Chapter 3

Order of Accuracy of Finite Difference Schemes

In this chapter we study schemes based on how accurately they approximate partial differential equations. We present the Lax–Wendroff and Crank–Nicolson schemes, both of which are second-order accurate schemes. A convenient method for deriving higher order accurate schemes, as well as a convenient notation, is provided by the symbolic difference calculus. We also discuss the effect of boundary conditions on the stabilty of schemes. The chapter closes by presenting the Thomas algorithm for solving for the solution of implicit schemes.

3.1 Order of Accuracy

In the previous two chapters we classified schemes as acceptable or not acceptable only on the basis of whether or not they are convergent. This, via the Lax–Richtmyer equivalence theorem, led us to consider stability and consistency. However, different convergent schemes may differ considerably in how well their solutions approximate the solution of the differential equation. This may be seen by comparing Figures 1.3.6 and 1.3.8, which show solutions computed with the Lax–Friedrichs and leapfrog schemes. Both of these schemes are convergent for λ equal to 0.8, yet the leapfrog scheme has a solution that is closer to the solution of the differential equation than does the Lax–Friedrichs scheme. In this section we define the order of accuracy of a scheme, which can be regarded as an extension of the definition of consistency. The leapfrog scheme has a higher order of accuracy than does the Lax–Friedrichs scheme, and thus, in general, its solutions will be more accurate than those of the Lax–Friedrichs scheme. The proof that schemes with higher order of accuracy generally produce more accurate solutions is in Chapter 10.

Before defining the order of accuracy of a scheme, we introduce two schemes, which, as we will show, are more accurate than most of the schemes we have presented so far. We will also have to pay more attention to the way the forcing function, $f(t, x)$, is incorporated into the scheme.

The Lax–Wendroff Scheme

To derive the Lax–Wendroff scheme [37] for the one-way wave equation, we begin by using the Taylor series in time for $u(t + k, x)$, where u is a solution to the inhomogeneous

one-way wave equation (1.1.1),

$$u(t+k,x) = u(t,x) + ku_t(t,x) + \frac{k^2}{2}u_{tt}(t,x) + O(k^3).$$

We now use the differential equation that u satisfies,

$$u_t = -au_x + f,$$

and the relation

$$u_{tt} = -au_{tx} + f_t = a^2u_{xx} - af_x + f_t$$

to obtain

$$u(t+k,x) = u(t,x) - ak\,u_x(t,x) + \frac{a^2k^2}{2}u_{xx}(t,x) + kf - \frac{ak^2}{2}f_x + \frac{k^2}{2}f_t + O(k^3).$$

Replacing the derivatives in x by second-order accurate differences and f_t by a forward difference, we obtain

$$\begin{aligned} u(t+k,x) =& u(t,x) - ak\frac{u(t,x+h) - u(t,x-h)}{2h} \\ &+ \frac{a^2k^2}{2}\,\frac{u(t,x+h) - 2u(t,x) + u(t,x-h)}{h^2} \\ &+ \frac{k}{2}\big[f(t+k,x) + f(t,x)\big] - \frac{ak^2}{2}\,\frac{\big[f(t,x+h) - f(t,x-h)\big]}{2h} \\ &+ O(kh^2) + O(k^3). \end{aligned}$$

This gives the Lax–Wendroff scheme

$$\begin{aligned} v_m^{n+1} =& v_m^n - \frac{a\lambda}{2}(v_{m+1}^n - v_{m-1}^n) + \frac{a^2\lambda^2}{2}(v_{m+1}^n - 2v_m^n + v_{m-1}^n) \\ &+ \frac{k}{2}(f_m^{n+1} + f_m^n) - \frac{ak\lambda}{4}(f_{m+1}^n - f_{m-1}^n), \end{aligned} \tag{3.1.1}$$

or, equivalently,

$$\begin{aligned} &\frac{v_m^{n+1} - v_m^n}{k} + a\frac{v_{m+1}^n - v_{m-1}^n}{2h} - \frac{a^2k}{2}\,\frac{(v_{m+1}^n - 2v_m^n + v_{m-1}^n)}{h^2} \\ &\quad = \frac{1}{2}(f_m^{n+1} + f_m^n) - \frac{ak}{4h}(f_{m+1}^n - f_{m-1}^n). \end{aligned} \tag{3.1.2}$$

Figure 3.1 shows a comparison of the Lax–Wendroff scheme and the Lax–Friedrichs schemes for the computation used in Example 1.3.1. The solution for the Lax–Wendroff scheme is shown with circles; it is the one that has the greater maximum. In general, the solution of the Lax–Wendroff scheme is closer to the exact solution, which is also shown. Notice that the solution to the Lax–Wendroff scheme goes below the x-axis, while the solution of the Lax–Friedrichs scheme is always on or above the axis.

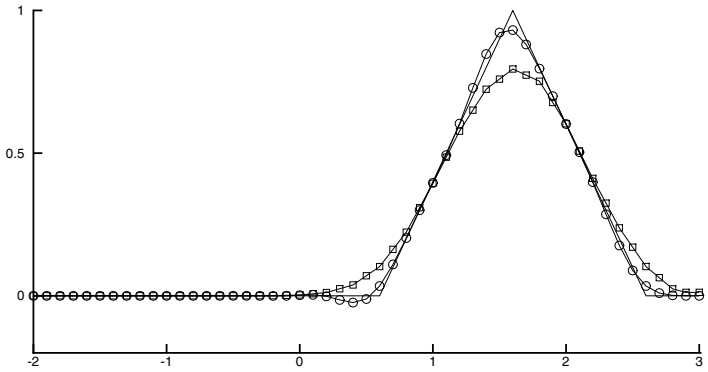

Figure 3.1. *Comparison of the Lax–Wendroff and Lax–Friedrichs schemes.*

The Crank–Nicolson Scheme

The Crank–Nicolson scheme is obtained by differencing the one-way wave equation (1.1.1) about the point $(t + k/2, x)$ to obtain second-order accuracy. We begin with the formula

$$u_t\left(t + \frac{1}{2}k, x\right) = \frac{u(t + k, x) - u(t, x)}{k} + O(k^2).$$

We also use the relation

$$\begin{aligned} u_x\left(t + \frac{1}{2}k, x\right) &= \frac{u_x(t + k, x) + u_x(t, x)}{2} + O(k^2) \\ &= \frac{1}{2}\left[\frac{u(t + k, x + h) - u(t + k, x - h)}{2h} + \frac{u(t, x + h) - u(t, x - h)}{2h}\right] \\ &\quad + O(k^2) + O(h^2). \end{aligned}$$

Using these approximations for $u_t + au_x = f$ about $(t + k/2, x)$, we obtain

$$\frac{v_m^{n+1} - v_m^n}{k} + a\frac{v_{m+1}^{n+1} - v_{m-1}^{n+1} + v_{m+1}^n - v_{m-1}^n}{4h} = \frac{f_m^{n+1} + f_m^n}{2} \tag{3.1.3}$$

or, equivalently,

$$\frac{a\lambda}{4}v_{m+1}^{n+1} + v_m^{n+1} - \frac{a\lambda}{4}v_{m-1}^{n+1} = -\frac{a\lambda}{4}v_{m+1}^n + v_m^n + \frac{a\lambda}{4}v_{m-1}^n + \frac{k}{2}(f_m^{n+1} + f_m^n). \tag{3.1.4}$$

A comparison of the Crank–Nicolson scheme and the backward-time and central-space scheme (1.6.1) is given in Figure 3.2.

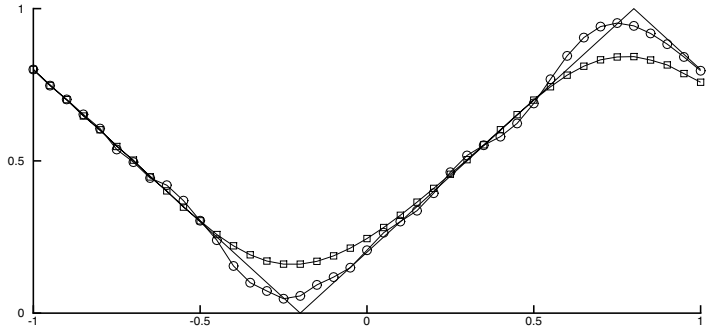

Figure 3.2. *Comparison of two implicit schemes.*

The solution for the initial data and exact solution is a saw-tooth curve and the exact solution is shown in the figure. The solution of the Crank–Nicolson scheme is shown with circles and is the solution closer to the exact solution. The solution to the backward-time central-space scheme is shown with squares marking the discrete points. In general, the Crank–Nicolson scheme has more accurate solutions than does the backward-time central scheme.

As we see from these two schemes that we have derived, a scheme for the partial differential equation $Pu = f$ can be written in general as $P_{k,h}v = R_{k,h}f$ in a natural way, where each expression $P_{k,h}v$ and $R_{k,h}f$ evaluated at a grid point (t_n, x_m) involves only a finite sum of terms involving $v_{m'}^{n'}$ or $f_{m'}^{n'}$, respectively. We are now able to give our first definition of the order of accuracy of a scheme.

Definition 3.1.1. *A scheme $P_{k,h}v = R_{k,h}f$ that is consistent with the differential equation $Pu = f$ is accurate of order p in time and order q in space if for any smooth function $\phi(t, x)$,*

$$P_{k,h}\phi - R_{k,h}P\phi = O(k^p) + O(h^q). \tag{3.1.5}$$

We say that such a scheme is accurate of order (p, q).

If we compare this definition with Definition 1.4.2, we see that consistency requires only that $P_{k,h}\phi - P\phi$ be $o(1)$, whereas Definition 3.1.1 takes into consideration the more detailed information on this convergence. The operator $R_{k,h}$ is required to be an approximation of the identity operator by the requirement that $P_{k,h}$ be consistent with P. The quantity $P_{k,h}\phi - R_{k,h}P\phi$ is called the *truncation error* of the scheme.

