

Part three

Multidimensional Problems

Copyright © 2002, Cambridge University Press.

All rights reserved. May not be reproduced in any form without permission from the publisher, except fair uses permitted under U.S. or applicable copyright law.

Practical problems involving conservation laws and hyperbolic systems must frequently be solved in more than one space dimension. Most of this book has been devoted to the one-dimensional theory and algorithms, but in the remaining chapters we will see that this forms the basis for understanding and solving multidimensional problems as well.

In two dimensions a conservation law takes the form

$$q_t + f(q)_x + g(q)_y = 0, \quad (18.1)$$

where $q(x, y, t)$ is a vector of m conserved quantities, and $f(q)$ and $g(q)$ are flux functions in the x - and y -directions, as described below. More generally, a quasilinear hyperbolic system has the form

$$q_t + A(q, x, y, t)q_x + B(q, x, y, t)q_y = 0, \quad (18.2)$$

where the matrices A and B satisfy certain conditions given below in Section 18.5. In three dimensions a third term would be added to each of these equations:

$$q_t + f(q)_x + g(q)_y + h(q)_z = 0, \quad (18.3)$$

and

$$q_t + A(q, x, y, z, t)q_x + B(q, x, y, z, t)q_y + C(q, x, y, z, t)q_z = 0, \quad (18.4)$$

respectively.

18.1 Derivation of Conservation Laws

We begin by deriving the conservation law (18.1) in two dimensions from the more fundamental integral form. Again the integral form can be used directly in the development of finite volume methods, as we will see beginning in Chapter 19.

As in one dimension (see Chapter 2), we derive the conservation law by considering an arbitrary spatial domain Ω over which q is assumed to be conserved, so that the integral of q over Ω varies only due to flux across the boundary of Ω . This boundary is denoted by

$\partial\Omega$. We thus have

$$\frac{d}{dt} \iint_{\Omega} q(x, y, t) dx dy = \text{net flux across } \partial\Omega. \quad (18.5)$$

The net flux is determined by integrating the flux of q normal to $\partial\Omega$ around this boundary.

Let $f(q)$ represent the flux of q in the x -direction (per unit length in y , per unit time). This means that the total flux through an interval from (x_0, y_0) to $(x_0, y_0 + \Delta y)$ over time Δt is roughly $\Delta t \Delta y f(q(x_0, y_0))$, for Δt and Δy sufficiently small.

Similarly, let $g(q)$ be the flux in the y -direction, and let $\vec{f}(q) = (f(q), g(q))$ be the flux vector. Finally, let $\vec{n}(s) = (n^x(s), n^y(s))$ be the outward-pointing unit normal vector to $\partial\Omega$ at a point $(x(s), y(s))$ on $\partial\Omega$, where s is the arclength parameterization of $\partial\Omega$. Then the flux at $\vec{x}(s) = (x(s), y(s))$ in the direction $\vec{n}(s)$ is

$$\vec{n}(s) \cdot \vec{f}(q(x(s), y(s), t)) = n^x(s)f(q) + n^y(s)g(q), \quad (18.6)$$

and (18.5) becomes

$$\frac{d}{dt} \iint_{\Omega} q(x, y, t) dx dy = - \int_{\partial\Omega} \vec{n} \cdot \vec{f}(q) ds. \quad (18.7)$$

If q is smooth then we can use the divergence theorem to rewrite this as

$$\frac{d}{dt} \iint_{\Omega} q(x, y, t) dx dy = - \iint_{\Omega} \vec{\nabla} \cdot \vec{f}(q) dx dy, \quad (18.8)$$

where the divergence of \vec{f} is

$$\vec{\nabla} \cdot \vec{f}(q) = f(q)_x + g(q)_y.$$

This leads to

$$\iint_{\Omega} [q_t + \vec{\nabla} \cdot \vec{f}(q)] dx dy = 0. \quad (18.9)$$

Since this must hold over any arbitrary region Ω , the integrand must be zero, and we obtain the conservation law (18.1) in differential form. Note that this argument is exactly analogous to what we did in Chapter 2 in one dimension. As in the one-dimensional case, this derivation assumes q is smooth and so the differential form holds only for smooth solutions. To properly compute discontinuous solutions we will again use finite volume methods based on the integral form.

The same argument extends to three dimensions with the flux vector

$$\vec{f}(q) = [f(q), g(q), h(q)]. \quad (18.10)$$

If we integrate over an arbitrary volume Ω , so that $\partial\Omega$ is the surface bounding this volume, then we obtain (18.3) for smooth solutions.

In the notation above we have assumed q is a scalar. If $q, f(q), g(q), h(q) \in \mathbb{R}^m$, then the vector \vec{f} of (18.10) is a vector in \mathbb{R}^{3m} , while $\vec{n}(s) = (n^x, n^y, n^z)$ is in \mathbb{R}^3 , and we interpret

dot products by the formula

$$\check{f}(q) \equiv \vec{n} \cdot \vec{f}(q) = n^x f(q) + n^y g(q) + n^z h(q). \quad (18.11)$$

In general we use an arrow on a symbol to denote a spatial vector with components corresponding to each spatial dimension. In addition to \vec{n} and \vec{f} , we also use $\vec{u} = (u, v, w)$ for the velocity vector and $\vec{A} = (A, B, C)$ as a vector of matrices. We will see below that we need to investigate the linear combination

$$\check{A} \equiv \vec{n} \cdot \vec{A} = n^x A + n^y B + n^z C \quad (18.12)$$

or to determine whether the system (18.4) is hyperbolic. We use the breve accent $\check{\cdot}$ to denote a quantity that has been restricted to a particular direction specified by \vec{n} . As a mnemonic device, the circular arc of the breve accent can be thought of as indicating rotation to the desired direction. This will be heavily used in Chapter 23, where we discuss numerical methods on general quadrilateral grids. One-dimensional Riemann problems will be solved in the direction normal to each cell edge in order to compute the normal fluxes, and doing so requires rotating the flux function to that direction.

