# Numerical Analysis Lecture Notes

Peter J. Olver

# 11. Finite Difference Methods for Partial Differential Equations

As you are well aware, most differential equations are much too complicated to be solved by an explicit analytic formula. Thus, the development of accurate numerical approximation schemes is essential for both extracting quantitative information as well as achieving a qualitative understanding of the behavior of their solutions. Even in cases, such as the heat and wave equations, where explicit solution formulas (either closed form or infinite series) exist, numerical methods still can be profitably employed. Indeed, the lessons learned in the design of numerical algorithms for "solved" examples are of inestimable value when confronting more challenging problems. Furthermore, one has the ability to accurately test a proposed numerical algorithm by running it on a known solution.

Basic numerical solution schemes for partial differential equations fall into two broad categories. The first are the *finite difference methods*, obtained by replacing the derivatives in the equation by the appropriate numerical differentiation formulae. However, there is no guarantee that the resulting numerical scheme will accurately approximate the true solution, and further analysis is required to elicit bona fide, convergent numerical algorithms. We thus start with a brief discussion of simple finite difference formulae for numerically approximating low order derivatives of functions. The ensuing sections establish some of the most basic finite difference schemes for the heat equation, first order transport equations, and the second order wave equation. As we will see, not all finite difference approximations lead to accurate numerical schemes, and the issues of stability and convergence must be dealt with in order to distinguish valid from worthless methods. In fact, inspired by Fourier analysis, the basic stability criterion for a finite difference scheme is based on how the scheme handles complex exponentials.

We will only introduce the most basic algorithms, leaving more sophisticated variations and extensions to a more thorough treatment, which can be found in numerical analysis texts, e.g., [**5**, **7**, **29**].

## 11.1. Finite Differences.

In general, to approximate the derivative of a function at a point, say $f'(x)$ or $f''(x)$, one constructs a suitable combination of sampled function values at nearby points. The underlying formalism used to construct these approximation formulae is known as the *calculus of finite differences*. Its development has a long and influential history, dating

One-Sided Difference Central Difference

**Figure 11.1.** Finite Difference Approximations.

back to Newton. The resulting *finite difference numerical methods* for solving differential equations have extremely broad applicability, and can, with proper care, be adapted to most problems that arise in mathematics and its many applications.

The simplest finite difference approximation is the ordinary *difference quotient*

$$\frac{u(x+h) - u(x)}{h} \approx u'(x), \tag{11.1}$$

used to approximate the first derivative of the function $u(x)$. Indeed, if $u$ is differentiable at $x$, then $u'(x)$ is, by definition, the limit, as $h \to 0$ of the finite difference quotients. Throughout our discussion, $h$, the *step size*, which may be either positive or negative, is assumed to be small: $|\, h \,| \ll 1$. When $h > 0$, (11.1) is referred to as a *forward difference*, while $h < 0$ gives a *backward difference*. Geometrically, the difference quotient equals the slope of the secant line through the two points $\big(x, u(x)\big)$ and $\big(x+h, u(x+h)\big)$ on the graph of the function. For small $h$, this should be a reasonably good approximation to the slope of the tangent line, $u'(x)$, as illustrated in the first picture in Figure 11.1.

How close an approximation is the difference quotient? To answer this question, we assume that $u(x)$ is at least twice continuously differentiable, and examine the first order Taylor expansion

$$u(x+h) = u(x) + u'(x)\, h + \tfrac{1}{2}\, u''(\xi)\, h^2. \tag{11.2}$$

We have used the Cauchy form for the remainder term, [**2**], in which $\xi$ represents some point lying between $x$ and $x+h$. The *error* or difference between the finite difference formula and the derivative being approximated is given by

$$\frac{u(x+h) - u(x)}{h} \;-\; u'(x) = \tfrac{1}{2}\, u''(\xi)\, h. \tag{11.3}$$

Since the error is proportional to $h$, we say that the finite difference quotient (11.3) is a *first order* approximation. When the precise formula for the error is not so important, we will write

$$u'(x) = \frac{u(x+h) - u(x)}{h} + \mathrm{O}(h). \tag{11.4}$$

The "big Oh" notation $\mathrm{O}(h)$ refers to a term that is proportional to $h$, or, more rigorously, bounded by a constant multiple of $h$ as $h \to 0$.

**Example 11.1.** Let $u(x) = \sin x$. Let us try to approximate $u'(1) = \cos 1 = .5403023\ldots$ by computing finite difference quotients

$$\cos 1 \approx \frac{\sin(1+h) - \sin 1}{h}.$$

The result for different values of $h$ is listed in the following table.

| $h$ | 1 | .1 | .01 | .001 | .0001 |
|---|---|---|---|---|---|
| approximation | .067826 | .497364 | .536086 | .539881 | .540260 |
| error | $-.472476$ | $-.042939$ | $-.004216$ | $-.000421$ | $-.000042$ |

We observe that reducing the step size by a factor of $\frac{1}{10}$ reduces the size of the error by approximately the same factor. Thus, to obtain 10 decimal digits of accuracy, we anticipate needing a step size of about $h = 10^{-11}$. The fact that the error is more of less proportional to the step size confirms that we are dealing with a first order numerical approximation.

To approximate higher order derivatives, we need to evaluate the function at more than two points. In general, an approximation to the $n^{\text{th}}$ order derivative $u^{(n)}(x)$ requires at least $n+1$ distinct sample points. For simplicity, we shall only use equally spaced points, leaving the general case to the exercises.

For example, let us try to approximate $u''(x)$ by sampling $u$ at the particular points $x$, $x+h$ and $x-h$. Which combination of the function values $u(x-h), u(x), u(x+h)$ should be used? The answer to such a question can be found by consideration of the relevant Taylor expansions

$$\begin{aligned} u(x+h) &= u(x) + u'(x)\,h + u''(x)\,\frac{h^2}{2} + u'''(x)\,\frac{h^3}{6} + \mathrm{O}(h^4), \\ u(x-h) &= u(x) - u'(x)\,h + u''(x)\,\frac{h^2}{2} - u'''(x)\,\frac{h^3}{6} + \mathrm{O}(h^4), \end{aligned} \tag{11.5}$$

where the error terms are proportional to $h^4$. Adding the two formulae together gives

$$u(x+h) + u(x-h) = 2\,u(x) + u''(x)\,h^2 + \mathrm{O}(h^4).$$

Rearranging terms, we conclude that

$$u''(x) = \frac{u(x+h) - 2\,u(x) + u(x-h)}{h^2} + \mathrm{O}(h^2), \tag{11.6}$$

The result is known as the *centered finite difference approximation* to the second derivative of a function. Since the error is proportional to $h^2$, this is a second order approximation.

**Example 11.2.** Let $u(x) = e^{x^2}$, with $u''(x) = (4x^2 + 2)\,e^{x^2}$. Let us approximate $u''(1) = 6e = 16.30969097\ \ldots$ by using the finite difference quotient (11.6):

$$6e \approx \frac{e^{(1+h)^2} - 2e + e^{(1-h)^2}}{h^2}.$$

The results are listed in the following table.

| $h$ | 1 | .1 | .01 | .001 | .0001 |
|---|---|---|---|---|---|
| approximation | 5.16158638 | 16.48289823 | 16.31141265 | 16.30970819 | 16.30969115 |
| error | 33.85189541 | .17320726 | .00172168 | .00001722 | .00000018 |

Each reduction in step size by a factor of $\frac{1}{10}$ reduces the size of the error by a factor of $\frac{1}{100}$ and results in a gain of two new decimal digits of accuracy, confirming that the finite difference approximation is of second order.

However, this prediction is not completely borne out in practice. If we take[†] $h = .00001$ then the formula produces the approximation 16.3097002570, with an error of .0000092863 — which is *less* accurate that the approximation with $h = .0001$. The problem is that round-off errors have now begun to affect the computation, and underscores the difficulty with numerical differentiation. Finite difference formulae involve dividing very small quantities, which can induce large numerical errors due to round-off. As a result, while they typically produce reasonably good approximations to the derivatives for moderately small step sizes, to achieve high accuracy, one must switch to a higher precision. In fact, a similar comment applied to the previous Example 11.1, and our expectations about the error were not, in fact, fully justified as you may have discovered if you tried an extremely small step size.

Another way to improve the order of accuracy of finite difference approximations is to employ more sample points. For instance, if the first order approximation (11.4) to the first derivative based on the two points $x$ and $x+h$ is not sufficiently accurate, one can try combining the function values at three points $x$, $x+h$ and $x-h$. To find the appropriate combination of $u(x-h), u(x), u(x+h)$, we return to the Taylor expansions (11.5). To solve for $u'(x)$, we subtract[‡] the two formulae, and so

$$u(x+h) - u(x-h) = 2\,u'(x)\,h + u'''(x)\,\frac{h^3}{3} + \mathrm{O}(h^4).$$

Rearranging the terms, we are led to the well-known *centered difference formula*

$$u'(x) = \frac{u(x+h) - u(x-h)}{2\,h} + \mathrm{O}(h^2), \tag{11.7}$$

which is a second order approximation to the first derivative. Geometrically, the centered difference quotient represents the slope of the secant line through the two points $\big(x-h, u(x-h)\big)$ and $\big(x+h, u(x+h)\big)$ on the graph of $u$ centered symmetrically about the point $x$. Figure 11.1 illustrates the two approximations; the advantages in accuracy in the centered difference version are graphically evident. Higher order approximations can be found by evaluating the function at yet more sample points, including, say, $x+2\,h, x-2\,h$, etc.