Example 3.1.1. We illustrate the use of this definition by showing that the Lax–Wendroff scheme (3.1.2) is accurate of order $(2, 2)$. We have, from (3.1.2),

$$P_{k,h}\phi = \frac{\phi_m^{n+1} - \phi_m^n}{k} + a\frac{\phi_{m+1}^n - \phi_{m-1}^n}{2h} - \frac{a^2k}{2}\frac{(\phi_{m+1}^n - 2\phi_m^n + \phi_{m-1}^n)}{h^2} \tag{3.1.6}$$

and

$$R_{k,h}f = \frac{1}{2}(f_m^{n+1} + f_m^n) - \frac{ak}{4h}(f_{m+1}^n - f_{m-1}^n)\,. \tag{3.1.7}$$

As before, we use the Taylor series on (3.1.6) evaluated at (t_n, x_m) to obtain

$$P_{k,h}\phi = \phi_t + \frac{k}{2}\phi_{tt} + a\phi_x - \frac{a^2k}{2}\phi_{xx} + O(k^2) + O(h^2). \tag{3.1.8}$$

For a smooth function $f(t, x)$, (3.1.7) becomes

$$R_{k,h}f = f + \frac{k}{2}f_t - \frac{ak}{2}f_x + O(k^2) + O(h^2),$$

and if $f = \phi_t + a\phi_x = P\phi$, this is

$$\begin{aligned} R_{k,h}P\phi &= \phi_t + a\phi_x + \frac{k}{2}\phi_{tt} + \frac{k}{2}a\phi_{xt} - \frac{ak}{2}\phi_{xt} - \frac{a^2k}{2}\phi_{xx} + O(k^2) + O(h^2) \\ &= \phi_t + a\phi_x + \frac{k}{2}\phi_{tt} - \frac{a^2k}{2}\phi_{xx} + O(k^2) + O(h^2), \end{aligned}$$

which agrees with (3.1.8) to $O(k^2) + O(h^2)$. Hence the Lax–Wendroff scheme (3.1.2) is accurate of order $(2, 2)$. □

We also see from this analysis that the Lax–Wendroff scheme with $R_{k,h}f_m^n = f_m^n$, i.e.,

$$v_m^{n+1} = v_m^n - \frac{a\lambda}{2}\left(v_{m+1}^n - v_{m-1}^n\right) + \frac{a^2\lambda^2}{2}\left(v_{m+1}^n - 2v_m^n + v_{m-1}^n\right) + k\, f_m^n, \tag{3.1.9}$$

is accurate of order $(1, 2)$.

Notice that to determine the order of accuracy we use the form (3.1.2) of the Lax–Wendroff scheme rather than (3.1.1), which is derived from (3.1.2) by multiplying by k and rearranging the terms. Without an appropriate normalization, in this case demanding that $P_{k,h}u$ be consistent with Pu, we can get incorrect results by multiplying the scheme by powers of k or h. An equivalent normalization is that $R_{k,h}$ applied to the function that is 1 everywhere gives the result 1, i.e.,

$$R_{k,h}1 = 1. \tag{3.1.10}$$

Definition 3.1.1 is not completely satisfactory. For example, it cannot be applied to the Lax–Friedrichs scheme, which contains the term $k^{-1}h^2\phi_{xx}$ in the Taylor series expansion of $P_{k,h}\phi$. We therefore give the following definition, which is more generally applicable. We assume that the time step is chosen as a function of the space step, i.e., $k = \Lambda(h)$, where Λ is a smooth function of h and $\Lambda(0) = 0$.

Definition 3.1.2. *A scheme $P_{k,h}v = R_{k,h}f$ with $k = \Lambda(h)$ that is consistent with the differential equation $Pu = f$ is accurate of order r if for any smooth function $\phi(t, x)$,*

$$P_{k,h}\phi - R_{k,h}P\phi = O(h^r).$$

If we take $\Lambda(h) = \lambda h$, then the Lax–Friedrichs scheme (1.3.5) is consistent with the one-way wave equation according to Definition 3.1.2.

Symbols of Difference Schemes

Another useful way of checking for the accuracy of a scheme is by comparing the symbols of the difference scheme to the symbol of the differential operator. Using the symbol is often a more convenient method than that given in Definitions 3.1.1 and 3.1.2.

Definition 3.1.3. *The symbol $p_{k,h}(s,\xi)$ of a difference operator $P_{k,h}$ is defined by*

$$P_{k,h}(e^{skn}e^{imh\xi}) = p_{k,h}(s,\xi)e^{skn}e^{imh\xi}.$$

That is, the symbol is the quantity multiplying the grid function $e^{skn}e^{imh\xi}$ after operating on this function with the difference operator.

As an example, for the Lax–Wendroff operator we have

$$p_{k,h}(s,\xi) = \frac{e^{sk}-1}{k} + \frac{ia}{h}\sin h\xi + 2\frac{a^2k}{h^2}\sin^2\frac{1}{2}h\xi$$

and

$$r_{k,h}(s,\xi) = \frac{1}{2}(e^{sk}+1) - \frac{iak}{2h}\sin h\xi.$$

The normalization (3.1.10) means

$$r_{k,h}(0,0) = 1.$$

Definition 3.1.4. *The symbol $p(s,\xi)$ of the differential operator P is defined by*

$$P(e^{st}e^{i\xi x}) = p(s,\xi)e^{st}e^{i\xi x}.$$

That is, the symbol is the quantity multiplying the function $e^{st}e^{ix\xi}$ after operating on this function with the differential operator.

In checking the accuracy of a scheme by using Taylor series and Definition 3.1.1, it is seen that the derivatives of ϕ serve primarily as arbitrary coefficients for the polynomials in h and k. The powers of the dual variables s and ξ can also serve as the coefficients of h and k in the definition of accuracy, as the following theorem states.

Theorem 3.1.1. *A scheme $P_{k,h}v = R_{k,h}f$ that is consistent with $Pu = f$ is accurate of order (p,q) if and only if for each value of s and ξ,*

$$p_{k,h}(s,\xi) - r_{k,h}(s,\xi)p(s,\xi) = O(k^p) + O(h^q), \tag{3.1.11}$$

or equivalently,

$$\frac{p_{k,h}(s,\xi)}{r_{k,h}(s,\xi)} - p(s,\xi) = O(k^p) + O(h^q). \tag{3.1.12}$$

Proof. By consistency we have for each smooth function ϕ that

$$P_{k,h}\phi - P\phi$$

tends to zero as h and k tend to zero; see Definition 1.4.2. Taking

$$\phi(t, x) = e^{st}e^{i\xi x},$$

we have that

$$p_{k,h}(s, \xi) - p(s, \xi) = o(1) \tag{3.1.13}$$

for each (s, ξ).

From Definition 3.1.1 for the order of accuracy and using this same function for $\phi(t, x)$, we have—by the definition of the symbol—that

$$p_{k,h}(s, \xi) - r_{k,h}(s, \xi)p(s, \xi) = O(k^p) + O(h^q),$$

which is (3.1.11). Hence from (3.1.13) and (3.1.11) we have that

$$r_{k,h}(s, \xi) = 1 + o(1), \tag{3.1.14}$$

and by dividing (3.1.11) by $r_{k,h}(s, \xi)$, we obtain (3.1.12).

To show that (3.1.12) implies (3.1.5), we again have by consistency that (3.1.14) holds, and hence (3.1.11) holds also. To obtain the Taylor series expansion for $P_{k,h}\phi$, we note that if

$$p_{k,h}(s, \xi) = \sum_{\ell, j \geq 0} A_{\ell, j}(k, h)s^{\ell}(i\xi)^j,$$

then

$$P_{k,h}\phi = \sum_{\ell, j \geq 0} A_{\ell, j}(k, h)\frac{\partial^{\ell+j}\phi}{\partial t^{\ell}\partial x^j}.$$

Therefore, (3.1.5) follows from (3.1.12). □

Corollary 3.1.2. *A scheme $P_{k,h}v = R_{k,h}f$ with $k = \Lambda(h)$ that is consistent with $Pu = f$ is accurate of order r if and only if for each value of s and ξ*

$$\frac{p_{k,h}(s, \xi)}{r_{k,h}(s, \xi)} - p(s, \xi) = O(h^r). \tag{3.1.15}$$

In practice, the form (3.1.11) is often more convenient than is (3.1.12) or (3.1.15) for showing the order of accuracy.

In Chapter 10 we show that if a scheme is accurate of order r, then the finite difference solution converges to the solution of the differential equation with the same order, provided that the initial data are sufficiently smooth.

Example 3.1.2. As an example of using Theorem 3.1.1, we prove that the Crank–Nicolson scheme (3.1.3) is accurate of order $(2, 2)$. From (3.1.3) we have that

$$p_{k,h}(s,\xi) = \frac{e^{sk}-1}{k} + ia\frac{e^{sk}+1}{2}\,\frac{\sin h\xi}{h}$$

and

$$r_{k,h}(s,\xi) = \frac{e^{sk}+1}{2}.$$

The left-hand side of (3.1.11) for this case is

$$\frac{e^{sk}-1}{k} + ia\frac{e^{sk}+1}{2}\,\frac{\sin h\xi}{h} - \frac{e^{sk}+1}{2}\,(s+ia\xi)\,. \tag{3.1.16}$$

We could use Taylor series expansions on this expression, but the work is reduced if we first multiply (3.1.16) by $e^{-sk/2}$. Since $e^{-sk/2}$ is $O(1)$, multiplying by it will not affect the determination of accuracy. We then have

$$\frac{e^{sk/2}-e^{-sk/2}}{k} + ia\frac{e^{sk/2}+e^{-sk/2}}{2}\,\frac{\sin h\xi}{h} - \frac{e^{sk/2}+e^{-sk/2}}{2}\,(s+ia\xi)\,. \tag{3.1.17}$$

The Taylor series expansions of the different expressions are then

$$\frac{e^{sk/2}-e^{-sk/2}}{k} = s + \frac{s^3k^2}{24} + O\left(k^4\right),$$

$$\frac{e^{sk/2}+e^{-sk/2}}{2} = 1 + \frac{s^2k^2}{8} + O\left(k^4\right),$$

and

$$\frac{\sin h\xi}{h} = \xi - \frac{\xi^3h^2}{6} + O\left(h^4\right).$$

Substituting these expansions in (3.1.17) we obtain

$$\begin{aligned}
&s + ia\xi + \frac{s^3k^2}{24} + ia\frac{s^2\xi k^2}{8} - ia\frac{\xi^3h^2}{6} \\
&\qquad - \left(1+\frac{s^2k^2}{8}\right)(s+ia\xi) + O\left(k^4+h^4+k^2h^2\right) \\
&= -\frac{k^2s^3}{12} - ia\frac{\xi^3h^2}{6} + O\left(k^4+h^4+k^2h^2\right) \\
&= O\left(k^2\right) + O\left(h^2\right).
\end{aligned}$$

Thus, the Crank–Nicolson scheme is accurate of order $(2, 2)$.

Using Taylor series expansions directly on (3.1.16) instead of (3.1.17) would have resulted in terms of order h and k in the expansion. These terms would have all canceled out, giving the same order of accuracy. Working with equation (3.1.17) greatly reduces the amount of algebraic manipulation that must be done to check the order of accuracy. Similar techniques can be used on other schemes. □

Order of Accuracy for Homogeneous Equations

For many initial value problems one is concerned only with the homogeneous equation $Pu = 0$ rather than the inhomogeneous equation $Pu = f$. In this case one can determine the order of accuracy without explicit knowledge of the operator $R_{k,h}$. We now show how this is done. It is important to make sure that our treatment of this topic applies to schemes for systems of differential equations as well as to single equations.