For simplicity we will mostly restrict our attention to the case of two dimensions, but the essential ideas extend directly to three dimensions, and this case is briefly discussed as we go along.

18.2 Advection

As a simple example, suppose a fluid is flowing with a known velocity $\vec{u} = (u(x, y, t), v(x, y, t))$ in the plane, and let the scalar $q(x, y, t)$ represent the concentration of a tracer, measured in units of mass per unit area in the plane (see Section 9.1). Then the flux functions are

$$\begin{aligned} f &= u(x, y, t) q(x, y, t), \\ g &= v(x, y, t) q(x, y, t), \end{aligned} \quad (18.13)$$

so that $\vec{f}(q) = \vec{u}q$ and we obtain the conservation law

$$q_t + (uq)_x + (vq)_y = 0. \quad (18.14)$$

Note that in this case f and g may depend explicitly on (x, y, t) as well as on the value of q , and that the derivation above carries over to this situation. If in fact $(u(x, y, t), v(x, y, t)) = (\bar{u}, \bar{v})$ is constant in space and time, so the fluid is simply translating at constant speed in a fixed direction, then (18.14) reduces to

$$q_t + \bar{u}q_x + \bar{v}q_y = 0.$$

The solution is then easily seen to be

$$q(x, y, t) = \check{q}(x - \bar{u}t, y - \bar{v}t), \quad (18.15)$$

so that the initial density simply translates at this velocity. The solution to the more general variable-coefficient advection equation is discussed in Section 20.5.

18.3 Compressible Flow

In Section 2.6 we saw the equations of compressible gas dynamics in one space dimension, in the simplest case where the equation of state relates the pressure directly to the density,

$$p = P(\rho). \quad (18.16)$$

These equations are easily extended to two space dimensions. The conservation-of-mass equation (continuity equation) for the density $\rho(x, y, t)$ is identical to the advection equation (18.14) derived in the previous section:

$$\rho_t + (\rho u)_x + (\rho v)_y = 0. \quad (18.17)$$

But now the velocities (u, v) are not known *a priori*. Instead the continuity equation must be coupled with equations for the conservation of x -momentum ρu and y -momentum ρv . Each of these momenta advects with the fluid motion, giving fluxes analogous to (18.14) with q replaced by ρu or ρv respectively. In addition, pressure variations lead to acceleration of the fluid. Variation in the x -direction, measured by p_x , accelerates the fluid in that direction and appears in the equation for $(\rho u)_t$, while p_y appears in the equation for $(\rho v)_t$. The conservation-of-momentum equations are

$$\begin{aligned} (\rho u)_t + (\rho u^2 + p)_x + (\rho uv)_y &= 0, \\ (\rho v)_t + (\rho uv)_x + (\rho v^2 + p)_y &= 0. \end{aligned} \quad (18.18)$$

These equations (18.17) and (18.18), together with the equation of state (18.16), give a closed system of three conservation laws for mass and momentum. If the equation of state is more complicated, then these equations will generally also have to be coupled with the equation for the conservation of energy, as discussed in Chapter 14. The simple case will suffice for our purposes now. In particular, from these equations we can derive the linearized equations of acoustics just as we did in Section 2.8 in one space dimension.

These gas dynamics equations can be written as a system of conservation laws of the form (18.1) with

$$\begin{aligned} q &= \begin{bmatrix} \rho \\ \rho u \\ \rho v \end{bmatrix}, & f(q) &= \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \end{bmatrix} = \begin{bmatrix} q^2 \\ (q^2)^2/q^1 + P(q^1) \\ q^2 q^3/q^1 \end{bmatrix}, \\ g(q) &= \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \end{bmatrix} = \begin{bmatrix} q^3 \\ q^2 q^3/q^1 \\ (q^3)^2/q^1 + P(q^1) \end{bmatrix}. \end{aligned} \quad (18.19)$$

These equations can also be written in quasilinear form

$$q_t + f'(q)q_x + g'(q)q_y = 0, \quad (18.20)$$

in terms of the Jacobian matrices

$$f'(q) = \begin{bmatrix} 0 & 1 & 0 \\ -u^2 + P'(\rho) & 2u & 0 \\ -uv & v & u \end{bmatrix}, \quad g'(q) = \begin{bmatrix} 0 & 0 & 1 \\ -uv & v & u \\ -v^2 + P'(\rho) & 0 & 2v \end{bmatrix}. \quad (18.21)$$

18.4 Acoustics

Linearizing the equations derived in the previous section about a constant state $q_0 = (\rho_0, u_0, v_0)$ gives

$$q_t + f'(q_0)q_x + g'(q_0)q_y = 0,$$

where q now represents perturbations from the constant state q_0 . In particular, if we wish to study acoustics in a stationary gas, then we can take $u_0 = v_0 = 0$ and the Jacobian matrices simplify considerably:

$$f'(q_0) = \begin{bmatrix} 0 & 1 & 0 \\ P'(\rho_0) & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad g'(q_0) = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ P'(\rho_0) & 0 & 0 \end{bmatrix}. \quad (18.22)$$

As in one dimension, we can now manipulate these equations to derive an equivalent linear system in terms of perturbations in pressure and velocity, a linear system of the form

$$q_t + Aq_x + Bq_y = 0, \quad (18.23)$$

where (again for $u_0 = v_0 = 0$)

$$q = \begin{bmatrix} p \\ u \\ v \end{bmatrix}, \quad A = \begin{bmatrix} 0 & K_0 & 0 \\ 1/\rho_0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 & K_0 \\ 0 & 0 & 0 \\ 1/\rho_0 & 0 & 0 \end{bmatrix}. \quad (18.24)$$