[†] This next computation depends upon the computer's precision; here we used single precision in MATLAB.

[‡] The terms $\mathrm{O}(h^4)$ do *not* cancel, since they represent potentially different multiples of $h^4$.

**Example 11.3.** Return to the function $u(x) = \sin x$ considered in Example 11.1. The centered difference approximation to its derivative $u'(1) = \cos 1 = .5403023\ \ldots$ is

$$\cos 1 \approx \frac{\sin(1+h) - \sin(1-h)}{2\,h}.$$

The results are tabulated as follows:

| $h$ | .1 | .01 | .001 | .0001 |
|---|---|---|---|---|
| approximation | .53940225217 | .54029330087 | .54030221582 | .54030230497 |
| error | −.00090005370 | −.00000900499 | −.00000009005 | −.00000000090 |

As advertised, the results are much more accurate than the one-sided finite difference approximation used in Example 11.1 at the same step size. Since it is a second order approximation, each reduction in the step size by a factor of $\frac{1}{10}$ results in two more decimal places of accuracy.

Many additional finite difference approximations can be constructed by similar manipulations of Taylor expansions, but these few very basic ones will suffice for our subsequent purposes. In the following subsection, we apply the finite difference formulae to develop numerical solution schemes for the heat and wave equations.

## 11.2. Numerical Algorithms for the Heat Equation.

Consider the heat equation

$$\frac{\partial u}{\partial t} = \gamma\, \frac{\partial^2 u}{\partial x^2}\,, \qquad 0 < x < \ell, \qquad t \geq 0, \tag{11.8}$$

representing a bar of length $\ell$ and constant thermal diffusivity $\gamma > 0$. To be concrete, we impose time-dependent Dirichlet boundary conditions

$$u(t,0) = \alpha(t), \qquad u(t,\ell) = \beta(t), \qquad t \geq 0, \tag{11.9}$$

specifying the temperature at the ends of the bar, along with the initial conditions

$$u(0,x) = f(x), \qquad 0 \leq x \leq \ell, \tag{11.10}$$

specifying the bar's initial temperature distribution. In order to effect a numerical approximation to the solution to this initial-boundary value problem, we begin by introducing a *rectangular mesh* consisting of points $(t_i, x_j)$ with

$$0 = t_0 < t_1 < t_2 < \cdots \qquad \text{and} \qquad 0 = x_0 < x_1 < \cdots < x_n = \ell.$$

For simplicity, we maintain a uniform mesh spacing in both directions, with

$$\Delta t = t_{i+1} - t_i, \qquad \Delta x = x_{j+1} - x_j = \frac{\ell}{n}\,,$$

representing, respectively, the time step size and the spatial mesh size. It will be essential that we do *not* a priori require the two to be the same. We shall use the notation

$$u_{i,j} \approx u(t_i, x_j) \qquad \text{where} \qquad t_i = i\,\Delta t, \qquad x_j = j\,\Delta x, \tag{11.11}$$

to denote the numerical approximation to the solution value at the indicated mesh point.

As a first attempt at designing a numerical method, we shall employ the simplest finite difference approximations to the derivatives. The second order space derivative is approximated by (11.6), and hence

$$\begin{aligned} \frac{\partial^2 u}{\partial x^2}\,(t_i, x_j) &\approx \frac{u(t_i, x_{j+1}) - 2\,u(t_i, x_j) + u(t_i, x_{j-1})}{(\Delta x)^2} + \mathrm{O}\big((\Delta x)^2\big) \\ &\approx \frac{u_{i,j+1} - 2\,u_{i,j} + u_{i,j-1}}{(\Delta x)^2} + \mathrm{O}\big((\Delta x)^2\big), \end{aligned} \tag{11.12}$$

where the error in the approximation is proportional to $(\Delta x)^2$. Similarly, the one-sided finite difference approximation (11.4) is used to approximate the time derivative, and so

$$\frac{\partial u}{\partial t}\,(t_i, x_j) \approx \frac{u(t_{i+1}, x_j) - u(t_i, x_j)}{\Delta t} + \mathrm{O}(\Delta t) \approx \frac{u_{i+1,j} - u_{i,j}}{\Delta t} + \mathrm{O}(\Delta t), \tag{11.13}$$

where the error is proportion to $\Delta t$. In practice, one should try to ensure that the approximations have similar orders of accuracy, which leads us to choose

$$\Delta t \approx (\Delta x)^2.$$

Assuming $\Delta x < 1$, this requirement has the important consequence that the time steps must be *much* smaller than the space mesh size.

*Remark*: At this stage, the reader might be tempted to replace (11.13) by the second order central difference approximation (11.7). However, this produces significant complications, and the resulting numerical scheme is not practical; see Exercise 11.2.11.

Replacing the derivatives in the heat equation (11.14) by their finite difference approximations (11.12), (11.13), and rearranging terms, we end up with the linear system

$$u_{i+1,j} = \mu\, u_{i,j+1} + (1 - 2\,\mu) u_{i,j} + \mu\, u_{i,j-1}, \qquad \begin{array}{l} i = 0, 1, 2, \dots, \\ j = 1, \dots, n-1, \end{array} \tag{11.14}$$

in which

$$\mu = \frac{\gamma\, \Delta t}{(\Delta x)^2}\,. \tag{11.15}$$

The resulting numerical scheme takes the form of an iterative linear system, in which, starting at the initial time $t_0$, the solution values $u_{i+1,j} \approx u(t_{i+1}, x_j)$ at time $t_{i+1}$ are calculated from those at the preceding time $t_i$.

The initial condition (11.10) means that we should initialize our numerical data by sampling the initial temperature at the mesh points:

$$u_{0,j} = f_j = f(x_j), \qquad j = 1, \dots, n-1. \tag{11.16}$$

Similarly, the boundary conditions (11.9) require that

$$u_{i,0} = \alpha_i = \alpha(t_i), \qquad u_{i,n} = \beta_i = \beta(t_i), \qquad i = 0, 1, 2, \dots\,. \tag{11.17}$$

For consistency, we should assume that the initial and boundary conditions agree at the corners of the domain:

$$f_0 = f(0) = u(0,0) = \alpha(0) = \alpha_0, \qquad f_n = f(\ell) = u(0,\ell) = \beta(0) = \beta_0.$$

The three equations (11.14–17) completely prescribe the numerical approximation algorithm for solving the initial-boundary value problem (11.8–10).

Let us rewrite the scheme in a more transparent matrix form. First, let

$$\mathbf{u}^{(i)} = \big( u_{i,1}, u_{i,2}, \dots, u_{i,n-1} \big)^T \approx \big( u(t_i, x_1), u(t_i, x_2), \dots, u(t_i, x_{n-1}) \big)^T \tag{11.18}$$

be the vector whose entries are the numerical approximations to the solution values at time $t_i$ at the *interior* nodes. We omit the boundary nodes $x_0 = 0$, $x_n = \ell$, since those values are fixed by the boundary conditions (11.9). Then (11.14) has the form

$$\mathbf{u}^{(i+1)} = A\,\mathbf{u}^{(i)} + \mathbf{b}^{(i)}, \tag{11.19}$$

where

$$A = \begin{pmatrix} 1-2\,\mu & \mu & & & & & \\ \mu & 1-2\,\mu & \mu & & & & \\ & \mu & 1-2\,\mu & \mu & & & \\ & & \mu & \ddots & \ddots & & \\ & & & \ddots & \ddots & \mu & \\ & & & & \mu & 1-2\,\mu \end{pmatrix}, \qquad \mathbf{b}^{(i)} = \begin{pmatrix} \mu\,\alpha_i \\ 0 \\ 0 \\ \vdots \\ 0 \\ \mu\,\beta_i \end{pmatrix}. \tag{11.20}$$

The coefficient matrix $A$ is symmetric and tridiagonal. The contributions (11.17) of the boundary nodes appear in the vector $\mathbf{b}^{(i)}$. This numerical method is known as an *explicit scheme* since each iterate is computed directly without relying on solving an auxiliary equation — unlike the implicit schemes to be discussed below.