We begin by extending the set of symbols we have been using. Thus far we have considered symbols of finite difference schemes and symbols of partial differential operators, but we will find it convenient to extend the class of symbols.

Definition 3.1.5. *A symbol $a(s, \xi)$ is an infinitely differentiable function defined for complex values of s with $\operatorname{Re} s \geq c$ for some constant c and for all real values of ξ.*

This definition includes as symbols not only the symbols of differential operators and finite difference operators, but also many other functions. Symbols of differential operators are polynomials in s and ξ, and symbols of difference operators are polynomials in e^{ks} with coefficients that are either polynomials or rational functions of $e^{ih\xi}$.

Definition 3.1.6. *A symbol $a(s, \xi)$ is congruent to zero modulo a symbol $p(s, \xi)$, written*

$$a(s, \xi) \equiv 0 \bmod p(s, \xi),$$

if there is a symbol $b(s, \xi)$ such that

$$a(s, \xi) = b(s, \xi)\, p(s, \xi).$$

We also write

$$a(s, \xi) \equiv c(s, \xi) \bmod p(s, \xi)$$

if

$$a(s, \xi) - c(s, \xi) \equiv 0 \bmod p(s, \xi).$$

We can now define the order of accuracy for homogeneous equations.

Theorem 3.1.3. *A scheme $P_{k,h}v = 0$, with $k = \Lambda(h)$, that is consistent with the equation $Pu = 0$ is accurate of order r if*

$$p_{k,h}(s, \xi) \equiv O(h^r) \bmod p(s, \xi). \tag{3.1.18}$$

Proof. By Definition 3.1.6 the relation (3.1.18) holds if and only if there is a symbol $\tilde{r}_{k,h}(s,\xi)$ such that

$$p_{k,h}(s,\xi) - \tilde{r}_{k,h}(s,\xi)p(s,\xi) = O(h^r).$$

Since $p(s,\xi)$ is a linear polynomial in s with coefficients that are polynomials in ξ and since $p_{k,h}(s,\xi)$ is essentially a polynomial in e^{sk} with coefficients that are rational functions of $e^{ih\xi}$, it is not difficult to show that there is a symbol $r_{k,h}(s,\xi)$ such that

$$r_{k,h}(s,\xi) \equiv \tilde{r}_{k,h}(s,\xi) + O(h^r)$$

and $r_{k,h}(s,\xi)$ is a polynomial in e^{sk} with coefficients that are rational functions of $e^{ih\xi}$. The replacement of $\tilde{r}_{k,h}(s,\xi)$ by $r_{k,h}(s,\xi)$ is not strictly necessary for the proof, but it is important from the point of view of constructing an actual difference operator $R_{k,h}$ whose symbol is $r_{k,h}(s,\xi)$ and that can actually be used in computation. □

If we wish to use the Taylor series method of Definition 3.1.1 for checking the accuracy of homogeneous equations, then we can proceed in a way analogous to Definition 3.1.6 and Theorem 3.1.3. Equivalently, we can show that if

$$P_{k,h}\phi = O(h^r)$$

for each formal solution to $P\phi = 0$, then the scheme is accurate of order r. By saying a *formal solution*, we emphasize that we do not require knowledge of the existence of solutions or of the smoothness of the solution; we merely use the relation $P\phi = 0$ in evaluating $P_{k,h}\phi$. As an example, for the Lax–Wendroff scheme for the homogeneous equation (1.1.1), we have

$$\begin{aligned}
\phi(t+k,x) &= \phi(t,x) + k\phi_t(t,x) + \frac{k^2}{2}\phi_{tt}(t,x) + O(k^3) \\
&= \phi(t,x) + k\left[-a\phi_x(t,x)\right] + \frac{k^2}{2}\left(a^2\phi_{xx}\right) + O(k^3) \\
&= \phi(t,x) - \frac{a\lambda}{2}\left[\phi(t,x+h) - \phi(t,x-h)\right] \\
&\quad + \frac{a^2\lambda^2}{2}\left[\phi(t,x+h) - 2\phi(t,x) + \phi(t,x-h)\right] + O(k^3) + O(kh^2).
\end{aligned}$$

Using this last expression in the formula for $P_{k,h}$ given by (3.1.6) we see that the Lax–Wendroff scheme is second-order accurate. In this derivation we have used the relations

$$\phi_t = -a\phi_x$$

and

$$\phi_{tt} = -a\phi_{xt} = a^2\phi_{xx}.$$

From the preceding expression we obtain the scheme (3.1.1) without the terms involving f.

As is seen in Chapter 10, even for the homogeneous initial value problem it is important to know that the symbol $r_{k,h}(s,\xi)$ exists in order to prove that the proper order of convergence is attained.

We use symbols to prove the following theorem, proved by Harten, Hyman, and Lax [29], about schemes for the one-way wave equation and other hyperbolic equations.

Theorem 3.1.4. *An explicit one-step scheme for hyperbolic equations that has the form*

$$v_m^{n+1} = \sum_{\ell=-\infty}^{\infty} \alpha_\ell \, v_{m+\ell}^n \tag{3.1.19}$$

for homogeneous problems can be at most first-order accurate if all the coefficients α_ℓ are nonnegative, except for the trivial schemes for the one-way wave equation with $a\lambda = \ell$, where ℓ is an integer, given by

$$v_m^{n+1} = v_{m-\ell}^n. \tag{3.1.20}$$

Proof. We prove the theorem only for the one-way wave equation (1.1.1). As shown in the discussion of Section 1.1, this is sufficient for the general case. The symbol of the scheme (3.1.19) is

$$p_{k,h}(s,\xi) = \frac{e^{sk} - \sum \alpha_\ell e^{i\ell h\xi}}{k}.$$

If we allow for a right-hand-side symbol $r_{k,h}(s,\xi) = 1 + O(k) + O(h)$, the accuracy of the scheme is determined by considering the expression

$$\frac{e^{sk} - \sum \alpha_\ell \, e^{i\ell h\xi}}{k} - (1 + O(k) + O(h))(s + ia\xi).$$

If this expression is to be bounded as k tends to 0, we must have that this expression is finite when s and ξ are 0. This implies that

$$\sum_{\ell=-\infty}^{\infty} \alpha_\ell = 1. \tag{3.1.21}$$

The terms in s to the first power agree, and the coefficients of ks^2 will cancel only if

$$r_{k,h} = 1 + \tfrac{1}{2}sk + O(\xi k) + O(h).$$

The only occurrence of terms with the monomial $s\xi k$ appears in the product of $r_{k,h}$ with $s + ia\xi$, and these will cancel only if $r_{k,h} = 1 + \frac{1}{2}k(s - ia\xi) + O(h)$. Moreover, the

term $O(h)$ must actually be $O(h^2)$, since there is no term of the form sh coming from the symbol of the scheme. The terms to the first power of ξ are

$$-i\frac{h}{k}\sum_{\ell=-\infty}^{\infty}\alpha_\ell\ell \;-\; ia,$$

and this expression must be zero if the scheme is to be first-order accurate. This gives the relation

$$\sum_{\ell=-\infty}^{\infty}\alpha_\ell\;\ell = -a\lambda. \tag{3.1.22}$$

Next consider the terms that are the coefficients of ξ^2. They are

$$-\frac{h^2}{2k}\sum_{\ell=-\infty}^{\infty}\alpha_\ell\ell^2 + \frac{ka^2}{2}.$$

To have second-order accuracy this expression must also be zero, giving

$$\sum_{\ell=-\infty}^{\infty}\alpha_\ell\;\ell^2 = a^2\lambda^2. \tag{3.1.23}$$

We now use the Cauchy–Schwarz inequality on these three relations (3.1.21), (3.1.22), and (3.1.23), for the coefficients of the scheme. We have, starting with (3.1.22),

$$|a\lambda| = \left|\sum_{\ell=-\infty}^{\infty}\alpha_\ell\ell\right| = \left|\sum_{\ell=-\infty}^{\infty}\sqrt{\alpha_\ell}\sqrt{\alpha_\ell}\ell\right|$$

$$\le \left(\sum_{\ell=-\infty}^{\infty}\alpha_\ell\right)^{\frac{1}{2}}\left(\sum_{\ell=-\infty}^{\infty}\alpha_\ell\ell^2\right)^{\frac{1}{2}} = |a\lambda|.$$

Since the first and last expressions in this string of inequalities and equalities are the same, it follows that all the expressions are equal. However, the Cauchy–Schwarz inequality is an equality only if all the terms with the same index are proportional. This means there must be a constant c such that

$$\sqrt{\alpha_\ell}\;\ell = c\sqrt{\alpha_\ell} \quad \text{for all } \ell,$$

and this implies that at most one α_ℓ is nonzero. It is then easy to check that the only way these relations can be satisfied is if $a\lambda$ is an integer, and the resulting schemes are the trivial schemes (3.1.20). This proves the theorem. □

An examination of equations (3.1.21), (3.1.22), and (3.1.23) shows that the Lax–Wendroff scheme is the explicit one-step second-order accurate scheme that uses the fewest number of grid points. (See Exercise 3.1.1.)

One consequence of this theorem is that schemes such as we are discussing that are more than first-order accurate will have oscillatory solutions. For example, as shown in Figure 3.1 the solution to the Lax–Wendroff scheme goes below the x-axis. This is the result of some of the coefficients in the scheme (the α_ℓ) being negative. The Lax–Friedrichs scheme has all coefficients nonnegative (when $|a\lambda| \leq 1$) and it has a positive solution as illustrated in Figure 3.1.

Schemes of the form (3.1.19) for which all the coefficients are nonnegative are called monotone schemes. Monotone schemes have the property that the maximum of the solution does not increase with time and, similarly, the minimum does not decrease. The theorem says that monotone schemes can be at most first-order accurate.