These are the *equations of acoustics* in two space dimensions, where again $K_0 = \rho_0 P'(\rho_0)$ is the bulk modulus of compressibility. More generally, for acoustics against a background flow with constant velocity $\vec{u}_0 = (u_0, v_0)$, the coefficient matrices are

$$A = \begin{bmatrix} u_0 & K_0 & 0 \\ 1/\rho_0 & u_0 & 0 \\ 0 & 0 & u_0 \end{bmatrix}, \quad B = \begin{bmatrix} v_0 & 0 & K_0 \\ 0 & v_0 & 0 \\ 1/\rho_0 & 0 & v_0 \end{bmatrix}. \quad (18.25)$$

18.5 Hyperbolicity

Recall that in one space dimension the linear system $q_t + Aq_x = 0$ is said to be *hyperbolic* if the matrix A is diagonalizable with real eigenvalues. In two space dimensions we need this condition to hold for each of the coefficient matrices A and B , but we also need something more: the same property should hold for any linear combination of these matrices. This is formalized in Definition 18.1 below after some motivation.

The essence of hyperbolicity is that wavelike solutions should exist. In one space dimension a linear system of m equations generally gives rise to m waves moving at constant speeds and unchanged shape. In two dimensions we should see this same behavior if we take special initial data that varies only in one direction – not just the x - or y -direction, but any arbitrary direction specified by a unit vector $\vec{n} = (n^x, n^y)$, so that the data has the form

$$q(x, y, 0) = \hat{q}(\vec{n} \cdot \vec{x}) = \hat{q}(n^x x + n^y y). \quad (18.26)$$

The contour lines of $q(x, y, 0)$ are straight lines, and we expect wave motion in the direction normal to these lines, which is the direction given by \vec{n} .

In particular, there should be special initial data of this form that yields a *single* wave propagating at some constant speed s , a *plane wave* of the form

$$q(x, y, t) = \check{q}(\vec{n} \cdot \vec{x} - st).$$

This is the multidimensional analogue of the simple wave discussed in Section 3.5. If we compute q_t , q_x , and q_y for this *Ansatz* and insert them into the equation $q_t + Aq_x + Bq_y = 0$, we find that

$$\check{A} \check{q}'(\vec{n} \cdot \vec{x} - st) = s \check{q}'(\vec{n} \cdot \vec{x} - st),$$

where

$$\check{A} = \vec{n} \cdot \vec{A} = n^x A + n^y B. \quad (18.27)$$

Except for the trivial case $\check{q} \equiv \text{constant}$, (18.27) can only hold if s is an eigenvalue of the matrix \check{A} , with $\check{q}'(\xi)$ a corresponding eigenvector of this matrix for each value of ξ . This leads to our definition of hyperbolicity in two space dimensions.

Definition 18.1. The constant-coefficient system $q_t + Aq_x + Bq_y = 0$ is (strongly) hyperbolic if, for every choice of \vec{n} , the matrix $\check{A} = \vec{n} \cdot \vec{A}$ is diagonalizable with real eigenvalues. The quasilinear system (18.20) is hyperbolic in some region of state space if the Jacobian matrix $\check{f}'(q) = \vec{n} \cdot \vec{f}'(q) = n^x f'(q) + n^y g'(q)$ is diagonalizable with real eigenvalues for every \vec{n} , for all q in this region.

Note in particular that for $\vec{n} = (1, 0)$ or $\vec{n} = (0, 1)$ we have propagation in the x - or y -direction respectively. In these cases we obtain the usual one-dimensional conditions on the matrices A and B separately. The obvious three-dimensional extension of this definition is given in Section 18.6.

For the acoustics equations with (18.24), we have

$$\check{A} = \begin{bmatrix} 0 & n^x K_0 & n^y K_0 \\ n^x / \rho_0 & 0 & 0 \\ n^y / \rho_0 & 0 & 0 \end{bmatrix}. \quad (18.28)$$

This matrix has eigenvalues that are independent of \vec{n} :

$$\lambda^1 = -c_0, \quad \lambda^2 = 0, \quad \lambda^3 = +c_0,$$

where $c_0 = \sqrt{K_0 / \rho_0}$ is the speed of sound. This is exactly what we should expect, since sound waves can propagate in any direction at the same speed (for the uniform isotropic medium we are considering here, with ρ_0 and K_0 constant).

For acoustics against a constant background flow, we expect sound waves to propagate at speed c_0 relative to the moving fluid. For the matrices A and B of (18.25) we have

$$\check{A} = \begin{bmatrix} \check{u}_0 & n^x K_0 & n^y K_0 \\ n^x/\rho_0 & \check{u}_0 & 0 \\ n^y/\rho_0 & 0 & \check{u}_0 \end{bmatrix}, \quad (18.29)$$

where $\check{u}_0 = \vec{n} \cdot \vec{u}_0$ is the fluid velocity in the \vec{n} -direction. Since this differs from (18.28) only by a multiple of the identity matrix, it has the same eigenvectors (given below in (18.33)), and the eigenvalues are simply shifted by \check{u}_0 :

$$\check{\lambda}^1 = \check{u}_0 - c_0, \quad \check{\lambda}^2 = \check{u}_0, \quad \check{\lambda}^3 = \check{u}_0 + c_0, \quad (18.30)$$

exactly as we expected.