**Example 11.4.** Let us fix the diffusivity $\gamma = 1$ and the bar length $\ell = 1$. For illustrative purposes, we take a spatial step size of $\Delta x = .1$. In Figure 11.2 we compare two (slightly) different time step sizes on the initial data

$$u(0,x) = f(x) = \begin{cases} -x, & 0 \le x \le \frac{1}{5}, \\ x - \frac{2}{5}, & \frac{1}{5} \le x \le \frac{7}{10}, \\ 1 - x, & \frac{7}{10} \le x \le 1. \end{cases} \tag{11.21}$$

The first sequence uses the time step $\Delta t = (\Delta x)^2 = .01$ and plots the solution at times $t = 0., .02, .04$. The numerical solution is already showing signs of instability, and indeed soon thereafter becomes completely wild. The second sequence takes $\Delta t = .005$ and plots the solution at times $t = 0., .025, .05$. (Note that the two sequences of plots have different vertical scales.) Even though we are employing a rather coarse mesh, the numerical solution is not too far away from the true solution to the initial value problem.

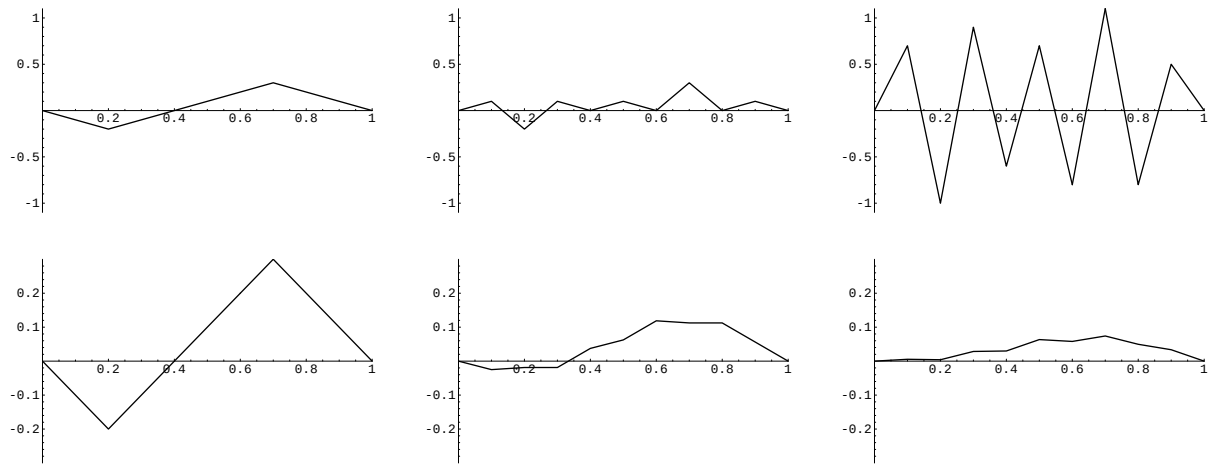


**Figure 11.2.** Numerical Solutions for the Heat Equation Based on the Explicit Scheme.

In light of this calculation, we need to understand why our scheme sometimes gives reasonable answers but at other times utterly fails. To this end, we investigate the effect of the numerical scheme on simple functions. As we know, the general solution to the heat equation can be decomposed into a sum over the various Fourier modes. Thus, we can concentrate on understanding what the numerical scheme does to an individual complex exponential† function, bearing in mind that we can then reconstruct the cumulative effect by taking suitable linear combinations.

Suppose that, at a time $t = t_i$, the solution is a pure exponential

$$u(t_i, x) = e^{\mathrm{i}\, k x}, \qquad \text{and so} \qquad u_{i,j} = u(t_i, x_j) = e^{\mathrm{i}\, k x_j}. \tag{11.22}$$

Substituting these values into our numerical equations (11.14), we find the updated value at time $t_{i+1}$ is also an exponential:

$$\begin{aligned} u_{i+1,j} &= \mu\, u_{i,j+1} + (1 - 2\mu) u_{i,j} + \mu\, u_{i,j-1} = \mu\, e^{\mathrm{i}\, k x_{j+1}} + (1 - 2\mu) e^{\mathrm{i}\, k x_j} + \mu\, e^{\mathrm{i}\, k x_{j-1}} \\ &= \mu\, e^{\mathrm{i}\, k (x_j + \Delta x)} + (1 - 2\mu) e^{\mathrm{i}\, k x_j} + \mu\, e^{\mathrm{i}\, k (x_j - \Delta x)} = \lambda\, e^{\mathrm{i}\, k x_j}, \end{aligned} \tag{11.23}$$

where

$$\lambda = \mu\, e^{\mathrm{i}\, k \Delta x} + (1 - 2\mu) + \mu\, e^{-\mathrm{i}\, k \Delta x} = 1 - 2\mu(1 - \cos k \Delta x) = 1 - 4\mu \sin^2 \tfrac{1}{2} k \Delta x. \tag{11.24}$$

Thus, the effect of a single step of the numerical scheme is to multiply the complex exponential (11.22) by the so-called *magnification factor* $\lambda$:

$$u(t_{i+1}, x) = \lambda\, e^{\mathrm{i}\, k x}, \qquad \text{and so} \qquad u_{i+1,j} = u(t_{i+1}, x_j) = \lambda\, e^{\mathrm{i}\, k x_j}. \tag{11.25}$$

† As usual, complex exponentials are much easier to work with than real trigonometric functions.

In other words, $e^{\,\mathrm{i}\,k\,x}$ assumes the role of an *eigenfunction*, with the magnification factor $\lambda$ being the corresponding *eigenvalue*, of the linear operator governing each step of the numerical scheme. Continuing, we find that the effect of $n$ further iterations of the scheme is to multiply the exponential by the $n^{\text{th}}$ power of the magnification factor:

$$u(t_{i+n}, x) = \lambda^n e^{\,\mathrm{i}\,k\,x}. \tag{11.26}$$

Thus, the stability of the scheme will be governed by the size of the magnification factor. If $|\,\lambda\,| > 1$, then $\lambda^n$ is exponentially growing as $n \to \infty$, and so the numerical solutions (11.26) become unbounded as $t \to \infty$. This is clearly incompatible with the analytical behavior of solutions to the heat equation, and so a necessary condition for the stability of our numerical scheme is that its magnification factor satisfy

$$|\,\lambda\,| \le 1. \tag{11.27}$$

This method of stability analysis was developed by the mid-twentieth century Hungarian mathematician — and father of the electronic computer — John von Neumann. The stability criterion (11.27) effectively distinguishes the stable, and hence valid numerical algorithms from the unstable, and hence worthless schemes. For the particular case (11.24), the von Neumann stability criterion requires

$$\mu = \frac{\gamma\,\Delta t}{(\Delta x)^2} \le \frac{1}{2}\,, \qquad \text{or} \qquad \Delta t \le \frac{(\Delta x)^2}{2\,\gamma}\,, \tag{11.28}$$

and thus restricts the allowable time step size. For instance, if we have $\Delta x = .01$, and $\gamma = 1$, then we can only use a time step size $\Delta t \le .00005$, which is minuscule. It would take an inordinately large number of time steps to compute the value of the solution at even a moderate times, e.g., $t = 1$. Moreover, owing to the limited accuracy of computers, the propagation of round-off errors might then cause a significant reduction in the overall accuracy of the final solution values. Since not all choices of space and time steps lead to a convergent scheme, the explicit scheme (11.14) is called *conditionally stable*.

An unconditionally stable method — one that does not restrict the time step — can be constructed by using the backwards difference formula

$$\frac{\partial u}{\partial t}\,(t_i, x_j) \approx \frac{u(t_i, x_j) - u(t_{i-1}, x_j)}{\Delta t} + \mathrm{O}\big(\,(\Delta t)^2\,\big) \tag{11.29}$$

to approximate the time derivative. Substituting (11.29) and the same approximation (11.12) for $u_{xx}$ into the heat equation, and then replacing $i$ by $i+1$, leads to the iterative system

$$-\mu\,u_{i+1,j+1} + (1+2\,\mu)u_{i+1,j} - \mu\,u_{i+1,j-1} = u_{i,j}, \qquad \begin{array}{l} i = 0, 1, 2, \dots\,, \\ j = 1, \dots, n-1, \end{array} \tag{11.30}$$

where the parameter $\mu = \gamma\,\Delta t/(\Delta x)^2$ is as above. The initial and boundary conditions also have the same form (11.16, 17). The system can be written in the matrix form

$$\widehat{A}\,\mathbf{u}^{(i+1)} = \mathbf{u}^{(i)} + \mathbf{b}^{(i+1)}, \tag{11.31}$$

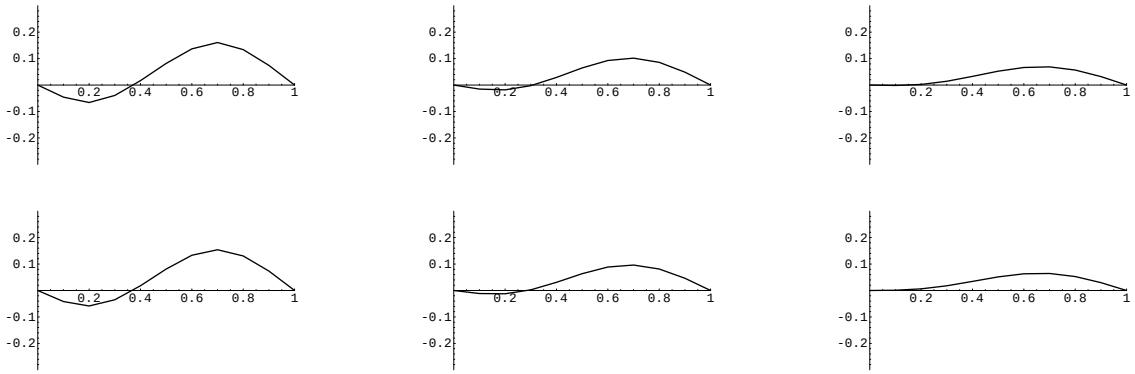



**Figure 11.3.** Numerical Solutions for the Heat Equation Based on the Implicit Scheme.

where $\widehat{A}$ is obtained from the matrix $A$ in (11.20) by replacing $\mu$ by $-\mu$. This serves to define an *implicit scheme*, since we have to solve a linear system of algebraic equations at each step in order to compute the next iterate $\mathbf{u}^{(i+1)}$. However, as the coefficient matrix $\widehat{A}$ is tridiagonal, the system can be solved very rapidly, [**42**], and so speed is not a significant issue in the practical implementation of this implicit scheme.

