write (6,1000)
      call ftcs (nxdim,ntdim,imax,nmax,f,dx,dt,alpha,n,t,iw,ix)
      if (iw.eq.1) stop
      do n=1,nmax,it
```

```
         t=float(n-1)*dt
         write (6,1010) n,t,(f(i,n),i=1,imax,ix)
      end do
      stop
 1000 format (' Diffusion equation solver (FTCS method)'/' '/
     1 '  n',2x,'time',3x,'f(i,n)'/' ')
 1010 format (i3,f5.1,11f6.2)
      end

      subroutine ftcs(nxdim,ntdim,imax,nmax,f,dx,dt,alpha,n,t,iw,ix)
c     the FTCS method for the diffusion equation
      dimension f(nxdim,ntdim)
      d=alpha*dt/dx**2
      do n=1,nmax-1
         t=t+dt
         do i=2,imax-1
            f(i,n+1)=f(i,n)+d*(f(i+1,n)-2.0*f(i,n)+f(i-1,n))
         end do
         if (iw.eq.1) write (6,1000) n+1,t,(f(i,n+1),i=1,imax,ix)
      end do
      return
 1000 format (i3,f7.3,11f6.2)
      end
```

The data set used to illustrate *subroutine ftcs* is taken from Example 10.1. The output generated by the program is presented in Output 10.1.

A Neumann (i.e., derivative) boundary condition on the right-hand side of the solution domain can be implemented by solving Eq. (10.87) at grid point imax. Example 10.6 can be solved to illustrate the application of this boundary condition.

Output 10.1. Solution of the diffusion equation by the FTCS method

```
 Diffusion equation solver (FTCS method)

  n  time   f(i,n)

  1  0.0  0.00 20.00 40.00 60.00 80.00100.00 80.00 60.00 40.00 20.00  0.00
 11  1.0  0.00 19.96 39.68 58.22 72.81 78.67 72.81 58.22 39.68 19.96  0.00
 21  2.0  0.00 19.39 37.81 53.73 64.87 68.91 64.87 53.73 37.81 19.39  0.00
 31  3.0  0.00 18.21 35.06 48.99 58.30 61.58 58.30 48.99 35.06 18.21  0.00
 41  4.0  0.00 16.79 32.12 44.51 52.63 55.45 52.63 44.51 32.12 16.79  0.00
 51  5.0  0.00 15.34 29.25 40.39 47.61 50.11 47.61 40.39 29.25 15.34  0.00
 61  6.0  0.00 13.95 26.57 36.63 43.11 45.35 43.11 36.63 26.57 13.95  0.00
 71  7.0  0.00 12.67 24.11 33.20 39.06 41.07 39.06 33.20 24.11 12.67  0.00
 81  8.0  0.00 11.49 21.86 30.10 35.39 37.21 35.39 30.10 21.86 11.49  0.00
 91  9.0  0.00 10.42 19.82 27.28 32.07 33.72 32.07 27.28 19.82 10.42  0.00
101 10.0  0.00  9.44 17.96 24.72 29.07 30.56 29.07 24.72 17.96  9.44  0.00
```


10.12.2 The Backward-Time Centered-Space (BTCS) Method

The first-order backward-time second-order centered-space (BTCS) approximation of Eq. (10.130) is given by Eq. (10.67):

$$-df_{i-1}^{n+1} + (1 + 2d)f_i^{n+1} - df_{i+1}^{n+1} = f_i^n \tag{10.132}$$

A FORTRAN subroutine, *subroutine btcs*, for solving the system equation arising from the application of Eq. (10.132) at every interior point in a finite difference grid is presented in Program 10.2. *Subroutine thomas* presented in Section 1.8.3 is used to solve the tridiagonal system equation. Only the statements which are different from the statements in *program main* and *program ftcs* in Section 10.12.1 are presented. *Program main* defines the data set and prints it, calls *subroutine btcs* to implement the solution, and prints the solution.

Program 10.2. The BTCS method for the diffusion equation program