In one space dimension we can diagonalize a general linear hyperbolic equation using the matrix of eigenvectors, decoupling it into independent scalar advection equations for each characteristic variable. For a linear system in more dimensions, we can do this in general only for the special case of a plane-wave solution. The full system $q_t + Aq_x + Bq_y = 0$ with arbitrary data can be diagonalized only if the coefficient matrices commute, e.g., if $AB = BA$ in (18.23), in which case the matrices have the *same* eigenvectors. Then A and B can be *simultaneously diagonalized* by a common eigenvector matrix R :

$$A = R\Lambda^x R^{-1}, \quad B = R\Lambda^y R^{-1},$$

where $\Lambda^x = \text{diag}(\lambda^{x1}, \dots, \lambda^{xm})$ and $\Lambda^y = \text{diag}(\lambda^{y1}, \dots, \lambda^{ym})$ contain the eigenvalues, which may be different. The system in (18.23) can then be diagonalized by setting $w = R^{-1}q$ to obtain

$$w_t + \Lambda^x w_x + \Lambda^y w_y = 0,$$

yielding m independent advection equations. Note that in this case there are only m distinct directions in which information propagates. The p th characteristic variable w^p propagates with velocity $(\lambda^{xp}, \lambda^{yp})$. This is not what we would expect in acoustics, for example, since sound waves can propagate in any direction.

If $AB \neq BA$, then there is no single transformation that will simultaneously diagonalize A and B . If the system is hyperbolic, then we can diagonalize each matrix separately,

$$A = R^x \Lambda^x (R^x)^{-1}, \quad B = R^y \Lambda^y (R^y)^{-1},$$

but the two matrices have different eigenvectors R^x and R^y , respectively. In this case the equations are more intricately coupled. This is the usual situation physically. In the case of acoustics, for example, the matrices A and B of (18.24) or (18.25) are not simultaneously diagonalizable. The matrix A of (18.25) has right eigenvectors

$$r^{x1} = \begin{bmatrix} -Z_0 \\ 1 \\ 0 \end{bmatrix}, \quad r^{x2} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad r^{x3} = \begin{bmatrix} Z_0 \\ 1 \\ 0 \end{bmatrix}, \quad \text{so that } R^x = \begin{bmatrix} -Z_0 & 0 & Z_0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \quad (18.31)$$

while the matrix B has right eigenvectors

$$r^{y1} = \begin{bmatrix} -Z_0 \\ 0 \\ 1 \end{bmatrix}, \quad r^{y2} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad r^{y3} = \begin{bmatrix} Z_0 \\ 0 \\ 1 \end{bmatrix}, \quad \text{so that } R^y = \begin{bmatrix} -Z_0 & 0 & Z_0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}, \quad (18.32)$$

where $Z_0 = q_0 c_0$ is again the impedance.

Note that for a plane wave moving in the x -direction these acoustics equations reduce to $q_t + Aq_x = 0$ with A given by (18.25). This system has exactly the same eigenstructure as the coupled acoustics–advection system of Section 3.10. In this case the y -component of the velocity perturbation, v , is simply advected with the background velocity u_0 and does not affect the acoustics. (Recall that in such a plane wave we assume the variables only vary with x . Variations of v in the y -direction would of course generate acoustic signals.)

A plane wave in the y -direction gives a similar structure, but now the eigenvectors r^{y1} and r^{y3} corresponding to acoustic waves involving the pressure p and vertical velocity perturbation v , while the x -component of the velocity perturbations, u , is simply advected at the background speed v_0 .

The more general matrix \tilde{A} from (18.29) has eigenvalues (18.30) and the eigenvectors

$$\check{r}^1 = \begin{bmatrix} -Z_0 \\ n^x \\ n^y \end{bmatrix}, \quad \check{r}^2 = \begin{bmatrix} 0 \\ -n^y \\ n^x \end{bmatrix}, \quad \check{r}^3 = \begin{bmatrix} Z_0 \\ n^x \\ n^y \end{bmatrix}, \quad (18.33)$$

which reduce to (18.31) or (18.32) when \vec{n} is in the x - or y -direction. Note that more generally the acoustic waves \check{r}^1 and \check{r}^3 have velocity components in the \vec{n} -direction, as we expect for these compressional waves. The 2-wave carries velocity perturbations in the orthogonal direction (a shear wave), which are simply advected with the flow. In Section 22.1 we consider elastic waves in a solid that resists shear motion, in which case shear waves have more interesting structure.

In one space dimension we could diagonalize the acoustics equations to obtain a coupled pair of advection equations. Solutions consist simply of two waves advecting with velocities $-c_0$ and $+c_0$ in the two possible directions. In two dimensions, even though the structure of each matrix is that of one-dimensional acoustics, the nondiagonalizable coupling between them leads to a much richer structure. In general we obtain waves propagating in all of the infinitely many possible directions in the plane.

18.6 Three-Dimensional Systems

The three-dimensional linear system

$$q_t + Aq_x + Bq_y + Cq_z = 0 \quad (18.34)$$

is hyperbolic provided that, for any direction defined by the unit vector $\vec{n} = (n^x, n^y, n^z)$, the matrix \tilde{A} given by (18.12) is diagonalizable with real eigenvalues and a complete set of eigenvectors. The eigenvalues have the interpretation of physical propagation velocities for plane waves in this direction.

The acoustics equations in three dimensions are a linear hyperbolic system for perturbations in the pressure and three velocity components (u, v, w). Rather than displaying the coefficient matrices A , B , and C separately, it is more compact to just display \check{A} in an arbitrary direction \vec{n} . We have

$$q = \begin{bmatrix} p \\ u \\ v \\ w \end{bmatrix}, \quad \check{A} = \begin{bmatrix} \check{u}_0 & n^x K_0 & n^y K_0 & n^z K_0 \\ n^x/\rho_0 & \check{u}_0 & 0 & 0 \\ n^y/\rho_0 & 0 & \check{u}_0 & 0 \\ n^z/\rho_0 & 0 & 0 & \check{u}_0 \end{bmatrix}, \quad (18.35)$$

where

$$\check{u}_0 = n^x u_0 + n^y v_0 + n^z w_0 \quad (18.36)$$

is the component of the background velocity in the direction \vec{n} . For any choice of direction, the eigenvalues of \check{A} are

$$\check{\lambda}^1 = \check{u}_0 - c_0, \quad \check{\lambda}^2 = \check{\lambda}^3 = \check{u}_0, \quad \check{\lambda}^4 = \check{u}_0 + c_0. \quad (18.37)$$

Note that there is a two-dimensional eigenspace corresponding to the eigenvalue $\check{\lambda}^2 = \check{\lambda}^3 = \check{u}_0$, since shear waves can now carry an arbitrary jump in each of the two velocity components orthogonal to \vec{n} .