Let us apply the von Neumann analysis to investigate the stability of the implicit scheme. Again, we need only look at the effect of the scheme on a complex exponential. Substituting (11.22, 25) into (11.30) and canceling the common exponential factor leads to the equation

$$\lambda\big(-\mu e^{\mathrm{i}\,k\,\Delta x} + 1 + 2\,\mu - \mu e^{-\mathrm{i}\,k\,\Delta x}\big) = 1.$$

We solve for the magnification factor

$$\lambda = \frac{1}{1 + 2\,\mu\big(1 - \cos k\,\Delta x\big)} = \frac{1}{1 + 4\,\mu \sin^2 \frac{1}{2}\,k\,\Delta x}\,. \tag{11.32}$$

Since $\mu > 0$, the magnification factor *always* less than 1 in absolute value, and so the stability criterion (11.27) is satisfied *for any choice of step sizes*. We conclude that the implicit scheme (11.14) is *unconditionally stable*.

**Example 11.5.** Consider the same initial-boundary value problem considered in Example 11.4. In Figure 11.3, we plot the numerical solutions obtained using the implicit scheme. The initial data is not displayed, but we graph the numerical solutions at times $t = .2, .4, .6$ with a mesh size of $\Delta x = .1$. On the top line, we use a time step of $\Delta t = .01$, while on the bottom $\Delta t = .005$. Unlike the explicit scheme, there is very little difference between the two — both come much closer to the actual solution than the explicit scheme. Indeed, even significantly larger time steps give reasonable numerical approximations to the solution.

Another popular numerical scheme for solving the heat equation is the *Crank–Nicolson method*

$$u_{i+1,j} - u_{i,j} = \tfrac{1}{2}\,\mu\,(u_{i+1,j+1} - 2\,u_{i+1,j} + u_{i+1,j-1} + u_{i,j+1} - 2\,u_{i,j} + u_{i,j-1}). \tag{11.33}$$

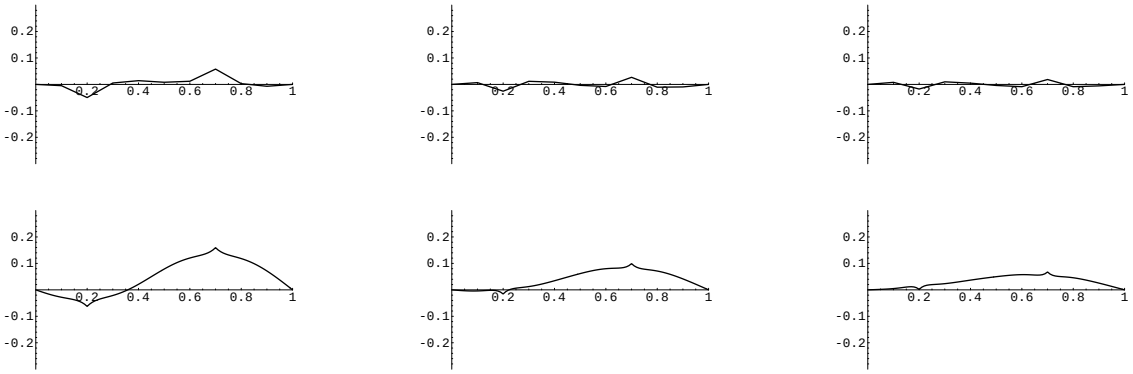


**Figure 11.4.** Numerical Solutions for the Heat Equation Based on the Crank–Nicolson Scheme.

which can be obtained by averaging the explicit and implicit schemes (11.14, 30). We can write (11.33) in matrix form

$$B\,\mathbf{u}^{(i+1)} = C\,\mathbf{u}^{(i)} + \tfrac{1}{2}\big(\,\mathbf{b}^{(i)} + \mathbf{b}^{(i+1)}\,\big),$$

where

$$B = \begin{pmatrix} 1+\mu & -\frac{1}{2}\mu & & \\ -\frac{1}{2}\mu & 1+\mu & -\frac{1}{2}\mu & \\ & -\frac{1}{2}\mu & \ddots & \ddots \\ & & \ddots & \ddots \end{pmatrix}, \qquad C = \begin{pmatrix} 1-\mu & \frac{1}{2}\mu & & \\ \frac{1}{2}\mu & 1-\mu & \frac{1}{2}\mu & \\ & \frac{1}{2}\mu & \ddots & \ddots \\ & & \ddots & \ddots \end{pmatrix}. \tag{11.34}$$

Applying the von Neumann analysis as before, we deduce that the magnification factor has the form

$$\lambda = \frac{1 - 2\,\mu \sin^2 \frac{1}{2}\,k\,\Delta x}{1 + 2\,\mu \sin^2 \frac{1}{2}\,k\,\Delta x}\,. \tag{11.35}$$

Since $\mu > 0$, then $|\,\lambda\,| \le 1$ for all choices of step size, and so the Crank–Nicolson scheme is also unconditionally stable. A detailed analysis will show that the errors are of the order of $(\Delta t)^2$ and $(\Delta x)^2$, and so it is reasonable to choose the time step to have the same order of magnitude as the space step, $\Delta t \approx \Delta x$. This gives the Crank–Nicolson scheme one advantage over the previous two methods. However, applying it to the initial value problem considered earlier points out a significant weakness. Figure 11.4 shows the result of running the scheme on the initial data (11.21). The top row has space and time step sizes $\Delta t = \Delta x = .1$, and does a rather poor job replicating the solution. The second row uses $\Delta t = \Delta x = .01$, and performs better except near the corners where an annoying and incorrect local time oscillation persists as the solution decays. Indeed, unlike the implicit scheme, the Crank–Nicolson method fails to rapidly damp out the high frequency modes associated with small scale features such as discontinuities and corners in the initial data. In such situations, a good strategy is to first evolve using the implicit scheme until the small scale noise is dissipated away, and then switch to Crank–Nicolson to use a much larger time step for the final large scale changes.

## 11.3. Numerical Solution Methods for First Order Partial Differential Equations.

We begin with the initial value problem for the elementary constant coefficient transport equation

$$\frac{\partial u}{\partial t} + c\,\frac{\partial u}{\partial x} = 0, \qquad u(0,x) = f(x). \tag{11.36}$$

The solution is a simple traveling wave

$$u(t,x) = f(x - ct), \tag{11.37}$$

that is constant along the characteristic lines of slope $c$ in the $t\,x$ plane. Although the analytical solution is completely elementary, there will be valuable lessons to be learned from an attempt to reproduce it by numerical approximation. Indeed, each of the numerical schemes developed below has an evident adaptation to transport equations with variable wave speeds $c(t,x)$, and even to nonlinear transport equations whose wave speed depends on the solution $u$ and can admit shock wave solutions.

As usual, we restrict our attention to a regular mesh $(t_i, x_j)$ with uniform time and space step sizes $\Delta t = t_{i+1} - t_i$ and $\Delta x = x_{j+1} - x_j$. We use $u_{i,j} \approx u(t_i, x_j)$ to denote our numerical approximation to the solution $u(t,x)$ at the indicated mesh point. The most elementary numerical solution scheme is obtained by replacing the time and space derivatives in the transport equation by their first order finite difference approximations (11.1):

$$\frac{\partial u}{\partial t}\,(t_i, x_j) \approx \frac{u_{i+1,j} - u_{i,j}}{\Delta t} + \mathrm{O}(\Delta t), \qquad \frac{\partial u}{\partial x}\,(t_i, x_j) \approx \frac{u_{i,j+1} - u_{i,j}}{\Delta x} + \mathrm{O}(\Delta x). \tag{11.38}$$

The result is the explicit numerical scheme

$$u_{i+1,j} = -\sigma\, u_{i,j+1} + (\sigma + 1) u_{i,j}, \tag{11.39}$$

in which the parameter

$$\sigma = \frac{c\,\Delta t}{\Delta x} \tag{11.40}$$

depends upon the wave speed and the ratio of space and time step sizes. Since we are employing first order approximations to both derivatives, we should choose the time and space step sizes to be of comparable size $\Delta t \approx \Delta x$. When working on a bounded interval, say $0 \le x \le \ell$, we will need to specify a value for the numerical solution at the left end, e.g., setting $u_{i,n} = 0$; this corresponds to imposing the boundary condition $u(t,\ell) = 0$.