```
      program main
c     main program to illustrate diffusion equation solvers
      dimension f(11,101),a(11,3),b(11),w(11)
      call btcs(nxdim,ntdim,imax,nmax,f,dx,dt,alpha,n,t,iw,ix,a,b,w
 1000 format (' Diffusion equation solver (BTCS method)'/' '/
     1 '  n',2x,'time',3x,'f(i,n)'/' ')
      end

      subroutine btcs (nxdim,ntdim,imax,nmax,f,dx,dt,alpha,n,t,iw,
     1 ix,a,b,w)
c     implements the BTCS method for the diffusion equation
      dimension f(nxdim,ntdim),a(nxdim,3),b(nxdim),w(nxdim)
      d=alpha*dt/dx**2
      a(1,2)=1.0
      a(1,3)=0.0
      b(1)=0.0
      a(imax,1)=0.0
      a(imax,2)=1.0
      b(imax)=0.0
      do n=1,nmax-1
         t=t+dt
         do i=2,imax-1
            a(i,1)=-d
            a(i,2)=1.0+2.0*d
            a(i,3)=-d
            b(i)=f(i,n)
         end do
```

```
         call thomas (nxdim,imax,a,b,w)
         do i=2,imax-1
            f(i,n+1)=w(i)
         end do
         if (iw.eq.1) write (6,1000) n+1,t,(f(i,n+1),i=1,imax,ix)
      end do
      return
 1000 format (i3,f5.1,11f6.2)
      end
```

The data set used to illustrate *subroutine btcs* is taken from Example 10.4. The output generated by the program is presented in Output 10.2.

Output 10.2. Solution of the diffusion equation by the BTCS method

```
Diffusion equation solver (BTCS method)

 n  time   f(i,n)

 1  0.0  0.00 20.00 40.00 60.00 80.00100.00 80.00 60.00 40.00 20.00  0.00
 3  1.0  0.00 19.79 39.25 57.66 73.06 80.76 73.06 57.66 39.25 19.79  0.00
 5  2.0  0.00 19.10 37.40 53.62 65.58 70.28 65.58 53.62 37.40 19.10  0.00
 7  3.0  0.00 18.03 34.91 49.20 59.06 62.65 59.06 49.20 34.91 18.03  0.00
 9  4.0  0.00 16.75 32.19 44.91 53.39 56.39 53.39 44.91 32.19 16.75  0.00
11  5.0  0.00 15.42 29.50 40.90 48.38 50.99 48.38 40.90 29.50 15.42  0.00
13  6.0  0.00 14.11 26.93 37.22 43.90 46.22 43.90 37.22 26.93 14.11  0.00
15  7.0  0.00 12.87 24.53 33.84 39.86 41.94 39.86 33.84 24.53 12.87  0.00
17  8.0  0.00 11.73 22.33 30.77 36.21 38.09 36.21 30.77 22.33 11.73  0.00
19  9.0  0.00 10.67 20.31 27.97 32.90 34.60 32.90 27.97 20.31 10.67  0.00
21 10.0  0.00  9.70 18.46 25.42 29.90 31.44 29.90 25.42 18.46  9.70  0.00
```

10.12.3 The Crank-Nicolson (CN) Method

The Crank-Nicolson approximation of Eq. (10.130) is given by Eq. (10.78):

$$-df_{i-1}^{n+1} + 2(1+d)f_i^{n+1} - df_{i+1}^{n+1} = df_{i-1}^{n} + 2(1-d)f_i^{n} + df_{i+1}^{n} \tag{10.133}$$

A FORTRAN subroutine, *subroutine cn*, for implementing the system equation arising from the application of Eq. (10.133) at every interior point in a finite difference grid is presented in Program 10.3. Only the statements which are different from the statements in *program main* and *program btcs* in Section 10.12.2 are presented. *Program main* defines the data set and prints it, calls *subroutine cn* to implement the solution, and prints the solution.

Program 10.3. The CN method for the diffusion equation program

```
      program main
c     main program to illustrate diffusion equation solvers
      call cn (nxdim,ntdim,imax,nmax,f,dx,dt,alpha,n,t,iw,ix,a,b,w)
 1000 format (' Diffusion equation solver (CN method)'/' '/
     1 '  n',2x,'time',3x,'f(i,n)'/' ')
      end

      subroutine cn (nxdim,ntdim,imax,nmax,f,dx,dt,alpha,n,t,iw,ix,
     1 a,b,w)
c     the CN method for the diffusion equation
            a(i,2)=2.0*(1.0+d)
            b(i)=d*f(i-1,n)+2.0*(1.0-d)*f(i,n)+d*f(i+1,n)
      end

      subroutine thomas (ndim,n,a,b,x)
c     the thomas algorithm for a tridiagonal system
      end
```

The data set used to illustrate *subroutine cn* is taken from Example 10.5. The output generated by the program is presented in Output 10.3.

Output 10.3. Solution of the diffusion equation by the CN method