18.7 Shallow Water Equations

In two space dimensions the shallow water equations take the form

$$\begin{aligned} h_t + (hu)_x + (hv)_y &= 0, \\ (hu)_t + \left(hu^2 + \frac{1}{2}gh^2 \right)_x + (huv)_y &= 0, \\ (hv)_t + (huv)_x + \left(hv^2 + \frac{1}{2}gh^2 \right)_y &= 0, \end{aligned} \quad (18.38)$$

where h is the depth and (u, v) the velocity vector, so that hu and hv are the momenta in the two directions. These are a natural generalization of the one-dimensional equations (13.5) and are identical to the two-dimensional compressible flow equations derived in Section 18.3 if we replace ρ by h there and use the hydrostatic equation of state

$$p = P(h) = \frac{1}{2}gh^2 \quad (18.39)$$

as derived in (13.3) (taking $\bar{\rho} = 1$). From (18.21), the flux Jacobian matrices are thus

$$f'(q) = \begin{bmatrix} 0 & 1 & 0 \\ -u^2 + gh & 2u & 0 \\ -uv & v & u \end{bmatrix}, \quad g'(q) = \begin{bmatrix} 0 & 0 & 1 \\ -uv & v & u \\ -v^2 + gh & 0 & 2v \end{bmatrix}. \quad (18.40)$$

Let $c = \sqrt{gh}$ be the speed of gravity waves. Then the matrix $f'(q)$ has eigenvalues and eigenvectors

$$\begin{aligned}\lambda^{x1} &= u - c, & \lambda^{x2} &= u, & \lambda^{x3} &= u + c, \\ r^{x1} &= \begin{bmatrix} 1 \\ u - c \\ v \end{bmatrix}, & r^{x2} &= \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, & r^{x3} &= \begin{bmatrix} 1 \\ u + c \\ v \end{bmatrix}. \end{aligned} \quad (18.41)$$

The Jacobian $g'(q)$ has a similar set of eigenvalues and eigenvectors,

$$\begin{aligned}\lambda^{y1} &= v - c, & \lambda^{y2} &= v, & \lambda^{y3} &= v + c, \\ r^{y1} &= \begin{bmatrix} 1 \\ u \\ v - c \end{bmatrix}, & r^{y2} &= \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix}, & r^{y3} &= \begin{bmatrix} 1 \\ u \\ v + c \end{bmatrix} \end{aligned} \quad (18.42)$$

in which the roles of u and v are switched along with x and y .

In each case the 1-wave and 3-wave are nonlinear gravity waves, while the 2-wave is linearly degenerate. Compare these with (13.64) and (13.65), the Jacobian matrix and eigenstructure for the one-dimensional shallow water equations augmented by a passive tracer.

18.7.1 The Plane-Wave Riemann Problem

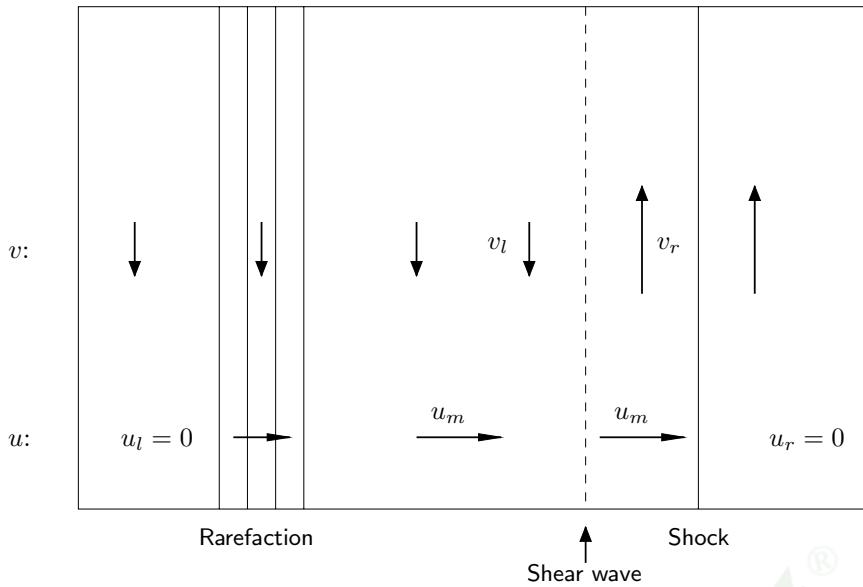
Consider a two-dimensional Riemann problem for the shallow water equations with variation only in the x -direction. In this case the velocity v plays no dynamic role in the gravity waves, and any jump in v is simply carried along passively at the fluid velocity u_m that arises between the two nonlinear waves. This is again a contact discontinuity that lies at the interface between the two original fluids. The fluid to the left always has y -velocity v_l while the one to the right has y -velocity v_r . Figure 13.20 illustrates this, if we let the dark and light regions now represent different velocities v . Figure 18.1 gives another illustration of this, showing a top view of a two-dimensional version of Figure 13.20. The contact discontinuity is also called a *shear wave* in this context.

One should recall that we are ignoring fluid viscosity with this hyperbolic model. In reality a jump discontinuity in shear velocity would be smeared out due to frictional forces (diffusion of the momentum hv in the x -direction) and may lead to Kelvin–Helmholtz instabilities along such an interface.