In Figure 11.5, we plot the solutions arising from the following initial conditions

$$u(0,x) = f(x) = .4\, e^{-300(x-.5)^2} + .1\, e^{-300(x-.65)^2}, \tag{11.41}$$

at times $t = 0, .15$ and $.3$. We use step sizes $\Delta t = \Delta x = .005$, and try four different values of the wave speed. The cases $c = .5$ and $c = -1.5$ are clearly exhibiting some form of numerical instability. The numerical solution $c = .5$ is a bit more reasonable, although one can already observe some degradation due to the relatively low accuracy of the scheme.

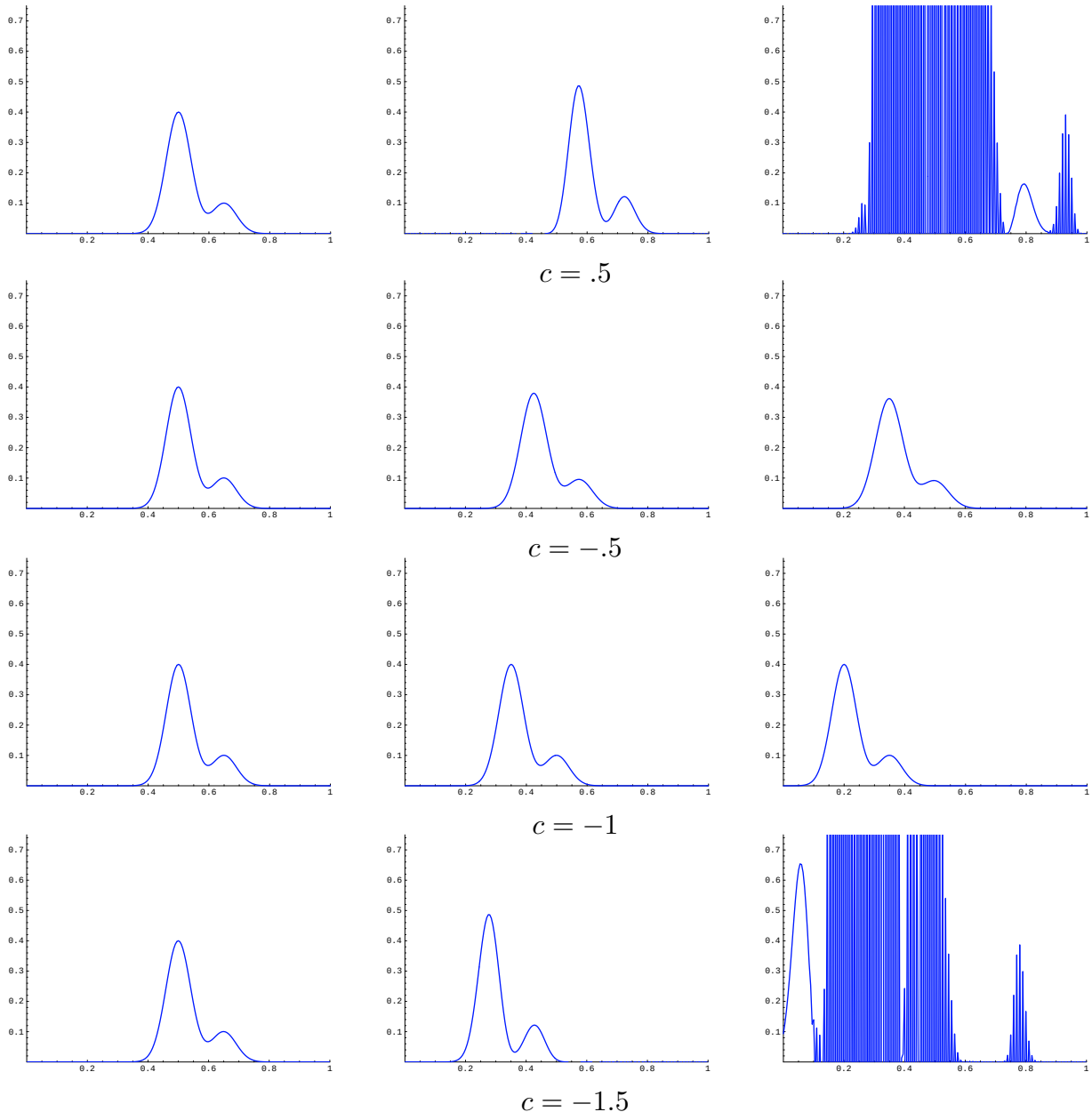



**Figure 11.5.** Numerical Solutions to the Transport Equation.

This can be overcome by selecting a smaller step size. The case $c = -1$ looks particularly good — but this is an accident. The analytic solution also happens to be an exact solution to the numerical equations, and so the only possible source of error is round-off.

The are two ways to understand the observed numerical instability. First, we recall the exact solution is constant along the characteristic lines $x = ct + \xi$, and hence the value of $u(t, x)$ only depends on the initial value $f(\xi)$ at the point $\xi = x - ct$. On the other hand, at time $t = t_n$, the numerical solution $u_{n,j} \approx u(t_n, x_j)$ computed using (11.39) depends on the values of $u_{n-1,j}$ and $u_{n-1,j+1}$. The latter two values have been

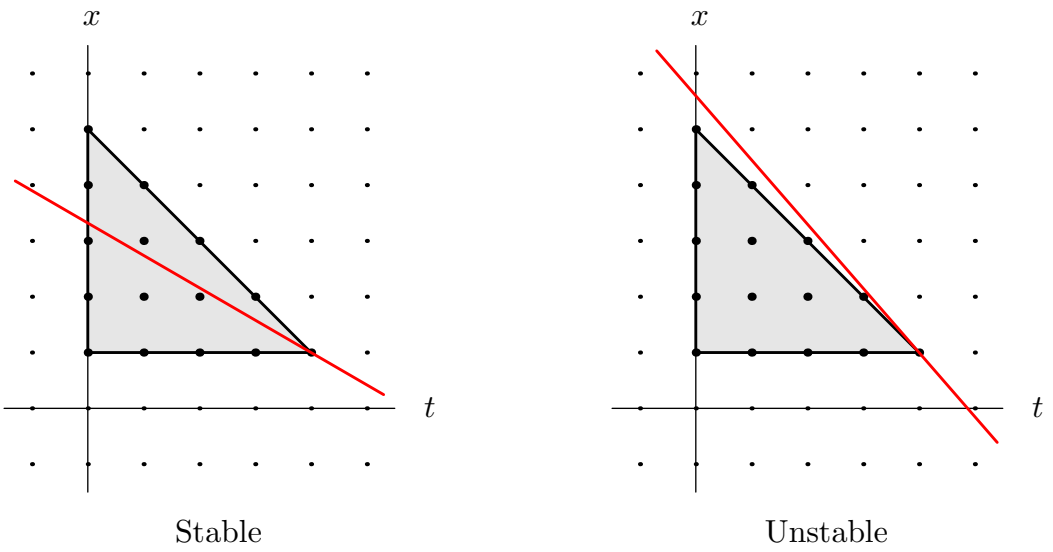



**Figure 11.6.** The CFL Condition.

computed from the previous approximations $u_{n-2,j}$, $u_{n-2,j+1}$, $u_{n-2,j+2}$. And so on. Going all the way back to the initial time $t_0 = 0$, we find that $u_{n,j}$ depends on the initial values $u_{0,j} = f(x_j)$, $\dots$ , $u_{0,j+n} = f(x_j + n\,\Delta x)$ at the mesh points lying in the interval $x_j \le x \le x_j + n\,\Delta x$. On the other hand, the actual solution $u(t_n, x_j)$ depends only on the value of $f(\xi)$ where

$$\xi = x_j - c t_n = x_j - c n\,\Delta t.$$

Thus, if $\xi$ lies outside the interval $[x_j, x_j + n\,\Delta x]$, then varying the initial condition $f(x)$ nearby $x = \xi$ will change the solution value $u(t_n, x_j)$ without affecting its numerical approximation $u_{n,j}$ at all! So the numerical scheme cannot possibly provide an accurate approximation to the solution value. As a result, we must require

$$x_j \le \xi = x_j - c n\,\Delta t \le x_j + n\,\Delta x, \qquad \text{and hence} \qquad 0 \le -c n\,\Delta t \le n\,\Delta x,$$

which we rewrite as

$$0 \ge \sigma = \frac{c\,\Delta t}{\Delta x} \ge -1, \qquad \text{or, equivalently,} \qquad -\frac{\Delta x}{\Delta t} \le c \le 0. \tag{11.42}$$

This is the simplest manifestation of the *Courant–Friedrichs–Lewy condition*, or *CFL condition* for short, established in the 1950's by three of the pioneers in the development of numerical solutions schemes for hyperbolic partial differential equations. Note that the CFL condition requires that the wave speed be *negative*, but not too negative. For negative wave speeds, the method is conditionally stable, since stability restricts the possible time step sizes.