```
Diffusion equation solver (CN method)

 n  time   f(i,n)

 1  0.0  0.00 20.00 40.00 60.00 80.00100.00 80.00 60.00 40.00 20.00  0.00
 3  1.0  0.00 19.92 39.59 58.20 72.93 78.79 72.93 58.20 39.59 19.92  0.00
 5  2.0  0.00 19.34 37.76 53.75 64.97 69.06 64.97 53.75 37.76 19.34  0.00
 7  3.0  0.00 18.19 35.06 49.04 58.41 61.72 58.41 49.04 35.06 18.19  0.00
 9  4.0  0.00 16.80 32.15 44.59 52.74 55.59 52.74 44.59 32.15 16.80  0.00
11  5.0  0.00 15.36 29.30 40.48 47.73 50.24 47.73 40.48 29.30 15.36  0.00
13  6.0  0.00 13.98 26.64 36.73 43.23 45.48 43.23 36.73 26.64 13.98  0.00
15  7.0  0.00 12.70 24.18 33.31 39.18 41.21 39.18 33.31 24.18 12.70  0.00
17  8.0  0.00 11.53 21.94 30.21 35.52 37.35 35.52 30.21 21.94 11.53  0.00
19  9.0  0.00 10.46 19.90 27.39 32.21 33.86 32.21 27.39 19.90 10.46  0.00
21 10.0  0.00  9.49 18.04 24.84 29.20 30.70 29.20 24.84 18.04  9.49  0.00
```

10.12.4 Packages For Solving The Diffusion Equation

Numerous libraries and software packages are available for solving the diffusion equation. Many work stations and main frame computers have such libraries attached to their operating systems.

Many commercial software packages contain algorithms for integrating diffusion type (i.e., parabolic) PDEs. Due to the wide variety of parabolic PDEs governing physical problems, many parabolic PDE solvers (i.e., programs) have been developed. For this reason, no specific programs are recommended in this section.

10.13 SUMMARY

The numerical solution of parabolic partial differential equations by finite difference methods is discussed in this chapter. Parabolic PDEs govern propagation problems which have an infinite physical information propagation speed. They are solved numerically by marching methods. The unsteady one-dimensional diffusion equation $\bar{f}_t = \alpha \bar{f}_{xx}$ is considered as the model parabolic PDE in this chapter.

Explicit finite difference methods, as typified by the FTCS method, are conditionally stable and require a relatively small step size in the marching direction to satisfy the stability criteria. Implicit methods, as typified by the BTCS method, are unconditionally stable. The marching step size is restricted by accuracy requirements, not stability requirements. For accurate solutions of transient problems, the marching step size for implicit methods cannot be very much larger than the stable step size for explicit methods. Consequently, explicit methods are generally preferred for obtaining accurate transient solutions. Asymptotic steady state solutions can be obtained very efficiently by the BTCS method with a large marching step size.

Nonlinear partial differential equations can be solved directly by explicit methods. When solved by implicit methods, systems of nonlinear FDEs must be solved. Multidimensional problems can be solved directly by explicit methods. When solved by implicit methods, large banded systems of FDEs results. Alternating-direction-implicit (ADI) methods and approximate-factorization-implicit (AFI) methods can be used to solve multidimensional problems.

After studying Chapter 10, you should be able to:

1. Describe the physics of propagation problems governed by parabolic PDEs
2. Describe the general features of the unsteady diffusion equation
3. Understand the general features of pure diffusion
4. Discretize continuous physical space
5. Develop finite difference approximations of exact partial derivatives of any order
6. Develop a finite difference approximation of an exact partial differential equation
7. Understand the differences between an explicit FDE and an implicit FDE
8. Understand the theoretical concepts of consistency, order, stability, and convergence, and how to demonstrate each
9. Derive the modified differential equation (MDE) actually solved by a FDE
10. Perform a von Neumann stability analysis
11. Implement the forward-time centered-space method
12. Implement the backward-time centered-space method
13. Implement the Crank-Nicolson method

14. Describe the complications associated with nonlinear PDEs
15. Explain the difference between Dirichlet and Neumann boundary conditions and how to implement both.