Order of Accuracy of the Solution

We have spent some time on rigorously defining the order of accuracy of finite difference schemes, and the importance of this concept is that it relates directly to the accuracy of the solutions that are computed using these schemes. The order of accuracy of the solution of a finite difference scheme is a quantity that can be determined by computation. For our purposes here and in the exercises, it is sufficient to define the order of accuracy of the solution of a finite difference scheme as follows. If we have an initial value problem for a partial differential equation with solution $u(t, x)$ and a finite difference scheme, we use the initial data of the differential equation evaluated at the grid points as initial data for the scheme, i.e., $v_m^0 = u(0, x_m)$. We also assume that the time step is a function of the space step, i.e., $k = \Lambda(h)$. We then determine the error at time $t_n = nk$ by

$$\begin{aligned} \text{Error}(t_n) &= \|u(t_n, \cdot) - v^n\|_h \\ &= \left(h \sum_m |u(t_n, x_m) - v_m^n|^2 \right)^{1/2}, \end{aligned} \tag{3.1.24}$$

where the sum is over all grid points. The order of accuracy of the solution is defined to be that number r, if it exists, such that

$$\text{Error}(t_n) = O(h^r).$$

In Chapter 10 it is shown that for smooth initial data, the order of accuracy of the solution is equal to the order of accuracy of the scheme. Moreover, for those cases in which the data are not smooth enough for the accuracy of the solution to equal that of the scheme, it is shown how the order of accuracy of the solution depends on both the order of accuracy of the scheme and the smoothness of the initial data.

Table 3.1.1 displays results of several computations illustrating the order of accuracy of solutions of several schemes.

The schemes are applied to a periodic computation to remove all effects of boundary conditions. The value of λ was 0.9 for all computations. Columns 2 and 4 show the error as measured by (3.1.24) for the initial value problem for the one-way wave equation with $a = 1$ and initial data

$$u_0(x) = \sin(2\pi x) \qquad \text{for} -1 \leq x \leq 1.$$

The error in the solution was measured at time 5.4. The first time step for the leapfrog scheme was computed with the forward-time central-space scheme.

Notice that the order of the error for the first-order accurate, forward-time backward-space scheme tends to 1 and that for the second-order accurate leapfrog scheme tends to 2.

Table 3.1.1
Comparison of order of accuracy of solutions.

	Forward t backward x		Leapfrog scheme	
h	Error	Order	Error	Order
1/10	6.584e–1		5.945e–1	
1/20	4.133e–1	0.672	1.320e–1	2.17
1/40	2.339e–1	0.821	3.188e–2	2.05
1/80	1.247e–1	0.907	7.937e–3	2.01
1/160	6.445e–2	0.953	1.652e–3	2.26

The order of accuracy of the solution, as given here, is dependent on the initial data for the scheme and on the norm. For example, if the error is measured as the maximum value of $|u(t, x_m) - v_m^n|$, then the order of accuracy of the solution can be different than, and usually not more than, the order obtained by the preceding definition. This topic is discussed in more detail in Chapter 10.

Table 3.1.2
Comparison of order of accuracy of solutions.

	Lax–Wendroff		Lax–Friedrichs	
h	Error	Order	Error	Order
1/10	1.021e–1		2.676e–1	
1/20	4.604e–2	1.149	1.791e–1	0.579
1/40	2.385e–2	0.949	1.120e–1	0.677
1/80	1.215e–2	0.974	6.718e–2	0.738
1/160	6.155e–3	0.981	3.992e–2	0.751

Table 3.1.2 displays results of several computations with a solution that is not smooth enough to give the solution the same order of accuracy as that of the scheme. Columns 2 and 4 show the error as measured by (3.1.24) for the initial value problem for the one-way wave equation with $a = 1$ and initial data

$$u_0(x) = \begin{cases} 1 - 2|x| & \text{if } |x| \le 1/2, \\ 0 & \text{otherwise.} \end{cases}$$

The value of λ was 0.9 for all computations and the error in the solution was measured at time 5.4. For this exact solution, the Lax–Wendroff scheme has solutions that converge with an order of accuracy 1, while the Lax–Friedrichs scheme has solutions with order of

accuracy 0.75. Convergence estimates proved in Chapter 10 give the rate of convergence of solutions if the initial data are not smooth.

Exercises

3.1.1. Using equations (3.1.21), (3.1.22), and (3.1.23), show that the Lax–Wendroff scheme is the only explicit one-step second-order accurate scheme that uses only the grid points $x_{m-1}, x_m,$ and x_{m+1} to compute the solution at x_m for the next time step.

3.1.2. Solve $u_t + u_x = 0,\ -1 \le x \le 1,\ 0 \le t \le 1.2$ with $u(0, x) = \sin 2\pi x$ and periodicity, i.e., $u(t, 1) = u(t, -1)$. Use two methods:

(a) Forward-time backward-space with $\lambda = 0.8$,
(b) Lax–Wendroff with $\lambda = 0.8$.

Demonstrate the first-order accuracy of the solution of (a) and the second-order accuracy of the solution of (b) using $h = \frac{1}{10}, \frac{1}{20}, \frac{1}{40},$ and $\frac{1}{80}$. Measure the error in the L^2 norm (3.1.24) and the maximum norm. (In the error computation, do not sum both grid points at $x = -1$ and $x = 1$ as separate points.)

3.1.3. Solve the equation of Exercise 1.1.5,

$$u_t + u_x = -\sin^2 u,$$

with the scheme (3.1.9), treating the $-\sin^2 u$ term as $f(t, x)$. Show that the scheme is first-order accurate. The exact solution is given in Exercise 1.1.5. Use a smooth function, such as $\sin(x - t)$, as initial data and boundary data.

3.1.4. Modify the scheme of Exercise 3.1.3 to be second-order accurate and explicit. There are several ways to do this. One way uses

$$\sin^2 v_m^{n+1} = \sin^2 v_m^n + \sin 2v_m^n \left(v_m^{n+1} - v_m^n\right) + O\left(k^2\right).$$

Another way is to evaluate explicitly the f_t term in the derivation of the Lax–Wendroff scheme and eliminate all derivatives with respect to t using the differential equation.

3.1.5. Determine the order of accuracy of the Euler backward scheme in Exercise 2.2.6.

3.1.6. Show that the scheme discussed in Example 2.2.6 has the symbol

$$\frac{e^{sk} - \cos h\xi}{k} + 4ai\,\frac{\sin^2 \frac{1}{2}h\xi \sin h\xi}{h^3}$$

and discuss the accuracy of the scheme.

3.2 Stability of the Lax–Wendroff and Crank–Nicolson Schemes

In this section we demonstrate the stability of the Lax–Wendroff and Crank–Nicolson schemes. The stability analysis of the Lax–Wendroff scheme is informative because similar steps can be used to show the stability of other schemes. From (3.1.1) the Lax–Wendroff scheme for the one-way wave equation is

$$v_m^{n+1} = v_m^n - \frac{a\lambda}{2}\left(v_{m+1}^n - v_{m-1}^n\right) + \frac{a^2\lambda^2}{2}\left(v_{m+1}^n - 2v_m^n + v_{m-1}^n\right).$$

Notice that we set $f = 0$ as required to obtain the amplification factor. We substitute $g^{n'}e^{im'\theta}$ for $v_{m'}^{n'}$ and then cancel the factor of $g^n e^{im\theta}$, obtaining the following equation for the amplification factor:

$$\begin{aligned} g(\theta) &= 1 - \frac{a\lambda}{2}(e^{i\theta} - e^{-i\theta}) + \frac{a^2\lambda^2}{2}(e^{i\theta} - 2 + e^{-i\theta}) \\ &= 1 - ia\lambda \sin\theta - a^2\lambda^2(1 - \cos\theta) \\ &= 1 - 2a^2\lambda^2 \sin^2 \frac{1}{2}\theta - ia\lambda \sin\theta. \end{aligned}$$

To compute the magnitude of $g(\theta)$ we compute $|g(\theta)|^2$ by summing the squares of the real and imaginary parts:

$$|g(\theta)|^2 = \left(1 - 2a^2\lambda^2 \sin^2 \tfrac{1}{2}\theta\right)^2 + (a\lambda \sin\theta)^2 . \tag{3.2.1}$$

To work with these two terms we use the half-angle formula on the imaginary part, obtaining

$$\begin{aligned} |g(\theta)|^2 &= \left(1 - 2a^2\lambda^2 \sin^2 \tfrac{1}{2}\theta\right)^2 + \left(2a\lambda \sin \tfrac{1}{2}\theta \cos \tfrac{1}{2}\theta\right)^2 \\ &= 1 - 4a^2\lambda^2 \sin^2 \tfrac{1}{2}\theta + 4a^4\lambda^4 \sin^4 \tfrac{1}{2}\theta + 4a^2\lambda^2 \sin^2 \tfrac{1}{2}\theta \cos^2 \tfrac{1}{2}\theta . \end{aligned}$$

Notice that two terms have $a^2\lambda^2$ as a factor and one has $a^4\lambda^4$ as a factor. We combine the two terms with $a^2\lambda^2$ first, and then factor the common factors as follows:

$$\begin{aligned} |g(\theta)|^2 &= 1 - 4a^2\lambda^2 \sin^2 \tfrac{1}{2}\theta(1 - \cos^2 \tfrac{1}{2}\theta) + 4a^4\lambda^4 \sin^4 \tfrac{1}{2}\theta \\ &= 1 - 4a^2\lambda^2 \sin^4 \tfrac{1}{2}\theta + 4a^4\lambda^4 \sin^4 \tfrac{1}{2}\theta \\ &= 1 - 4a^2\lambda^2 \left(1 - a^2\lambda^2\right) \sin^4 \tfrac{1}{2}\theta. \end{aligned} \tag{3.2.2}$$

From this form for $|g(\theta)|^2$ we can see that it is less than or equal to 1 only if the quantity to the right of the first minus sign is nonnegative. All the factors except $1 - a^2\lambda^2$

are certainly nonnegative. To insure that $|g(\theta)|^2$ is always at most 1, we must have this quantity nonnegative; i.e., the Lax–Wendroff scheme is stable if and only if $|a\lambda| \le 1$.

For the Crank–Nicolson scheme from (3.1.3) we have

$$\frac{v_m^{n+1} - v_m^n}{k} + a\frac{v_{m+1}^{n+1} - v_{m-1}^{n+1} + v_{m+1}^n - v_{m-1}^n}{4h} = 0,$$

where we have set $f = 0$ as required in obtaining the amplification factor. Substituting $g^{n'}e^{im'\theta}$ for $v_{m'}^{n'}$ and then canceling, we obtain

$$\frac{g-1}{k} + a\frac{ge^{i\theta} - ge^{-i\theta} + e^{i\theta} - e^{-i\theta}}{4h} = 0.$$

Or,

$$g - 1 + a\lambda\frac{g+1}{2} i \sin\theta = 0,$$

from which we obtain the following expression for the amplification factor:

$$g(\theta) = \frac{1 - i\frac{1}{2}a\lambda\sin\theta}{1 + i\frac{1}{2}a\lambda\sin\theta}.$$

As the ratio of a complex number and its conjugate we have immediately that $|g(\theta)| = 1$. Alternatively,

$$|g(\theta)|^2 = \frac{1 + (\frac{1}{2}a\lambda\sin\theta)^2}{1 + (\frac{1}{2}a\lambda\sin\theta)^2} = 1.$$

This scheme is stable for any value of λ; it is *unconditionally stable.*

Exercises

3.2.1. Show that the (forward-backward) MacCormack scheme

$$\tilde{v}_m^{n+1} = v_m^n - a\lambda(v_{m+1}^n - v_m^n) + kf_m^n,$$

$$v_m^{n+1} = \tfrac{1}{2}(v_m^n + \tilde{v}_m^{n+1} - a\lambda(\tilde{v}_m^{n+1} - \tilde{v}_{m-1}^{n+1}) + kf_m^{n+1})$$

is a second-order accurate scheme for the one-way wave equation (1.1.1). Show that for $f = 0$ it is identical to the Lax–Wendroff scheme (3.1.1).