The CFL condition can be recast in a more geometrically transparent manner as follows. For the finite difference scheme (11.39), the *numerical domain of dependence* of a point $(t_n, x_j)$ is the triangle

$$T_{(t_n, x_j)} = \big\{\, (t,x) \;\big|\; 0 \le t \le t_n,\ x_j \le x \le x_j + t_n - t \,\big\}. \tag{11.43}$$

The reason for this term is that the numerical approximation to the solution at $(t_n, x_j)$ depends on the computed values at the mesh points lying within its numerical domain of dependence; see Figure 11.6. The *CFL condition* (11.42) requires that, for $0 \le t \le t_i$, the characteristic passing through the point $(t_n, x_j)$ lies entirely within the numerical domain of dependence (11.43). If the characteristic ventures outside the domain, then the scheme will be numerically unstable. With this geometric formulation, the CFL criterion can be applied to both linear and nonlinear transport equations that have non-uniform wave speeds.

The CFL criterion (11.42) is reconfirmed by a von Neumann stability analysis. We test the numerical scheme on an exponential function, as in (11.22, 25): Substituting

$$u_{i,j} = e^{\mathrm{i}\,k\,x_j}, \qquad u_{i+1,j} = \lambda e^{\mathrm{i}\,k\,x}, \tag{11.44}$$

into (11.39) leads to

$$\lambda e^{\mathrm{i}\,k\,x} = -\sigma\, e^{\mathrm{i}\,k\,x_{j+1}} + (\sigma+1) e^{\mathrm{i}\,k\,x_j} = \big(-\sigma\, e^{\mathrm{i}\,k\,\Delta x} + \sigma + 1\big) e^{\mathrm{i}\,k\,x_j}.$$

The resulting magnification factor

$$\lambda = 1 + \sigma\big(1 - e^{\mathrm{i}\,k\,\Delta x}\big)$$

satisfies the stability criterion (11.27) if and only if

$$|\,\lambda\,|^2 = 1 + 2\sigma(\sigma+1)(1 - \cos(k\,\Delta x)) \le 1,$$

for all $k$. Thus, stability requires that $-1 \le \sigma \le 0$, in complete accord with the CFL condition (11.42).

To obtain a finite difference scheme that can be used for positive wave speeds, we replace the forward finite difference approximation to $\partial u/\partial x$ by the corresponding backwards difference quotient — (11.1) with $h = -\Delta x$ — leading to the alternative first order numerical scheme

$$u_{i+1,j} = (1-\sigma) u_{i,j} + \sigma\, u_{i,j-1}, \tag{11.45}$$

where $\sigma = c\Delta t/\Delta x$ is as above. A similar analysis, left to the reader, produces the corresponding stability criterion

$$0 \le \sigma = \frac{c\,\Delta t}{\Delta x} \le 1,$$

now permitting a positive range of wave speeds.

In this manner, we have produced one numerical scheme that works for a range of negative wave speeds, and an alternative scheme for positive speeds. The question arises — particularly when one is desaling with equations with variable wave speeds — whether one can devise a scheme that is (conditionally) stable for *both* positive and negative wave speeds. One might be tempted to use the centered difference approximation (11.7) for

$$\frac{\partial u}{\partial x}\,(t_i, x_j) \;\approx\; \frac{u_{i,j+1} - u_{i,j-1}}{\Delta x} + \mathrm{O}\big((\Delta x)^2\big). \tag{11.46}$$

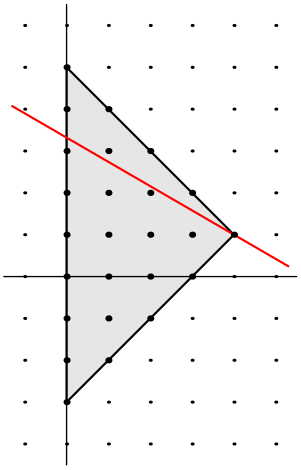
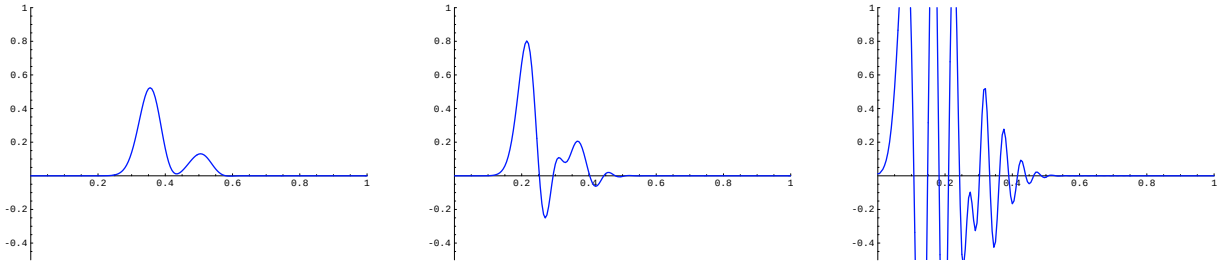


**Figure 11.7.** The CFL Condition for the Centered Difference Scheme.



$t = .15$ $t = .3$ $t = .45$

**Figure 11.8.** Centered Difference Numerical Solution to the Transport Equation.

Substituting this and the previous approximation to the time derivative into (11.36) leads to the numerical scheme

$$u_{i+1,j} = -\tfrac{1}{2}\sigma\, u_{i,j+1} + u_{i,j} + \tfrac{1}{2}\sigma\, u_{i,j-1}, \tag{11.47}$$

where $\sigma = c\,k/h$ is the same as above. In this case, the *numerical domain of dependence* of the mesh point $(t_n, x_j)$ consists of the mesh points in the triangle

$$\widetilde{T}_{(t_n,x_j)} = \big\{\, (t,x) \;\big|\; 0 \le t \le t_n,\ x_j - t_n + t \le x \le x_j + t_n - t \,\big\}. \tag{11.48}$$

The CFL condition asks that, for $0 \le t \le t_n$, the characteristic going through $(t_n, x_j)$ lie within this triangle, as in Figure 11.7, which requires

$$|\,\sigma\,| = \left|\frac{c\,\Delta t}{\Delta x}\right| \le 1, \qquad \text{or, equivalently,} \qquad |\,c\,| \le \frac{\Delta x}{\Delta t}\,. \tag{11.49}$$

Unfortunately, although it satisfies the CFL condition over this range of wave speeds, the centered difference scheme is, in fact, always *unstable*! For instance, the instability of the numerical solution to the preceding initial value problem (11.41) for $c = 1$ can be seen in Figure 11.8. This is confirmed by applying the von Neumann analysis: We

substitute (11.44) into (11.47), and cancel the common exponential factors. The resulting magnification factor

$$\lambda = 1 - \mathrm{i}\,\sigma \sin(k\,\Delta x)$$

satisfies[†] $|\,\lambda\,| > 1$ except when $\sigma = 0$, which would mean $c = 0$. Thus, the centered difference scheme (11.47) is unstable for all (nonzero) wave speeds!

One elementary means of overcoming the sign restriction on the wave speed is to use the forward difference scheme (11.39) when the wave speed is negative and the backwards scheme (11.45) when it is positive. The resulting scheme, valid for varying wave speeds $c(t,x)$,

$$u_{i+1,j} = \begin{cases} -\sigma\, u_{i,j+1} + (\sigma+1)u_{i,j}, & c \le 0, \\ (1-\sigma)u_{i,j} + \sigma\, u_{i,j-1}, & c > 0, \end{cases} \qquad \text{where} \qquad \begin{array}{l} \sigma = \dfrac{c\,\Delta t}{\Delta x}\,, \\ c = c_{i,j} = c(t_i, x_j), \end{array} \tag{11.50}$$

is known as the *upwind scheme*, since the second mesh point always lies "upwind" — that is away from the direction of motion — of the reference point $(t_i, x_j)$. The upwind scheme works reasonably well over short time intervals assuming the space step size is chose sufficiently small and the time step satisfies the CFL condition (11.42). However, over longer time intervals, as we already observed in Figure 11.5, it tends to exhibit an unacceptable rate of damping of waves or, alternatively, require an unacceptably small step size. One way of overcoming this defect is to use the popular *Lax–Wendroff scheme*

$$u_{i+1,j} = \tfrac{1}{2}\,\sigma(1+\sigma)\,u_{i,j+1} + (1-\sigma^2)u_{i,j} - \tfrac{1}{2}\,\sigma(1-\sigma)\,u_{i,j-1}, \tag{11.51}$$

which is based on a quadratic approximation to the derivatives, [**39**]. The stability analysis of this scheme is relegated to the exercises.