The true solution to this Riemann problem is easily computed using the one-dimensional theory. We simply solve the one-dimensional problem ignoring v , and then introduce a jump in v at the contact surface.

To verify that the two-dimensional shallow water equations are hyperbolic, we compute the Jacobian matrix $\check{f}'(q) = \vec{n} \cdot \vec{f}'(q)$ in an arbitrary direction \vec{n} ,

$$\check{f}'(q) = \begin{bmatrix} 0 & n^x & n^y \\ n^x gh - uu & \check{u} + n^x u & n^y u \\ n^y gh - vu & n^x v & \check{u} + vn^y \end{bmatrix}, \quad (18.43)$$



Copyright © 2002, Cambridge University Press.
All rights reserved. May not be reproduced in any form without permission from the publisher.

Fig. 18.1. Solution to a Riemann problem in the x -direction for the two-dimensional shallow water equations. The depth h and normal velocity u are as shown in Figure 13.20, and the shading in that figure represents v .

where $\check{u} = \vec{n} \cdot \vec{u}$. This matrix has eigenvalues and eigenvectors given by

$$\begin{aligned}\check{\lambda}^1 &= \check{u} - c, & \check{\lambda}^2 &= \check{u}, & \check{\lambda}^3 &= \check{u} + c, \\ \check{r}^1 &= \begin{bmatrix} 1 \\ u - n^x c \\ v - n^y c \end{bmatrix}, & \check{r}^2 &= \begin{bmatrix} 0 \\ -n^y \\ n^x \end{bmatrix}, & \check{r}^3 &= \begin{bmatrix} 1 \\ u + n^x c \\ v + n^y c \end{bmatrix}. \end{aligned} \quad (18.44)$$

The expressions (18.41) and (18.42) are special cases of this. For any direction \vec{n} the eigenvalues are real and correspond to wave speeds $0, \pm c$ relative to the moving fluid.

18.8 Euler Equations

The two-dimensional Euler equations have the same form as the compressible flow equations presented in Section 18.3, but with the addition of an energy equation for the general case where the equation of state is more complicated than (18.16):

$$q = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ E \end{bmatrix}, \quad f(q) = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ (E + p)u \end{bmatrix}, \quad g(q) = \begin{bmatrix} \rho v \\ \rho u v \\ \rho v^2 + p \\ (E + p)v \end{bmatrix}. \quad (18.45)$$

The equation of state for a γ -law polytropic gas is the obvious extension of (14.23). The total energy is the sum of internal and kinetic energy,

$$E = \frac{p}{\gamma - 1} + \frac{1}{2}\rho(u^2 + v^2). \quad (18.46)$$

The Jacobian matrix $f'(q)$ has the four eigenvalues

$$\lambda^1 = u - c, \quad \lambda^2 = u, \quad \lambda^3 = u, \quad \lambda^4 = u + c. \quad (18.47)$$

As in one dimension, the sound speed is $c = \sqrt{\gamma p / \rho}$. The eigenvectors are

$$r^{x1} = \begin{bmatrix} 1 \\ u - c \\ v \\ H - uc \end{bmatrix}, \quad r^{x2} = \begin{bmatrix} 1 \\ u \\ v \\ \frac{1}{2}(u^2 + v^2) \end{bmatrix}, \quad r^{x3} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ v \end{bmatrix}, \quad r^{x4} = \begin{bmatrix} 1 \\ u + c \\ v \\ H + uc \end{bmatrix}. \quad (18.48)$$

The eigenvalues and eigenvectors in the y -direction are similar, with the roles of u and v reversed.

18.8.1 The Plane-Wave Riemann Problem

Consider a Riemann problem in which the data varies only in x . For the one-dimensional Euler equations the density can be discontinuous across the contact discontinuity, as illustrated in Figure 14.1. In the two-dimensional extension, there can also be a jump in the transverse velocity v across the contact discontinuity, exactly as was illustrated for the two-dimensional shallow water equations in Figure 18.1. The jump in density and the jump in shear velocity are carried by two independent linearly degenerate waves that both travel at the same velocity. These two waves correspond to the two eigenvalues $\lambda^2 = \lambda^3 = u$ of the Jacobian matrix. (The two vectors r^{x2} and r^{x3} in (18.48) are just one possible basis for this two-dimensional eigenspace.)

This two-dimensional Riemann problem is easily solved based on the one-dimensional theory, just as in the case of the shallow water equations. We can follow the procedure of Section 14.11, ignoring the transverse velocity v , since the primitive variables u and p are still continuous across the contact discontinuity. We then introduce a jump in v from v_l to v_r at the contact discontinuity. Note that this also gives a jump in E in the eigenvector r^{x3} , since v comes into the equation of state (18.46).

18.8.2 Three-Dimensional Euler Equations

In three space dimensions the Euler equations are similar, but with the addition of a fifth equation for the conservation of momentum ρw in the z -direction, where w is the z -component

of velocity. The conserved quantities and fluxes are then

$$q = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ E \end{bmatrix}, \quad f(q) = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ (E+p)u \end{bmatrix}, \quad g(q) = \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ (E+p)v \end{bmatrix}, \quad h(q) = \begin{bmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ (E+p)w \end{bmatrix}. \quad (18.49)$$

The equation of state now includes the kinetic energy $\frac{1}{2}\rho(u^2 + v^2 + w^2) = \frac{1}{2}\rho\vec{u} \cdot \vec{u}$, where $\vec{u} = (u, v, w)$ is the velocity vector. In any arbitrary direction \vec{n} there are two nonlinear acoustic fields with eigenvalues $(\vec{n} \cdot \vec{u}) \pm c$, and three linearly degenerate fields with eigenvalue $\vec{n} \cdot \vec{u}$. These three fields correspond to jumps in the density (entropy waves), and jumps in the two transverse velocities (shear waves). For example, if $\vec{n} = (1, 0, 0)$, then we are looking in the x -direction and arbitrary jumps in ρ , v , and w across the contact discontinuity can all propagate with speed u .