3.2.2. Show that the backward-time central-space scheme (1.6.1) is unconditionally stable.

3.2.3. Show that the box scheme

$$\begin{aligned}\frac{1}{2k}\left[(v_m^{n+1} + v_{m+1}^{n+1}) - (v_m^n + v_{m+1}^n)\right] \\ + \frac{a}{2h}\left[(v_{m+1}^{n+1} - v_m^{n+1}) + (v_{m+1}^n - v_m^n)\right] \\ = \frac{1}{4}\left(f_{m+1}^{n+1} + f_m^{n+1} + f_{m+1}^n + f_m^n\right)\end{aligned} \tag{3.2.3}$$

is an approximation to the one-way wave equation $u_t + au_x = f$ that is accurate of order $(2, 2)$ and is stable for all values of λ.

3.2.4. Using the box scheme (3.2.3), solve the one-way wave equation

$$u_t + u_x = \sin(x - t)$$

on the interval $[0, 1]$ for $0 \le t \le 1.2$ with $u(0, x) = \sin x$ and with $u(t, 0) = -(1 + t)\sin t$ as the boundary condition.

Demonstrate the second-order accuracy of the solution using $\lambda = 1.2$ and $h = \frac{1}{10}, \frac{1}{20}, \frac{1}{40}$, and $\frac{1}{80}$. Measure the error in the L^2 norm (3.1.24) and the maximum norm. To implement the box scheme note that v_0^{n+1} is given by the boundary data, and then each value of v_{m+1}^{n+1} can be determined from v_m^{n+1} and the other values.

3.2.5. Show that the following modified box scheme for $u_t + au_x = f$ is accurate of order $(2, 4)$ and is unconditionally stable. The scheme is

$$\begin{aligned}
&\frac{1}{16}(-v_{m+2}^{n+1} + 9v_{m+1}^{n+1} + 9v_m^{n+1} - v_{m-1}^{n+1}) \\
&\quad + \frac{a\lambda}{48}(-v_{m+2}^{n+1} + 27v_{m+1}^{n+1} - 27v_m^{n+1} + v_{m-1}^{n+1}) \\
&= \frac{1}{16}(-v_{m+2}^{n} + 9v_{m+1}^{n} + 9v_m^{n} - v_{m-1}^{n}) \\
&\quad - \frac{a\lambda}{48}(-v_{m+2}^{n} + 27v_{m+1}^{n} - 27v_m^{n} + v_{m-1}^{n}) \\
&\quad + \frac{k}{32}(-f_{m+2}^{n+1} + 9f_{m+1}^{n+1} + 9f_m^{n+1} - f_{m-1}^{n+1} \\
&\qquad - f_{m+2}^{n} + 9f_{m+1}^{n} + 9f_m^{n} - f_{m-1}^{n}).
\end{aligned}$$

3.3 Difference Notation and the Difference Calculus

To assist in our analysis and discussion of schemes we introduce some notation for finite differences. The forward and backward difference operators are defined by

$$\delta_+ v_m = \frac{v_{m+1} - v_m}{h} \tag{3.3.1}$$

and

$$\delta_- v_m = \frac{v_m - v_{m-1}}{h}, \tag{3.3.2}$$

respectively. We will occasionally use the notation δ_{x+} and δ_{x-} for these operators and define

$$\delta_{t+} v_m^n = \frac{v_m^{n+1} - v_m^n}{k}$$

for the forward difference in t; we similarly define δ_{t-}.

The central (first) difference operator δ_0 or δ_{x0} is defined by

$$\delta_0 v_m = \frac{1}{2}(\delta_+ v_m + \delta_- v_m) = \frac{v_{m+1} - v_{m-1}}{2h}$$

or, more succinctly,

$$\delta_0 = \tfrac{1}{2}(\delta_+ + \delta_-).$$

The central second difference operator is $\delta_+\delta_-$, which we also denote by δ^2. We have

$$\delta^2 v_m = \frac{v_{m+1} - 2v_m + v_{m-1}}{h^2},$$

and also

$$\delta^2 = (\delta_+ - \delta_-)/h.$$

We now demonstrate the use of this notation in deriving fourth-order accurate approximations to the first and second derivative operators. By Taylor series we have

$$\begin{aligned} \delta_0 u &= \frac{du}{dx} + \frac{h^2}{6}\frac{d^3u}{dx^3} + O(h^4) = \left(1 + \frac{h^2}{6}\frac{d^2}{dx^2}\right)\frac{du}{dx} + O(h^4) \\ &= \left(1 + \frac{h^2}{6}\delta^2\right)\frac{du}{dx} + O(h^4), \end{aligned} \tag{3.3.3}$$

where we have used

$$\frac{d^2 f}{dx^2} = \delta^2 f + O(h^2).$$

We may rewrite the formula (3.3.3) for $\delta_0 u$ as

$$\frac{du}{dx} = \left(1 + \frac{h^2}{6}\delta^2\right)^{-1} \delta_0 u + O(h^4). \tag{3.3.4}$$

The inverse of the operator $1 + \frac{h^2}{6}\delta^2$ is used only in a symbolic sense. In practice, the inverse is always eliminated by operating on both sides of the expression with $1 + \frac{h^2}{6}\delta^2$. Applying this formula to the simple equation

$$\frac{du}{dx} = f, \tag{3.3.5}$$

we have the equation

$$\left(1 + \frac{h^2}{6}\delta^2\right)^{-1} \delta_0 u(x_m) = f(x_m)$$

to fourth order. From this we have

$$\delta_0 u(x_m) = \left(1 + \frac{h^2}{6}\delta^2\right) f(x_m)$$

or

$$\begin{aligned}\frac{v_{m+1} - v_{m-1}}{2h} &= f_m + \frac{1}{6}(f_{m+1} - 2f_m + f_{m-1}) \\ &= \frac{1}{6}(f_{m+1} + 4f_m + f_{m-1}).\end{aligned}$$

Notice that replacing the right-hand side with only f_m is a second-order accurate formula. This formula will be used in Chapter 4.

Another fourth-order difference formula may be derived by using the formula

$$\delta_0 u = \frac{du}{dx} + \frac{h^2}{6}\delta^2\delta_0 u + O(h^4),$$

which may be rewritten as

$$\left(1 - \frac{h^2}{6}\delta^2\right)\delta_0 u = \frac{du}{dx} + O(h^4).$$

Applied to (3.3.5) we obtain the fourth-order approximation

$$\left(1 - \frac{h^2}{6}\delta^2\right)\delta_0 v_m = f_m$$

or

$$\frac{-v_{m+2} + 8v_{m+1} - 8v_{m-1} + v_{m-2}}{12h} = f_m.$$

For the second-order derivative we have the two formulas

$$\frac{d^2}{dx^2} = \left(1 + \frac{h^2}{12}\delta^2\right)^{-1}\delta^2 + O(h^4) \tag{3.3.6}$$

and

$$\frac{d^2}{dx^2} = \left(1 - \frac{h^2}{12}\delta^2\right)\delta^2 + O(h^4). \tag{3.3.7}$$

It is of some use to develop the formalism relating differences to derivatives. Let $\partial = \partial_x = \frac{d}{dx}$. Then by Taylor series,

$$u(x + h) = \sum_{j=0}^{\infty} \frac{h^j}{j!}\,\partial^j u(x) = e^{h\partial} u(x). \tag{3.3.8}$$

This formalism may be regarded as a purely symbolic operation for obtaining difference equations. If we adopt this view, then we should always check the accuracy of the formulas by the methods of Section 3.1. We may also regard this formalism as a shorthand notation for general Taylor series methods. For example, we can write out the expressions in (3.3.3) without writing down the symbol u. If we use this shorthand notation properly, the results will be consistent with the methods of Section 3.1, and there is no need to perform additional checks on the accuracy of schemes derived by this formalism. Therefore, we may express formulas (3.3.1) and (3.3.2) as

$$\delta_+ = \frac{e^{h\partial} - 1}{h} \tag{3.3.9}$$

and

$$\delta_- = \frac{1 - e^{-h\partial}}{h}. \tag{3.3.10}$$

Also,

$$\delta_0 = \frac{1}{2}(\delta_+ + \delta_-) = \frac{e^{h\partial} - e^{-h\partial}}{2h} = \frac{\sinh h\partial}{h} \tag{3.3.11}$$

and

$$\begin{aligned} \delta^2 = \delta_+\delta_- &= h^{-2}(e^{h\partial} - 1)(1 - e^{-h\partial}) \\ &= \left[h^{-1}(e^{h\partial/2} - e^{-h\partial/2})\right]^2 \\ &= \left(\frac{\sinh \frac{1}{2}h\partial}{\frac{1}{2}h}\right)^2. \end{aligned} \tag{3.3.12}$$

Notice that to obtain the symbols of these operators according to Definitions 3.1.3 and 3.1.4 we need only replace ∂ by $i\xi$.

We may generalize formula (3.3.4) as follows. From (3.3.11) we have

$$\delta_0 = \frac{\sinh h\partial}{h} = \frac{\sinh h\partial}{h\partial}\partial \tag{3.3.13}$$

and from (3.3.12) we have

$$h\delta = 2\sinh \tfrac{1}{2}h\partial,$$

where δ is defined by this relation. Thus

$$h\partial = 2\sinh^{-1} \tfrac{1}{2}h\delta$$

or

$$\partial = \frac{\sinh^{-1} \frac{1}{2}h\delta}{\frac{1}{2}h} \tag{3.3.14}$$

and so, from (3.3.13),

$$\delta_0 = \frac{\sinh[2\sinh^{-1}(\frac{1}{2}h\delta)]}{2\sinh^{-1} \frac{1}{2}h\delta}\partial$$

or

$$\begin{aligned}\partial &= \frac{2\sinh^{-1}(\frac{1}{2}h\delta)}{\sinh[2\sinh^{-1}(\frac{1}{2}h\delta)]}\delta_0 \\ &= \left[1+\left(\frac{h\delta}{2}\right)^2\right]^{-1/2} \frac{\sinh^{-1}\frac{1}{2}h\delta}{\frac{1}{2}h\delta}\delta_0.\end{aligned} \tag{3.3.15}$$

One may use the expression (3.3.15) to substitute for the derivatives with respect to x in differential equations and similarly use the square of (3.3.14) to substitute for the second derivative. By expanding the Taylor series to high enough powers of h, approximations to any order of accuracy can be obtained.