16. Describe the general features of the unsteady convection-diffusion equation
17. Understand how to solve steady state problems as the asymptotic solution in time of an appropriate unsteady propagation problem
18. Choose a finite difference method for solving a parabolic PDE

EXERCISE PROBLEMS

Section 10.2 General Features of Parabolic PDEs

1. Consider the unsteady one-dimensional diffusion equation $\bar{f}_t = \alpha \bar{f}_{xx}$. Classify this PDE. Determine the characteristic curves. Discuss the significance of these results as regards domain of dependence, range of influence, signal propagation speed, auxiliary conditions, and numerical solution procedures.
2. Develop the exact solution of the heat diffusion problem presented in Section 10.1, Eq. (10.3).
3. By hand, calculate the exact solution for $T(0.5, 10.0)$.

Section 10.4 The Forward-Time Centered-Space (FTCS) Method

4. Derive the FTCS approximation of the unsteady one-dimensional diffusion equation, Eq. (10.25), including the leading truncation error terms in Δt and Δx.
5.* By hand calculation, determine the solution of the example heat diffusion problem by the FTCS method at $t = 0.5$ s for $\Delta x = 0.1$ cm and $\Delta t = 0.1$ s.
6. By hand calculation, derive the results presented in Figures 10.12 and 10.13.
7. Implement the program presented in Section 10.12.1 to reproduce Table 10.2. Compare the results with the exact solution presented in Table 10.1.
8. Solve Problem 7 with $\Delta x = 0.1$ cm and $\Delta t = 0.5$ s. Compare the results with the exact solution presented in Table 10.1.
9. Solve Problem 8 with $\Delta x = 0.05$ cm and $\Delta t = 0.125$ s. Compare the errors and the ratios of the errors for the two solutions at $t = 5.0$ s.

Section 10.5 Consistency, Order, Stability, and Convergence

Consistency and Order

10. Derive the MDE corresponding to the FTCS approximation of the diffusion equation, Eq. (10.25). Analyze consistency and order.
11. Derive the MDE corresponding to the Richardson approximation of the diffusion equation, Eq. (10.56). Analyze consistency and order.

12. Derive the MDE corresponding to the DuFort-Frankel approximation of the diffusion equation, Eq. (10.61). Analyze consistency and order.
13. Derive the MDE corresponding to the BTCS approximation of the diffusion equation, Eq. (10.67). Analyze consistency and order.
14. Derive the MDE corresponding to the Crank-Nicolson approximation of the diffusion equation, Eq. (10.78). Analyze consistency and order.

Stability

15. Perform a von Neumann stability analysis of Eq. (10.25).
16. Perform a von Neumann stability analysis of Eq. (10.56).
17. Perform a von Neumann stability analysis of Eq. (10.61).
18. Perform a von Neumann stability analysis of Eq. (10.67).
19. Perform a von Neumann stability analysis of Eq. (10.78).

Section 10.6 The Richardson and Du-Fort-Frankel methods

20. Derive the Richardson approximation of the unsteady one-dimensional diffusion equation, Eq. (10.56), including the leading truncation error terms in Δt and Δx.
21. Derive the DuFort-Frankel approximation of the unsteady one-dimensional diffusion equation, Eq. (10.61), including the leading truncation error terms in Δt and Δx.

Section 10.7 Implicit Methods

The Backward-Time Centered-Space (BTCS) Method

22. Derive the BTCS approximation of the unsteady one-dimensional diffusion equation, Eq. (10.67), including the leading truncation error terms in Δt and Δx.
23.* By hand calculation, determine the solution of the example heat diffusion problem by the BTCS method at $t = 0.5$ s for $\Delta x = 0.1$ cm and $\Delta t = 0.5$ s.
24. Implement the program presented in Section 10.12.2 to reproduce the results presented in Figure 10.18. Compare the results with the exact solution presented in Table 10.1.
25. Implement the program presented in Section 10.12.2 and repeat the calculations requested in the previous problem for $\Delta x = 0.05$ cm and $\Delta t = 0.125$ s. Compare the errors and ratios of the errors for the two solutions at $t = 10.0$ s.
25. Implement the program presented in Section 10.12.2 to reproduce the results presented in Figure 10.19.

The Crank-Nicolson Method