## 11.4. Numerical Solution Methods for the Wave Equation.

Let us now develop the basic numerical solution techniques for the second order wave equation. As above, although we are in possession of the explicit d'Alembert solution formula, the lessons learned in designing stable schemes in this simple case will carry over to more complicated equations, including inhomogeneous media and higher dimensional problems, where analytic solution formulas are no longer readily available.

Consider the wave equation

$$\frac{\partial^2 u}{\partial t^2} = c^2\, \frac{\partial^2 u}{\partial x^2}\,, \qquad 0 < x < \ell, \qquad t \ge 0, \tag{11.52}$$

modeling vibrations of a homogeneous bar of length $\ell$ with constant wave speed $c > 0$. We impose Dirichlet boundary conditions

$$u(t,0) = \alpha(t), \qquad u(t,\ell) = \beta(t), \qquad t \ge 0, \tag{11.53}$$

[†] Unless $\sin(k\,\Delta x) = 0$, but we need the stability criterion to hold for all values of $k$.

along with the usual initial conditions

$$u(0,x) = f(x), \qquad \frac{\partial u}{\partial t}\,(0,x) = g(x), \qquad 0 \le x \le \ell. \tag{11.54}$$

We adopt the same uniformly spaced mesh as before

$$t_i = i\,\Delta t, \qquad x_j = j\,\Delta x, \qquad \text{where} \qquad \Delta x = \frac{\ell}{n}\,.$$

In order to discretize the wave equation, we replace the second order derivatives by their standard finite difference approximations (11.6), namely

$$\begin{aligned} \frac{\partial^2 u}{\partial t^2}\,(t_i,x_j) &\approx \frac{u(t_{i+1},x_j) - 2\,u(t_i,x_j) + u(t_{i-1},x_j)}{(\Delta t)^2} + \mathrm{O}\big((\Delta t)^2\big), \\ \frac{\partial^2 u}{\partial x^2}\,(t_i,x_j) &\approx \frac{u(t_i,x_{j+1}) - 2\,u(t_i,x_j) + u(t_i,x_{j-1})}{(\Delta x)^2} + \mathrm{O}\big((\Delta x)^2\big), \end{aligned} \tag{11.55}$$

Since the errors are of orders of $(\Delta t)^2$ and $(\Delta x)^2$, we expect to be able to choose the space and time step sizes of comparable magnitude:

$$\Delta t \approx \Delta x.$$

Substituting the finite difference formulae (11.55) into the partial differential equation (11.52), and rearranging terms, we are led to the iterative system

$$u_{i+1,j} = \sigma^2\,u_{i,j+1} + 2\,(1-\sigma^2)\,u_{i,j} + \sigma^2\,u_{i,j-1} - u_{i-1,j}, \qquad \begin{matrix} i = 1,2,\dots, \\ j = 1,\dots,n-1, \end{matrix} \tag{11.56}$$

for the numerical approximations $u_{i,j} \approx u(t_i,x_j)$ to the solution values at the mesh points. The positive parameter

$$\sigma = \frac{c\,\Delta t}{\Delta x} > 0 \tag{11.57}$$

depends upon the wave speed and the ratio of space and time step sizes. The boundary conditions (11.53) require that

$$u_{i,0} = \alpha_i = \alpha(t_i), \qquad u_{i,n} = \beta_i = \beta(t_i), \qquad i = 0,1,2,\dots. \tag{11.58}$$

This allows us to rewrite the system in matrix form

$$\mathbf{u}^{(i+1)} = B\,\mathbf{u}^{(i)} - \mathbf{u}^{(i-1)} + \mathbf{b}^{(i)}, \tag{11.59}$$

where

$$B = \begin{pmatrix} 2\,(1-\sigma^2) & \sigma^2 & & & \\ \sigma^2 & 2\,(1-\sigma^2) & \sigma^2 & & \\ & \sigma^2 & \ddots & \ddots & \\ & & \ddots & \ddots & \sigma^2 \\ & & & \sigma^2 & 2\,(1-\sigma^2) \end{pmatrix}, \quad \mathbf{u}^{(i)} = \begin{pmatrix} u_{i,1} \\ u_{i,2} \\ \vdots \\ u_{i,n-2} \\ u_{ii,n-1} \end{pmatrix}, \quad \mathbf{b}^{(i)} = \begin{pmatrix} \sigma^2\,\alpha_i \\ 0 \\ \vdots \\ 0 \\ \sigma^2\,\beta_i \end{pmatrix}. \tag{11.60}$$

The entries of $\mathbf{u}^{(i)}$ are, as in (11.18), the numerical approximations to the solution values at the *interior* nodes. Note that (11.59) describes a *second order iterative scheme*, since computing the next iterate $\mathbf{u}^{(i+1)}$ requires the value of the preceding two, $\mathbf{u}^{(i)}$ and $\mathbf{u}^{(i-1)}$.

The one subtlety is how to get the method started. We know $\mathbf{u}^{(0)}$ since $u_{0,j} = f_j = f(x_j)$ is determined by the initial position. However, we also need to find $\mathbf{u}^{(1)}$ with entries $u_{1,j} \approx u(\Delta t, x_j)$ at time $t_1 = \Delta t$ in order launch the iteration and compute $\mathbf{u}^{(2)}, \mathbf{u}^{(3)}, \dots$, but the initial velocity $u_t(0,x) = g(x)$ prescribes the derivatives $u_t(0, x_j) = g_j = g(x_j)$ at time $t_0 = 0$ instead. One way to resolve this difficulty would be to utilize the finite difference approximation

$$g_j = \frac{\partial u}{\partial t}\,(0, x_j) \approx \frac{u(\Delta t, x_j) - u(0, x_j)}{\Delta t} \approx \frac{u_{1,j} - f_j}{\Delta t} \tag{11.61}$$

to compute the required values

$$u_{1,j} = f_j + \Delta t\, g_j.$$

However, the approximation (11.61) is only accurate to order $\Delta t$, whereas the rest of the scheme has errors proportional to $(\Delta t)^2$. The effect would be to introduce an unacceptably large error at the initial step.

To construct an initial approximation to $\mathbf{u}^{(1)}$ with error on the order of $(\Delta t)^2$, we need to analyze the local error in (11.61) in more detail. Note that, by Taylor's theorem,

$$\begin{aligned} \frac{u(\Delta t, x_j) - u(0, x_j)}{\Delta t} &= \frac{\partial u}{\partial t}\,(0, x_j) + \frac{\Delta t}{2}\,\frac{\partial^2 u}{\partial t^2}\,(0, x_j) + \mathrm{O}\big((\Delta t)^2\big) \\ &= \frac{\partial u}{\partial t}\,(0, x_j) + \frac{c^2\,\Delta t}{2}\,\frac{\partial^2 u}{\partial x^2}\,(0, x_j) + \mathrm{O}\big((\Delta t)^2\big), \end{aligned}$$

where, in the final equality, we have used the fact that $u(t,x)$ solves the wave equation. Therefore, we find

$$\begin{aligned} u_{1,j} = u(\Delta t, x_j) &\approx u(0, x_j) + \Delta t\,\frac{\partial u}{\partial t}\,(0, x_j) + \frac{c^2\,(\Delta t)^2}{2}\,\frac{\partial^2 u}{\partial x^2}\,(0, x_j) \\ &= f(x_j) + \Delta t\, g(x_j) + \frac{c^2\,(\Delta t)^2}{2}\,f''(x_j) \approx f_j + \Delta t\, g_j + \frac{c^2\,(\Delta t)^2}{2\,(\Delta x)^2}(f_{j+1} - 2\,f_j + f_{j-1})\,, \end{aligned}$$

where we can use the finite difference approximation (11.6) for the second derivative of $f(x)$ if no explicit formula is known. Therefore, when we initiate the scheme by setting

$$u_{1,j} = \tfrac{1}{2}\,\sigma^2\,f_{j+1} + (1 - \sigma^2) f_j + \tfrac{1}{2}\,\sigma^2\,f_{j-1} + \Delta t\, g_j, \tag{11.62}$$

or, in matrix form,

$$\mathbf{u}^{(0)} = \mathbf{f}, \qquad \mathbf{u}^{(1)} = \tfrac{1}{2}\,B\,\mathbf{u}^{(0)} + \Delta t\,\mathbf{g} + \tfrac{1}{2}\,\mathbf{b}^{(0)}, \tag{11.63}$$

we will have maintained the desired order $(\Delta t)^2$ (and $(\Delta x)^2$) accuracy.