It is important to realize that not all schemes arise by a straightforward application of these formulas. The Lax–Wendroff scheme is a good example of a scheme relying on clever manipulations to obtain second-order accuracy in time, even though the scheme is a one-step scheme. Other examples of higher order accuracy schemes using similar ideas are given in Chapter 4.

Derivation of Schemes Using the Symbolic Calculus

To illustrate the use of the symbolic calculus, we derive several higher order accurate schemes.

Example 3.3.1. We first derive a $(4, 4)$ scheme for the one-way wave equation. The starting point for the derivation is the Taylor series expansion for a solution of $u_t + au_x = f$,

$$\begin{aligned}\frac{u_m^{n+2}-u_m^{n-2}}{4k} =& u_t + \frac{2k^2}{3}u_{ttt} + O(k^4) \\ =& \left(1+\frac{2k^2}{3}\delta_t^2\right)u_t + O(k^4) \\ =& \left(1+\frac{2k^2}{3}\delta_t^2\right)(-au_x+f) + O(k^4) \\ =& -\left(1+\frac{2k^2}{3}\delta_t^2\right)a\left(1-\frac{h^2\delta^2}{6}\right)\delta_0 u \\ &+\left(1+\frac{2k^2}{3}\delta_t^2\right)f + O(k^4) + O(h^4) \\ =& -a\left(1-\frac{h^2\delta^2}{6}\right)\delta_0\left(\frac{2u_m^{n+1}-u_m^n+2u_m^{n-1}}{3}\right) \\ &+\left(\frac{2f_m^{n+1}-f_m^n+2f_m^{n-1}}{3}\right) + O(k^4) + O(h^4).\end{aligned}$$

This gives the $(4, 4)$ scheme

$$\frac{v_m^{n+2} - v_m^{n-2}}{4k} + a\left(1 - \frac{h^2\delta^2}{6}\right)\delta_0\left(\frac{2v_m^{n+1} - v_m^n + 2v_m^{n-1}}{3}\right) = \frac{2f_m^{n+1} - f_m^n + 2f_m^{n-1}}{3}.$$

In Chapter 4 we present methods to show that this scheme is stable for

$$|a\lambda| < \frac{\sqrt{3}}{4}\frac{1}{(1+\sqrt{6})(\sqrt{6} - 3/2)^{1/2}} \approx 0.128825$$

(see Exercises 4.2.1 and 4.4.5). □

Example 3.3.2. As a second example we derive a scheme that is a hybrid between the Lax–Wendroff scheme (3.1.2) and the Crank–Nicolson scheme (3.1.3) for the one-way wave equation. We begin by considering $u(t_{n+1/3}, x)$:

$$\begin{aligned}\frac{u^{n+1} - u^n}{k} &= \frac{e^{2k\partial_t/3} - e^{-k\partial_t/3}}{k}u^{n+1/3} = u_t^{n+1/3} + \frac{k}{6}u_{tt}^{n+1/3} + O(k^2)\\ &= u_t^{n+1/3} + \frac{k}{6}\left(a^2u_{xx}^{n+1/3} + (f_t^{n+1/3} - af_x^{n+1/3})\right) + O(k^2),\end{aligned}$$

and using the relation $\varphi^{n+1/3} = (\varphi^{n+1} + 2\varphi^n)/3 + O(k^2)$, we obtain

$$\begin{aligned}&\frac{v_m^{n+1} - v_m^n}{k} + a\delta_0\left(\frac{v_m^{n+1} + 2v_m^n}{3}\right) - \frac{k}{6}a^2\delta^2\left(\frac{v_m^{n+1} + 2v_m^n}{3}\right)\\ &\qquad = \frac{f_m^{n+1} + f_m^n}{2} - \frac{ak}{6}\delta_0\left(\frac{f_m^{n+1} + 2f_m^n}{3}\right).\end{aligned} \tag{3.3.16}$$

This scheme is a $(2, 2)$ scheme and is stable for $|a\lambda| \le 3$. See Exercise 3.3.7. □

Example 3.3.3. For our last example we derive an implicit $(2, 2)$ scheme for the one-way wave equation. We have from (3.3.10) that

$$\partial_t v^{n+1} = -k^{-1}\ln(1 - k\delta_{t-})v^{n+1}$$

and by (3.3.8),

$$\begin{aligned}\partial_t u^{n+2/3} &= e^{-k\partial_t/3}\partial_t u^{n+1}\\ &= -\left(1 - \frac{1}{3}k\delta_{t-}\right)k^{-1}\ln(1 - k\delta_{t-})u^{n+1} + O(k^2)\\ &= \left(1 - \frac{1}{3}k\delta_{t-}\right)\left(\delta_{t-} + \frac{1}{2}k\delta_{t-}^2\right)u^{n+1} + O(k^2)\\ &= \left(\delta_{t-} + \frac{1}{6}k\delta_{t-}^2\right)u^{n+1} + O(k^2)\\ &= \frac{7u^{n+1} - 8u^n + u^{n-1}}{6k} + O(k^2).\end{aligned}$$

Using this relation with $u_x^{n+2/3} = (2u_x^{n+1} + u_x^n)/3 + O(k^2)$ we obtain

$$\frac{7v_m^{n+1} - 8v_m^n + v_m^{n-1}}{6k} + a\delta_0\left(\frac{2v_m^{n+1} + v_m^n}{3}\right) = f_m^{n+2/3}.$$

In Example 4.3.1 it is shown that this scheme is unconditionally stable. □

Exercises

3.3.1. Derive (3.3.6) and (3.3.7).

3.3.2. Obtain (3.3.4) directly from (3.3.15).

3.3.3. Obtain (3.3.7) from $\partial^2 = \frac{4}{h^2}\left(\sinh^{-1}\frac{1}{2}h\delta\right)^2$, which is equivalent to (3.3.14).

3.3.4. Determine the stability and accuracy of the following scheme, a modification of the Lax–Wendroff scheme, for $u_t + au_x = f$. For the stability analysis, but not the accuracy analysis, assume that λ is a constant:

$$v_m^{n+1} = v_m^n - \frac{1}{2}ak\left(1 - \frac{h^2}{6}\delta^2\right)^{-1}\left(\delta_0 v_m^n - \frac{1}{2}a^2k\delta^2 v_m^n\right)$$
$$+ \frac{k}{2}(f_m^{n+1} + f_m^n) - \frac{ak^2}{4}\left(1 - \frac{h^2}{6}\delta^2\right)^{-1}\delta_0 f_m^n.$$

3.3.5. Show that the scheme for $u_t + au_x = f$ given by

$$v_m^{n+1} = v_m^n - ak\left(1 - \frac{h^2\delta^2}{6}\right)\delta_0 v_m^n$$
$$+ \frac{a^2k^2}{2}\left[\left(\frac{4}{3} + a^2\lambda^2\right)\delta^2 v_m^n - \left(\frac{1}{3} + a^2\lambda^2\right)\delta_0^2 v_m^n\right]$$
$$+ \frac{k}{2}(f_m^{n+1} + f_m^n) - \frac{ak^2}{2}\delta_0 f_m^n$$

is accurate of order $(2, 4)$ and stable if

$$|a\lambda| \le \left(\frac{\sqrt{17} - 1}{6}\right)^{1/2} \approx 0.721469.$$

Note that $O(kh^2) \le O(k^2) + O(h^4)$. *Hint:* The computation of $|g|^2$ can be done similarly to that of the Lax–Wendroff scheme.

3.3.6. Show that the improved Crank–Nicolson scheme for $u_t + au_x = f$,

$$\frac{v_m^{n+1} - v_m^n}{k} + a\left(1 + \frac{h^2}{6}\delta^2\right)^{-1} \delta_0 \left(\frac{v_m^{n+1} + v_m^n}{2}\right) = \frac{f_m^{n+1} + f_m^n}{2},$$

is accurate of order $(2, 4)$ and is unconditionally stable. The scheme may also be written as

$$\begin{aligned}
&\frac{1}{6}v_{m+1}^{n+1} + \frac{2}{3}v_m^{n+1} + \frac{1}{6}v_{m-1}^{n+1} + \frac{a\lambda}{4}(v_{m+1}^{n+1} - v_{m-1}^{n+1}) \\
&\quad = \frac{1}{6}v_{m+1}^n + \frac{2}{3}v_m^n + \frac{1}{6}v_{m-1}^n - \frac{a\lambda}{4}(v_{m+1}^n - v_{m-1}^n) \\
&\qquad + \frac{k}{12}(f_{m+1}^{n+1} + 4f_m^{n+1} + f_{m-1}^{n+1} + f_{m+1}^n + 4f_m^n + f_{m-1}^n).
\end{aligned}$$

3.3.7. Show that the scheme derived in Example 3.3.2 is stable for $|a\lambda| \leq 3$.

3.3.8. Use the relationship $\partial = h^{-1}\ln(1 + h\delta_+)$ from (3.3.9) to derive the second-order accurate one-sided approximation

$$\frac{du}{dx}(x_0) \approx \frac{-3u(x_0) + 4u(x_1) - u(x_2)}{2h}.$$

3.4 Boundary Conditions for Finite Difference Schemes

In solving initial-boundary value problems such as (1.2.1) by finite difference schemes, we must use the boundary conditions required by the partial differential equation in order to determine the finite difference solution. Many schemes also require additional boundary conditions, called *numerical boundary conditions*, to determine the solution uniquely. We introduce our study of numerical boundary conditions by considering the Lax–Wendroff scheme applied to the initial-boundary value problem (1.2.1). In Chapter 11 we discuss the theory of boundary conditions in more detail.