27. Derive the Crank-Nicolson approximation of the unsteady one-dimensional diffusion equation, Eq. (10.78), including the leading truncation error terms in Δt and Δx.
28.* By hand calculation, determine the solution of the example heat diffusion problem by the Crank-Nicolson method at $t = 0.5$ s for $\Delta x = 0.1$ cm and $\Delta t = 0.5$ s.
29. Implement the program presented in Section 10.12.3 to reproduce the results presented in Figure 10.21. Compare the results with the exact solution presented in Table 10.1.

30. Implement the program developed in Section 10.12.3 and repeat the calculations requested in the previous problem for $\Delta x = 0.05$ cm and $\Delta t = 0.25$ s. Compare the errors and the ratios of the errors for the two solutions at $t = 10.0$ s.
31. Use the program presented in Section 10.12.3 to reproduce the results presented in Figure 10.22.

Section 10.8 Derivative Boundary Conditions

32. Derive Eq. (10.87) for a right-hand side derivative boundary condition.
33.* By hand calculation using Eq. (10.87) at the boundary point, determine the solution of the example heat diffusion problem presented in Section 10.8 at $t = 2.5$ s for $\Delta x = 0.1$ cm and $\Delta t = 0.5$ s.
34. Modify the program presented in Section 10.12.1 to incorporate a derivative boundary condition on the right-hand boundary. Check out the program by reproducing Figure 10.25.

Section 10.9 Nonlinear Equations and Multidimensional Problems

Nonlinear Equations

35. Consider the following nonlinear parabolic PDE for the generic dependent variable $f(x, y)$, which serves as a model equation in fluid mechanics:

$$ff_x = \alpha f_{yy} \tag{A}$$

where $f(x, 0) = f_1$, $f(x, Y) = f_2$, and $f(0, y) = F(y)$. (a) Derive the FTCS approximation of Eq. (A). (b) Perform a von Neumann stability analysis of the linearized FDE. (c) Derive the MDE corresponding to the linearized FDE. Investigate consistency and order. (d) Discuss a strategy for solving this problem numerically.
36. Solve the previous problem for the BTCS method. Discuss a strategy for solving this problem numerically (a) using linearization, and (b) using Newton's method.
37. Equation (A) can be written as

$$(f^2/2)_x = \alpha f_{yy} \tag{B}$$

(a) Derive the FTCS approximation for this form of the PDE. (b) Derive the BTCS approximation for this form of the PDE.

Multidimensional Problems

38. Consider the unsteady two-dimensional diffusion equation:

$$\bar{f}_t = \alpha(\bar{f}_{xx} + \bar{f}_{yy}) \tag{C}$$

(a) Derive the FTCS approximation of Eq. (C), including the leading truncation error terms in Δt, Δx, and Δy. (b) Derive the corresponding MDE. Analyze consistency and order. (c) Perform a von Neumann stability analysis of the FDE.
39. Solve Problem 38 using the BTCS method.

40. Derive the FTCS approximation of the unsteady two-dimensional convection-diffusion equation:

$$\bar{f}_t + u\bar{f}_x + v\bar{f}_y = \alpha(\bar{f}_{xx} + \bar{f}_{yy}) \tag{D}$$

41. Derive the MDE for the FDE derived in Problem 40.
42. Derive the amplification factor G for the FDE derived in Problem 40.
43. Derive the BTCS approximation of the unsteady two-dimensional convection-diffusion equation, Eq. (D).
44. Derive the MDE for the FDE derived in Problem 43.
45. Derive the amplification factor G for the FDE derived in Problem 43.

Section 10.10 The Convection-Diffusion Equation

Introduction

46. Consider the unsteady one-dimensional convection-diffusion equation:

$$\bar{f}_t + u\bar{f}_x = \alpha\bar{f}_{xx} \tag{E}$$

Classify Eq. (E). Determine the characteristic curves. Discuss the significance of these results as regards domain of dependence, range of influence, signal propagation speed, auxiliary conditions, and numerical solution procedures.
47. Develop the exact solution for the heat transfer problem presented in Section 10.10, Eqs. (10.109) and (10.115).
48. By hand calculation, evaluate the exact solution of the heat transfer problem for $P = 10$ for $T(0.8, 5.0)$ and $T(0.8, \infty)$.

The Forward-Time Centered-Space Method

49. Derive the FTCS approximation of the unsteady one-dimensional convection-diffusion equation, Eq. (10.116), including the leading truncation error terms in Δt and Δx.