18.9 Symmetry and Reduction of Dimension

For some problems we may be able to reduce the complexity of the numerical problem substantially by taking advantage of symmetry. For example, if we are solving a problem where the solution is known to be radially symmetric, then we should be able to rewrite the equations in polar or spherical coordinates, obtaining a system that reduces to a problem in the single space variable r . The transformed equations will typically involve *geometric source terms*.

For example, when rewritten in polar r - θ coordinates, the compressible flow equations (18.19) take the form

$$\frac{\partial}{\partial t} \begin{bmatrix} r\rho \\ r\rho U \\ r\rho V \end{bmatrix} + \frac{\partial}{\partial r} \begin{bmatrix} r\rho U \\ r\rho U^2 + p \\ r\rho UV \end{bmatrix} + \frac{1}{r} \frac{\partial}{\partial \theta} \begin{bmatrix} r\rho V \\ r\rho UV \\ r\rho V^2 + p \end{bmatrix} = 0, \quad (18.50)$$

where $U(r, \theta, t)$ is the velocity in the radial direction and $V(r, \theta, t)$ is the velocity in the θ -direction. If we assume that $V(r, \theta, t) \equiv 0$ and there is no variation in the θ -direction, then these equations reduce to the two equations

$$\begin{aligned} (r\rho)_t + (r\rho U)_r &= 0, \\ (r\rho U)_t + (r\rho U^2 + p)_r &= 0. \end{aligned} \quad (18.51)$$

This system can be rewritten as

$$\begin{aligned} \rho_t + (\rho U)_r &= -(\rho U)/r, \\ (\rho U)_t + (\rho U^2 + p)_r &= -(\rho U^2)/r, \end{aligned} \quad (18.52)$$

which has exactly the same form as the one-dimensional system of equations (2.38), but with the addition of a geometric source term on the right-hand side.

The full two- or three-dimensional Euler equations with radial symmetry yield

$$\begin{aligned}\rho_t + (\rho U)_r &= -\frac{\alpha}{r}(\rho U), \\ (\rho U)_t + (\rho U^2 + p)_r &= -\frac{\alpha}{r}(\rho U^2), \\ E_t + ((E + p)U)_r &= -\frac{\alpha}{r}((E + p)U),\end{aligned}\tag{18.53}$$

Copyright © 2002, Cambridge University Press.
All rights reserved. May not be reproduced in any form without permission from the publisher, except fair uses permitted under U.S. or applicable copyright law.

where $\alpha = 1$ in two dimensions and $\alpha = 2$ in three dimensions.

Even if the real problems of interest must be studied multidimensionally, radially symmetric solutions are very valuable in testing and validating numerical codes. A highly accurate solution to the one-dimensional problem can be computed on a fine grid and used to test solutions computed with the multidimensional solver. This is useful not only in checking that the code gives essentially the correct answer in at least some special cases, but also in determining whether the numerical method is isotropic or suffers from *grid-orientation effects* that lead to the results being better resolved in some directions than in others. See Section 21.7.1 for one such example.

The D -dimensional acoustics equations with radial symmetry reduce to

$$\begin{aligned}p_t + K_0 U_r &= -\frac{\alpha}{r}(K_0 U), \\ \rho_0 U_t + p_r &= 0,\end{aligned}\tag{18.54}$$

where again $\alpha = D - 1$.

Exercises

18.1. Consider the system $q_t + Aq_x + Bq_y = 0$ with

$$A = \begin{bmatrix} 3 & 1 \\ 1 & 3 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 2 \\ 2 & 0 \end{bmatrix}.$$

Show that these matrices are simultaneously diagonalizable, and determine the general solution to this system with arbitrary initial data. In particular, sketch how the solution evolves in the x - y plane with data

$$q^1(x, y, 0) = \begin{cases} 1 & \text{if } x^2 + y^2 \leq 1, \\ 0 & \text{otherwise,} \end{cases} \quad q^2(x, y, 0) \equiv 0.$$

- 18.2. (a) Suppose that A and B are both symmetric matrices. Show that the system $q_t + Aq_x + Bq_y = 0$ must then be hyperbolic.
 (b) The matrices A and B are *simultaneously symmetrizable* if there is an invertible matrix M such that $M^{-1}AM$ and $M^{-1}BM$ are both symmetric. Show that in this case the system $q_t + Aq_x + Bq_y = 0$ must be hyperbolic.
 (c) Show that the matrices in (18.25) for the linearized acoustics equations are simultaneously symmetrizable with a matrix M of the form $M = \text{diag}(d_1, 1, 1)$.

- 18.3. Consider the two-dimensional system $q_t + Aq_x + Bq_y = 0$ with matrices

$$A = \begin{bmatrix} 1 & 10 \\ 0 & 2 \end{bmatrix}, \quad B = \begin{bmatrix} 2 & 0 \\ 10 & 1 \end{bmatrix},$$

each of which is diagonalizable with real eigenvalues. Show, however, that this system is not hyperbolic. (See also Exercise 19.1.)

- 18.4. Determine the eigenvectors of the three-dimensional acoustics matrix \tilde{A} from (18.35).
18.5. Show that the three-dimensional system (2.115) of Maxwell's equations is hyperbolic.

Multidimensional Numerical Methods

In the remainder of the book we concentrate on finite volume methods for multidimensional hyperbolic equations. We will begin by considering uniform Cartesian grids in two dimensions, using the notation illustrated in Figure 19.1(a). The value Q_{ij}^n represents a cell average over the (i, j) grid cell at time t_n ,

$$Q_{ij}^n \approx \frac{1}{\Delta x \Delta y} \int_{y_{j-1/2}}^{y_{j+1/2}} \int_{x_{i-1/2}}^{x_{i+1/2}} q(x, y, t_n) dx dy. \quad (19.1)$$

As in one dimension, we can use the integral form of the equations to determine how this cell average varies with time, and develop finite volume methods based on numerical approximations to the fluxes at each cell edge. Various approaches to doing this are summarized starting in Section 19.2.