When we use the Lax–Wendroff scheme on equation (1.2.1), the scheme can be applied only at the interior grid points and not at the boundary points. This is because the scheme requires grid points to the left and right of (t_n, x_m) when computing v_m^{n+1}, and at the boundaries either x_{m-1} or x_{m+1} is not a grid point. Assuming that a is positive, the value of v_0^n is supplied by the boundary data as required by the differential equation. At x_M, where x_M is the last grid point on the right, we must use some means other than the scheme to compute v_M^{n+1}. This additional condition is called a numerical boundary condition. Numerical boundary conditions should be some form of extrapolation that determines the solution on the boundary in terms of the solution in the interior. For

example, each of the following are numerical boundary conditions for (1.2.1):

$$v_M^{n+1} = v_{M-1}^{n+1}, \tag{3.4.1a}$$

$$v_M^{n+1} = 2v_{M-1}^{n+1} - v_{M-2}^{n+1}, \tag{3.4.1b}$$

$$v_M^{n+1} = v_{M-1}^{n}, \tag{3.4.1c}$$

$$v_M^{n+1} = 2v_{M-1}^{n} - v_{M-2}^{n-1}. \tag{3.4.1d}$$

Formulas (3.4.1a) and (3.4.1b) are simple extrapolations of the solution at interior grid points to the boundary. Formulas (3.4.1c) and (3.4.1d) are sometimes called *quasi-characteristic extrapolation,* since the extrapolation is done from points near the characteristics.

Numerical boundary conditions often take the form of one-sided differences of the partial differential equation. For example, rather than formulas (3.4.1) we might use

$$v_M^{n+1} = v_M^n - a\lambda(v_M^n - v_{M-1}^n). \tag{3.4.2}$$

However, we can easily see that (3.4.2) is the result of using the Lax–Wendroff scheme at v_M^{n+1} where v_{M+1}^n is determined by

$$v_{M+1}^n = 2v_M^n - v_{M-1}^n,$$

which is essentially (3.4.1b). This example also illustrates the use of extra points beyond the boundary to aid in the determination of the boundary values.

It is often easier to use extrapolation formulas such as (3.4.1) than to use extra points or one-sided differences. Moreover, the extrapolations can give as accurate answers as the other methods. The one-sided differences and extra points are occasionally justified by ad hoc physical arguments, which can be more confusing than useful.

There is one difficulty with numerical boundary conditions, which we do not have space to discuss in detail in this chapter, namely, that the numerical boundary condition coupled with a particular scheme can be unstable. This topic is discussed further in Chapter 11. For example, (3.4.1a) and (3.4.1b) together with the leapfrog scheme are unstable, whereas (3.4.1c) and (3.4.1d) are stable. For the Crank–Nicolson scheme, conditions (3.4.1c) and (3.4.1d) are unstable when $a\lambda$ is larger than 2, but (3.4.1a) and (3.4.1b) are stable. The proofs that these boundary conditions are stable or unstable, as the case may be, are given in Chapter 11.

The analysis of the stability of a problem involving both initial data and boundary conditions is done by considering the several parts. First, the scheme must be stable for the initial value problem considered on an unbounded domain. This is done with von Neumann analysis. The stability of the boundary conditions is done for each boundary separately. Conditions at one boundary cannot have a significantly ameliorating effect on an unstable boundary condition at the other boundary. As the preceding examples show, a boundary condition may be stable or unstable depending on the scheme with which it is being used.

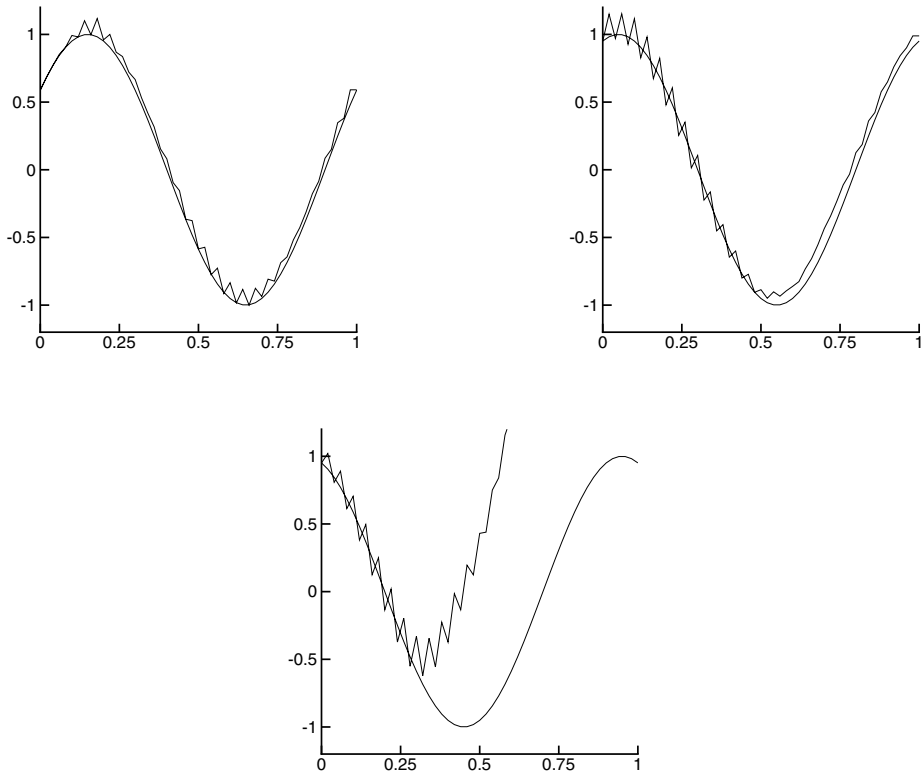

Figure 3.3. *Unstable boundary condition for the leapfrog scheme.*

Example 3.4.1. An example of an unstable boundary condition is shown in Figure 3.3. The leapfrog scheme is used with equation (1.2.1), with a equal to 1. The grid spacing is 0.02 and λ is equal to 0.9. At the left boundary, where x equals 0, u is specified to be the exact solution $\sin 2\pi(x - t)$. The Lax–Friedrichs scheme is used for the first time step. At the right boundary, where x is 1, (3.4.1a) is used. The three plots in the figure show the effect at the times 0.9, 1.8, and 2.7. The growth arising from an unstable boundary condition is not as dramatic as that arising from using an unstable scheme. The growth may be $O(n)$ for an unstable boundary condition, whereas it is exponential in n for an unstable scheme.

Figure 3.3 illustrates one additional difficulty with unstable boundary conditions: that the oscillations that are the result of the instability may not stay in the vicinity of the boundary. In the first plot the oscillations are spread throughout the interval, and in the plot at time 1.8, in the upper right, the oscillations are concentrated near the other boundary. This is due to the slow growth of the instability and the presence of the parasitic mode for the leapfrog scheme that propagates errors in the opposite direction from the differential equation. Parasitic modes are discussed in Chapters 4 and 5. After sufficient time, as shown

in the third plot, the effect of the boundary instability is seen at that boundary. When the effects of the boundary instability are observed far from the boundary, it can be difficult for programmers to determine that the boundary condition is the source of the oscillations. □

In practice, if we suspect that there is a numerical boundary condition instability, the easiest thing to do is to change to a different form of extrapolation to eliminate it. There is an analytical means of checking for these instabilities, but the algebraic manipulations are often quite involved, as will be seen in Chapter 11. If a computer program using a finite difference scheme is being used to solve a system of equations, it is usually easier to implement other boundary conditions than it is to analyze the original conditions to determine their stability.

One final comment should be made on this topic. In solving initial-boundary value problems by finite differences, it is best to distinguish clearly between those boundary conditions required by the partial differential equation and the numerical boundary conditions. By making this distinction, we can avoid solving overdetermined or underdetermined partial differential equation initial-boundary value problems.

Exercise

3.4.1. Solve the initial-boundary value problem (1.2.1) with the leapfrog scheme and the following boundary conditions. Use $a = 1$. Only (d) should give good results. Why?

(a) At $x = 0$, specify $u(t, 0)$; at $x = 1$, use boundary condition (3.4.1b).
(b) At $x = 0$, specify $u(t, 0)$; at $x = 1$, specify $u(t, 1) = 0$.
(c) At $x = 0$, use boundary condition (3.4.1b); at $x = 1$, use (3.4.1c).
(d) At $x = 0$, specify $u(t, 0)$; at $x = 1$, use boundary condition (3.4.1c).

3.5 Solving Tridiagonal Systems

To use the Crank–Nicolson scheme and many other implicit schemes such as (1.6.1), we must know how to solve tridiagonal systems of linear equations. We now present a convenient algorithm, called the *Thomas algorithm,* to solve tridiagonal systems that arise in finite difference schemes. This is the algorithm used to compute the solutions displayed in Figure 3.2.

Consider the system of equations

$$a_i w_{i-1} + b_i w_i + c_i w_{i+1} = d_i, \quad i = 1, \ldots, m-1, \tag{3.5.1}$$

with the boundary conditions

$$w_0 = \beta_0 \quad \text{and} \quad w_m = \beta_m. \tag{3.5.2}$$

We will solve this system by Gaussian elimination without partial pivoting. It reduces to this: We want to replace (3.5.1) by relationships of the form

$$w_i = p_{i+1} w_{i+1} + q_{i+1}, \quad i = 0, 1, 2, \ldots, m-1, \tag{3.5.3}$$

where the values of p_{i+1} and q_{i+1} are to be determined. For (3.5.3) to be consistent with (3.5.1), we substitute (3.5.3) into (3.5.1) for w_{i-1} and examine the resulting relation between w_i and w_{i+1}:

$$a_i(p_i w_i + q_i) + b_i w_i + c_i w_{i+1} = d_i$$

or

$$w_i = -(a_i p_i + b_i)^{-1} c_i w_{i+1} + (a_i p_i + b_i)^{-1} (d_i - a_i q_i).$$

Comparing this expression with (3.5.3) we must have

$$\begin{aligned} p_{i+1} &= -(a_i p_i + b_i)^{-1} c_i, \\ q_{i+1} &= (a_i p_i + b_i)^{-1} (d_i - a_i q_i), \end{aligned} \tag{3.5.4}$$

for consistency of the formulas. Thus if we know p_1 and q_1, then we can use (3.5.4) to compute p_i and q_i for i greater than 1. The values of p_1 and q_1 are obtained from the boundary condition (3.5.2) at $i = 0$. At i equal to 0 we have the two formulas $w_0 = p_1 w_1 + q_1$ and $w_0 = \beta_0$. These conditions are consistent if $p_1 = 0$ and $q_1 = \beta_0$. With these initial values for p_1 and q_1, formulas (3.5.4) then give all the values of p_i and q_i up to i equal to m. To get the values of w_i we use (3.5.3) starting with w_m, which is given.

We now consider other boundary conditions. If we have

$$w_0 = w_1 + \beta_0,$$

then we set $p_1 = 1$ and $q_1 = \beta_0$. If we have the boundary conditions

$$w_m = w_{m-1} + \beta_m,$$

then the relation

$$w_{m-1} = p_m w_m + q_m$$

also holds, and we combine these two relations to obtain

$$w_m = (1 - p_m)^{-1} (q_m + \beta_m).$$

If $p_m = 1$, then w_m cannot be defined, and the system with this boundary condition is singular.