50. Derive the modified differential equation (MDE) corresponding to Eq. (10.116). Analyze consistency and order.
51. Perform a von Neumann stability analysis of Eq. (10.116).
52. By hand calculation, determine the solution of the example heat transfer problem for $P = 10.0$ at $t = 1.0$ s by the FTCS method for $\Delta x = 0.1$ cm and $\Delta t = 0.5$ s. Compare the results with the exact solution in Table 10.7.
53. Modify the program presented in Section 10.12.1 to implement the numerical solution of the example convection-diffusion problem by the FTCS method. Use the program to reproduce the results presented in Figure 10.30.
54. Use the program to solve the example convection-diffusion problem with $\Delta x = 0.05$ cm and $\Delta t = 0.125$ s.

The Backward-Time Centered-Space Method

55. Derive the BTCS approximation of the unsteady one-dimensional convection-diffusion equation, Eq. (10.123), including the leading truncation error terms in Δt and Δx.
56. Derive the modified differential equation (MDE) corresponding to Eq. (10.123). Analyze consistency and order.
57. Perform a von Neumann stability analysis of Eq. (10.123).

58. By hand calculation, determine the solution of the example heat transfer problem for $P = 10.0$ at $t = 1.0$ s with $\Delta x = 0.1$ cm and $\Delta t = 1.0$ s.
59. By hand calculation, estimate the asymptotic steady state solution of the example heat transfer problem for $P = 10.0$ with $\Delta x = 0.1$ cm by letting $\Delta t = 1000.0$ s.
60. Modify the program presented in Section 10.12.2 to implement the numerical solution of the example convection-diffusion problem by the BTCS method. Use the program to reproduce the results presented in Figure 10.32.
61. Use the program to solve the convection-diffusion problem for $\Delta x = 0.05$ cm and $\Delta t = 0.25$ s. Compare the errors and the ratios of the errors for the two solutions $t = 5.0$ s.

Section 10.11 Asymptotic Steady State Solution of Propagation Problems

62. Consider steady heat transfer in a rod with an insulated end, as discussed in Section 8.6. The steady boundary-value problem is specified by

$$\hat{T}_{xx} - \alpha^2(\hat{T} - T_a) = 0 \qquad \hat{T}(0) = T_1 \text{ and } \hat{T}_x(L) = 0 \tag{F}$$

where $\alpha^2 = hP/kA$, which is defined in Section 8.6. The exact solution for $T_1 = 100.0$, $\alpha = 2.0$, and $L = 1.0$ is given by Eq. (8.70) and illustrated in Figure 8.10. This steady state problem can be solved as the asymptotic solution in time of the following unsteady problem:

$$\beta\bar{T}_t = \bar{T}_{xx} - \alpha^2(\bar{T} - T_a) \qquad \bar{T}(0) = T_1 \text{ and } \bar{T}_x(L) = 0 \tag{G}$$

with the initial temperature distribution $\bar{T}(x, 0) = F(x)$, where $\beta = \rho C/k$, ρ is the density of the rod (kg/m^3), C is the specific heat (J/kg-K), and k is the thermal conductivity (J/s-m-K). Equation (G) can be derived by combining the analyses presented in Sections II.5 and II.6. (a) Derive Eq. (G). (b) Develop the FTCS approximation of Eq. (G). (c) Let $\hat{T}(0.0) = 100.0$, $\hat{T}_x(1.0) = 0.0$, $T_a = 0.0$, $L = 1.0$, $\alpha = 2.0$, $\beta = 10.0$, and the initial temperature distribution $\bar{T}(x, 0.0) = 100.0(1.0 - x)$. Solve for the steady state solution by solving Eq. (G) by the FTCS method with $\Delta x = 0.1$ cm and $\Delta t = 0.1$ s. Compare the results with the exact solution presented in Table 8.9.
63. Solve Problem 61 using the BTCS method. Try large values of Δt to reach the steady state as rapidly as possible.

Section 10.12 Programs

64. Implement the forward-time centered-space (FTCS) program for the diffusion equation presented in Section 10.12.1. Check out the program using the given data set.
65. Solve any of Problems 5 to 9 with the program.
66. Implement the backward-time centered-space (BTCS) program for the diffusion equation presented in Section 10.12.2. Check out the program using the given data set.
67. Solve any of Problem 23 to 26 with the program.
68. Implement the Crank-Nicolson program for the diffusion equation presented in Section 10.12.3. Check out the program using the given data set.
69. Solve any of Problems 28 to 31 with the program.