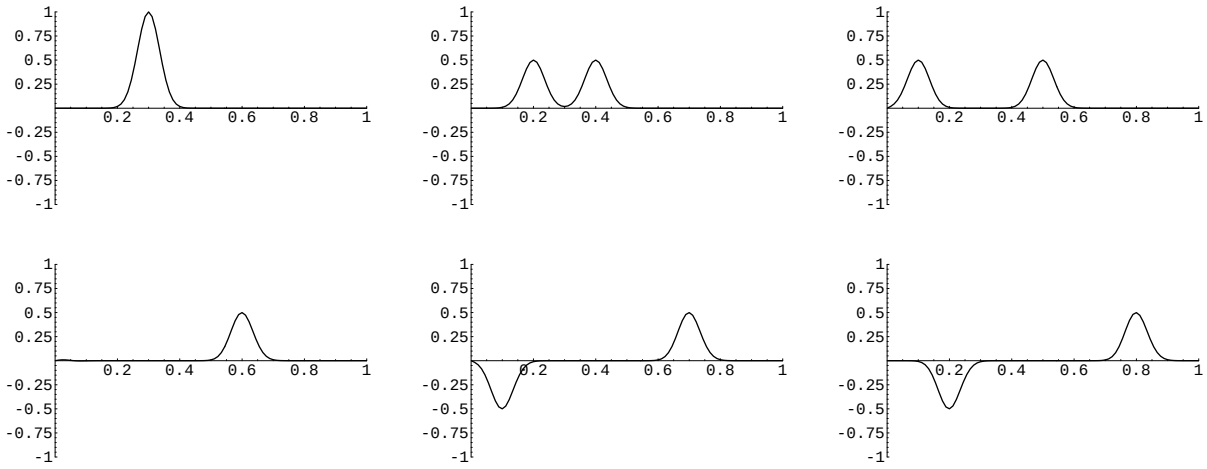


**Figure 11.9.** Numerically Stable Waves.

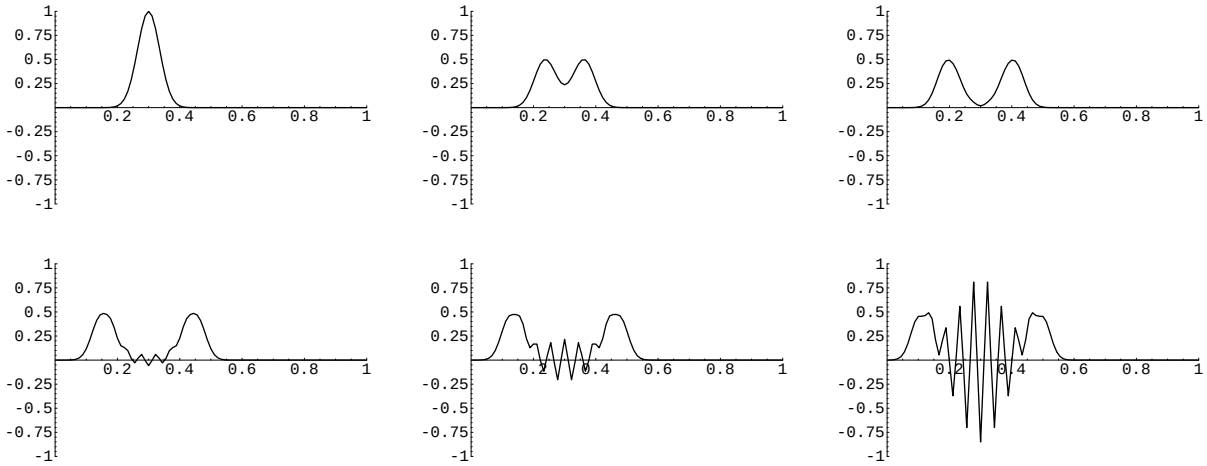


**Figure 11.10.** Numerically Unstable Waves.

**Example 11.6.** Consider the particular initial value problem

$$u_{tt} = u_{xx}, \qquad \begin{aligned} u(0,x) &= e^{-400\,(x-.3)^2}, \qquad u_t(0,x) = 0, \qquad && 0 \le x \le 1, \\ u(t,0) &= u(1,0) = 0, && t \ge 0, \end{aligned}$$

subject to homogeneous Dirichlet boundary conditions on the interval $[0,1]$. The initial data is a fairly concentrated single hump centered at $x = .3$, and we expect it to split into two half sized humps, which then collide with the ends. Let us choose a space discretization consisting of 90 equally spaced points, and so $\Delta x = \frac{1}{90} = .0111\ldots$. If we choose a time step of $\Delta t = .01$, whereby $\sigma = .9$, then we get reasonably accurate solution over a fairly long time range, as plotted in Figure 11.9 at times $t = 0, .1, .2, \ldots, .5$. On the other hand, if we double the time step, setting $\Delta t = .02$, so $\sigma = 1.8$, then, as plotted in Figure 11.10 at times $t = 0, .05, .1, .14, .16, .18$, we observe an instability that eventually overwhelms the numerical solution. Thus, the numerical scheme appears to only be conditionally stable.

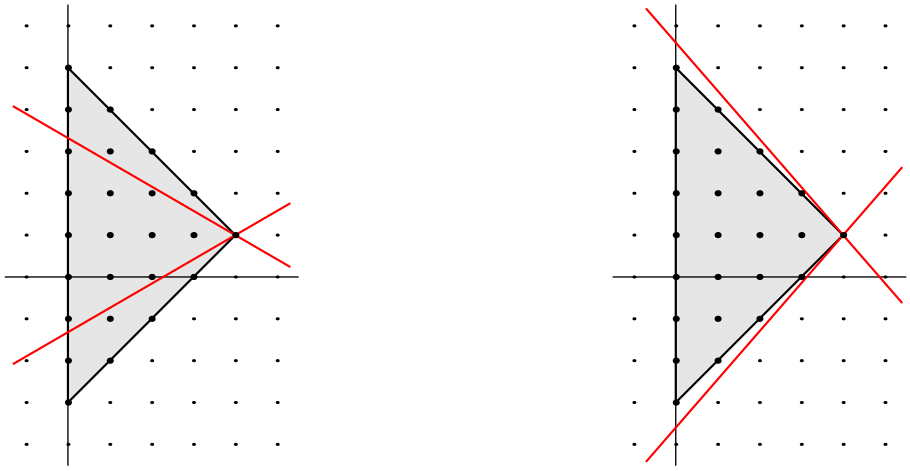



**Figure 11.11.** The CFL Condition for the Wave Equation.

The stability analysis of this numerical scheme proceeds along the same lines as in the first order case. The CFL condition for stability requires that the characteristics emanating from a mesh point $(t_n, x_j)$ must, for $0 \le t \le t_n$, remain in its numerical domain of dependence, which, for our particular numerical scheme, is the same triangle

$$\widetilde{T}_{(t_n,x_j)} = \big\{\, (t,x) \;\big|\; 0 \le t \le t_n,\ x_j - t_n + t \le x \le x_j + t_n - t \,\big\},$$

that we plotted Figure 11.11. Since the characteristics are the lines of slope $\pm c$, the CFL condition is the same as in (11.49):

$$\sigma = \frac{c\,\Delta t}{\Delta x} \le 1, \qquad \text{or, equivalently,} \qquad 0 \le c \le \frac{\Delta x}{\Delta t}. \tag{11.64}$$

This explains the difference between the numerically stable and unstable cases exhibited above.

However, as we noted above, the CFL condition is only necessary for stability of the numerical scheme; one should perform a von Neumann stability analysis for a complete verification. Inspired by fourier analysis, the goal is to determine how the scheme affects complex exponentials. Specifically, after one time step, the scheme has the effect of multiplying a complex exponential $e^{\mathrm{i}\,k x}$ by the (local) *magnification factor* $\lambda = \lambda(k)$. In this case, we set

$$u_{i-1,j} = e^{\mathrm{i}\,k x_j}, \qquad u_{i,j} = \lambda\, e^{\mathrm{i}\,k x_j}, \qquad u_{i+1,j} = \lambda^2\, e^{\mathrm{i}\,k x_j}, \tag{11.65}$$

since going from $t_{i-1}$ to $t_i$ multiplies the exponential by $\lambda$, and from $t_i$ to $t_{i+1}$ multiplies by an additional factor $\lambda$. Substituting (11.65) into[†] (11.56) and canceling the common exponential, we find that each magnification factor must satisfy the following quadratic equation:

$$\lambda^2 = \big(\, 2 - 4\sigma^2 \sin^2 \tfrac{1}{2} k\,\Delta x \,\big)\lambda + 1,$$

[†] Stability is a long term effect, and so we need not concern ourselves with the anomalous first time step.

whence

$$\lambda = \alpha \pm \sqrt{\alpha^2 - 1}\,, \qquad \text{where} \qquad \alpha = 1 - 2\sigma^2 \sin^2 \tfrac{1}{2} k \Delta x. \tag{11.66}$$

Thus, there are two different magnification factors associated with each complex exponential — which is a consequence of the scheme being of second order. Stability requires that both be $\leq 1$ in modulus. Now, if the CFL condition (11.64) holds, then $|\,\alpha\,| \leq 1$, which implies that the magnification factors (11.66) are complex numbers of modulus $|\,\lambda\,| = 1$, and thus the numerical scheme satisfies the stability criterion (11.27). On the other hand, if $\sigma > 1$, then, for some values of $k$, we have $\alpha < -1$, which implies that the two factors (11.66) are both real, one of which is $< -1$, and thus violates the stability criterion. Thus, the CFL condition (11.64) does indeed distinguish between the (conditionally) stable and unstable numerical schemes for the wave equation.