In general, the values of p_1 and q_1 are determined by the boundary condition at i equal to 0, and the value of w_m is determined by the boundary condition at i equal to m, together with the relation (3.5.3) if necessary.

For the Thomas algorithm to be well-conditioned, we should have

$$|p_i| \leq 1. \tag{3.5.5}$$

This is equivalent to having the multipliers in Gaussian elimination be at most 1 in magnitude. From (3.5.3) we see that the error in w_{i+1} is multiplied by p_{i+1} to contribute to

the error in w_i. If (3.5.5) is violated for several values of i, then there will be an increase in the error. This error growth is due to ill-conditioning in the Thomas algorithm, and using Gaussian elimination with partial pivoting should remove this error magnification. Condition (3.5.5) has nothing to do with the stability or instability of the scheme.

The condition (3.5.5) should be checked when using the Thomas algorithm. Here are two special cases where (3.5.5) holds.

1. Diagonal dominance, i.e., $|a_i| + |c_i| \le |b_i|$.
2. $0 \le -c_i \le b_i,\ 0 \le a_i$ with $0 \le p_1 \le 1$, or $0 \le -a_i,\ 0 \le c_i \le b_i$ with $-1 \le p_1 \le 0$.

The formulas for tridiagonal systems can be extended to block tridiagonal systems in which the a_i, b_i, and c_i are square matrices and the unknown w_i are vectors. In this case the p_i are also matrices and the q_i are vectors. The method also extends to pentadiagonal systems.

Here is a sample of pseudocode for the Thomas algorithm for the Crank–Nicolson scheme. The function `Data` refers to the boundary data that must be supplied as part of the scheme. The boundary condition at the right end of the grid is (3.4.1c). This code must be included in a loop over all time steps.

```
# Set the parameters.
aa = -a*lambda/4
bb = 1
cc = -aa

# Set the first elements of the p and q arrays.
p(1) = 0.
q(1) = Data(time)
# Compute the p and q arrays recursively.
loop on m from 1 to M-1
   dd = v(m) - a*lambda*( v(m+1) - v(m-1))/2
   denom = (aa* p(m) + bb )
   p(m+1) = -cc / denom
   q(m+1) = (dd - q(m)*aa ) /denom
end of loop on m
# Apply the boundary condition at the last point.
v(M) = v(M-1)
# Compute all interior values.
loop on m from M-1 to 0
   v(m) = p(m+1)*v(m+1) + q(m+1)
end of loop on m
```

Periodic Tridiagonal Systems

If we use the Crank–Nicolson scheme or a similar scheme to solve a problem with periodic solutions, then we obtain periodic tridiagonal systems. These can be solved by an extension of the previous algorithm.

Consider the system

$$a_i w_{i-1} + b_i w_i + c_i w_{i+1} = d_i, \quad i = 1, \ldots, m, \tag{3.5.6}$$

with $w_0 = w_m$ and $w_{m+1} = w_1$. This periodic system can be solved as follows. Solve three systems as for the nonperiodic case, each for $i = 1, \ldots, m$:

$$a_i x_{i-1} + b_i x_i + c_i x_{i+1} = d_i$$

with $x_0 = 0$ and $x_{m+1} = 0$,

$$a_i y_{i-1} + b_i y_i + c_i y_{i+1} = 0$$

with $y_0 = 1$ and $y_{m+1} = 0$, and

$$a_i z_{i-1} + b_i z_i + c_i z_{i+1} = 0$$

with $z_0 = 0$ and $z_{m+1} = 1$.

Since these systems have the same matrix but different data, they use the same p_i's but different q_i's. (For the last of these systems, $q_i = 0$.)

Then we construct w_i as

$$w_i = x_i + r y_i + s z_i.$$

It is easy to see that w_i satisfies (3.5.6) for $i = 1, \ldots, m$. We choose r and s to guarantee the periodicity. The relationship $w_0 = w_m$ becomes

$$r = r y_0 = x_m + r y_m + s z_m$$

and $w_{m+1} = w_1$ becomes

$$s = s z_{m+1} = x_1 + r y_1 + s z_1.$$

These are two equations in the two unknowns r and s. The solution is

$$r = \frac{x_m(1 - z_1) + x_1 z_m}{D},$$

$$s = \frac{x_m y_1 + x_1(1 - y_m)}{D},$$

with

$$D = (1 - y_m)(1 - z_1) - y_1 z_m.$$

These formulas for solving periodic tridiagonal systems as well as the formula in Exercise 3.5.8 are special cases of the Sherman–Morrison formula for computing the inverse of a matrix given the inverse of a rank 1 modification of the matrix (see Exercise 3.5.10).

Exercises

3.5.1. Solve $u_t + u_x = 0$ on $-1 \le x \le 1$ for $0 \le t \le 1$ with the Crank–Nicolson scheme using the Thomas algorithm. For initial data and boundary data at x equal to -1, use the exact solution $u(t, x) = \sin \pi(x - t)$. Use $\lambda = 1.0$ and $h = 1/10, 1/20,$ and $1/40$. For the numerical boundary condition use

$$v_M^{n+1} - v_M^n + \lambda(v_M^{n+1} - v_{M-1}^{n+1}) = 0,$$

where $x_M = 1$. Comment on the accuracy of the method.

Note: When programming the method it is easiest to first debug your program using the boundary condition $v_M^{n+1} = v_{M-1}^n$. After you are sure the program works with this condition, you can then change to another boundary condition.

3.5.2. Solve $u_t + u_x + u = 0$ on $-1 \le x \le 1$ for $0 \le t \le 1$ with the Crank–Nicolson scheme using the Thomas algorithm. For initial data and boundary data at x equal to -1, use the two exact solutions:

(a) $u(t, x) = e^{-t} \sin \pi(x - t)$,
(b) $u(t, x) = \max(0, e^{-t} \cos \pi(x - t))$.

Use $\lambda = 1.0$ and $h = 1/10, 1/20,$ and $1/40$. Be sure that the undifferentiated term is treated accurately. For the numerical boundary condition use each of the following two methods:

$$\text{(a)} \quad v_M^{n+1} - v_M^n + \lambda(v_M^{n+1} - v_{M-1}^{n+1}) + k v_M^{n+1} = 0$$

and

$$\text{(b)} \quad v_M^{n+1} = 2v_{M-1}^{n+1} - v_{M-2}^{n+1},$$

where M is the grid index corresponding to x equal to 1. Comment on the accuracy of the methods. See the note in Exercise 3.5.1.

3.5.3. Solve $u_t + u_x - u = 0$ on $-1 \le x \le 1$ for $0 \le t \le 1$ with the Crank–Nicolson scheme using the Thomas algorithm. For initial data take

$$u(0, x) = \begin{cases} \cos^2 \pi x & \text{if } |x| \le 1/2, \\ 0 & \text{otherwise,} \end{cases}$$

and for boundary data take $u(t, -1) = 0$. Use $\lambda = 1.0$ and $h = 1/10, 1/20,$ and $1/40$. Be sure that the undifferentiated term is treated accurately. For the numerical boundary condition use each of the two methods

$$\text{(a)} \qquad v_M^{n+1} - v_M^n + \lambda(v_M^{n+1} - v_{M-1}^{n+1}) - k v_M^{n+1} = 0$$

and

$$\text{(b)} \qquad v_M^{n+1} = 2v_{M-1}^{n+1} - v_{M-2}^{n+1}.$$

Comment on the accuracy of the methods. See the note in Exercise 3.5.1.

3.5.4. Solve $u_t + u_x - u = -t \sin \pi(x - t)$ on $-1 \le x \le 1$ for $0 \le t \le 1.2$ with the Crank–Nicolson scheme using the Thomas algorithm. For initial data and boundary data at $x = -1$ use the exact solution $u(t, x) = (1 + t) \sin \pi(x - t)$. Use $\lambda = 1.0$ and $h = 1/10, 1/20,$ and $1/40$. Be sure that the undifferentiated term is treated accurately. For the numerical boundary condition at $x_M = 1$ use

$$v_M^{n+1} - v_M^n + \lambda(v_M^{n+1} - v_{M-1}^{n+1}) - k v_M^{n+1} = k f(t_{n+1}, x_M).$$

Comment on the accuracy of the method. See the note in Exercise 3.5.1.

3.5.5. Show that the condition (3.5.5) is violated for the Crank–Nicolson scheme (3.1.4) when $p_1 = 0$ and $a\lambda > 4$.

3.5.6. Show that the second-order differential equation

$$a(x)\frac{d^2u}{dx^2} + b(x)\frac{du}{dx} + c(x)u = d(x)$$

for $\alpha \le x \le \beta$ with $u(\alpha) = A$ and $u(\beta) = B$ can be solved by an algorithm similar to the Thomas algorithm. Set

$$\frac{du}{dx} = p(x)u + q(x)$$

and determine equations for $p(x)$ and $q(x)$. Discuss how $p \ge 0$ is the analogue to (3.5.5).

3.5.7. Repeat some of the calculations of Exercise 3.5.2 with the $(2, 4)$ accurate scheme of Exercise 3.3.6, modified to include the undifferentiated term. Can you attain a benefit from the fourth-order accuracy?

3.5.8. Show that the following algorithm also solves the periodic tridiagonal system (3.5.6).

1. Solve $a_i x_{i-1} + b_i x_i + c_i x_{i+1} = d_i, \quad i = 1, \ldots, m,$ with $x_0 = \sigma x_1$ and $x_{m+1} = \sigma x_m,$ where $\sigma = \text{sign}(a_1 b_1)$.
2. Solve $a_i y_{i-1} + b_i y_i + c_i y_{i+1} = 0, \quad i = 1, \ldots, m,$ with $y_0 = \sigma y_1 + 1$ and $y_m = \sigma y_{m+1} + 1$.
3. The solution w_i is then obtained as $w_i = x_i - r y_i,$ where $r = \frac{x_0 - x_m}{y_0 - y_m}$.

3.5.9. In the algorithm of Exercise 3.5.8, why shouldn't we take $\sigma = 1$ when a_1 and b_1 have opposite signs?

3.5.10. Verify the following formula, called the Sherman–Morrison formula, for a linear system of equations with matrix A.

If $Ay = b$ and $Az = u,$ then $(A + uv^T)x = b$ has the solution

$$x = y - \frac{v^T y}{(1 + v^T z)} z.$$

This formula is useful for computing the solution x of $(A + uv^T)x = b$ if we have a convenient method of solving equations of the form $Ay = b$.