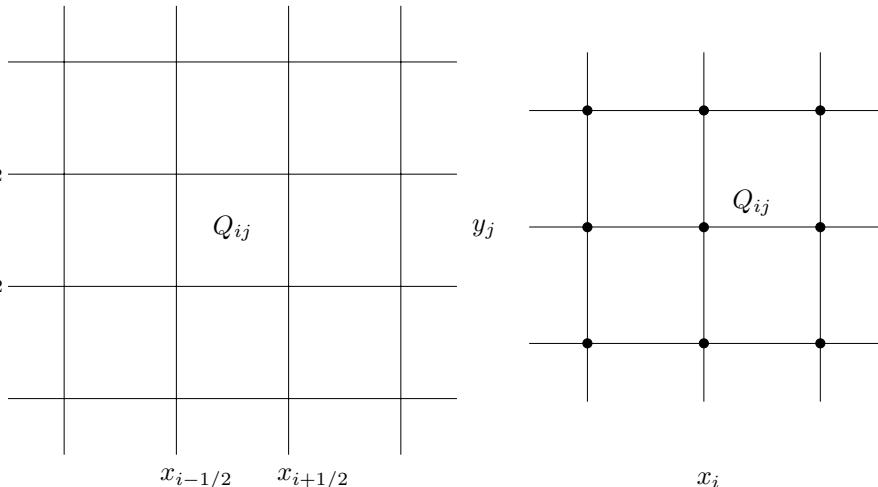
19.1 Finite Difference Methods

Rather than working with the cell averages and the integral form of the equations, one could instead view Q_{ij}^n as a pointwise approximation to the value of $q(x_i, y_j, t_n)$ at the point indicated in Figure 19.1(b). Discretizing the differential equations by finite differences then gives a finite difference method. As in one dimension, this approach often gives methods that look very similar to related finite volume methods. We will concentrate on finite volume methods, since this viewpoint allows the derivation of methods that are more robust when discontinuities are present, as well as being exactly conservative. However, it is sometimes useful to think of the methods in terms of their finite difference interpretation, in particular when computing the local truncation error by comparing with a Taylor series expansion of the true solution at the point (x_i, y_j) . The flux differences arising in a finite volume method are often seen to give approximations to terms in this Taylor expansion, at least when applied to smooth solutions.

19.1.1 Taylor Series Expansion of the Exact Solution

We develop the Taylor series expansion of the exact solution at a point after a single time step for the constant-coefficient linear hyperbolic system

$$q_t + Aq_x + Bq_y = 0. \quad (19.2)$$



Copyright © 2002, Cambridge University Press.
All rights reserved. May not be reproduced in any form without permission from the publisher, except fair uses permitted under U.S. or applicable copyright law.

Fig. 19.1. (a) Finite volume grid in two space dimensions, where Q_{ij} represents a cell average.
(b) Finite difference grid in two dimensions, where Q_{ij} represents a pointwise value.

We will require higher-order time derivatives in this expansion, which are most easily found from the expression

$$\partial_t^j q = [-(A\partial_x + B\partial_y)]^j q. \quad (19.3)$$

So, in particular,

$$q_{tt} = A^2 q_{xx} + AB q_{yx} + BA q_{xy} + B^2 q_{yy}. \quad (19.4)$$

Note that although $q_{yx} = q_{xy}$, in general $AB \neq BA$, and so we cannot combine the middle two terms.

The Taylor series expansion at (x_i, y_j) after time Δt can be written as

$$\begin{aligned} q(x_i, y_j, t_n + \Delta t) &= q + \Delta t q_t + \frac{1}{2} \Delta t^2 q_{tt} + \dots \\ &= q - \Delta t (A q_x + B q_y) \\ &\quad + \frac{1}{2} \Delta t^2 (A^2 q_{xx} + AB q_{yx} + BA q_{xy} + B^2 q_{yy}) + \dots \end{aligned} \quad (19.5)$$

Note that if A and B vary with x and y , then (19.3) still holds, but (19.4) becomes

$$q_{tt} = A(A q_x)_x + A(B q_y)_x + B(A q_x)_y + B(B q_y)_y. \quad (19.6)$$

The expansion becomes somewhat more complicated, as would the Lax–Wendroff method developed in the next subsection.

19.1.2 The Lax–Wendroff Method

The Lax–Wendroff method for the equation (19.2) is obtained by keeping only the terms shown in (19.5) and replacing the derivatives by centered finite differences, just as in

one space dimension. The purely one-dimensional derivatives are approximated as in one dimension, e.g.,

$$q_{yy} = \frac{1}{\Delta y^2} (Q_{i,j-1}^n - 2Q_{ij}^n + Q_{i,j+1}^n).$$

In addition, there are now cross-derivative terms, which can also be approximated to second-order with centered differences,

$$q_{xy} = q_{yx} \approx \frac{1}{4 \Delta x \Delta y} [(Q_{i+1,j+1}^n - Q_{i-1,j+1}^n) - (Q_{i+1,j-1}^n - Q_{i-1,j-1}^n)]. \quad (19.7)$$

Using these approximations in (19.5) gives

$$\begin{aligned} Q_{ij}^{n+1} &= Q_{ij}^n - \frac{\Delta t}{2 \Delta x} A (Q_{i+1,j}^n - Q_{i-1,j}^n) - \frac{\Delta t}{2 \Delta y} B (Q_{i,j+1}^n - Q_{i,j-1}^n) \\ &\quad + \frac{\Delta t^2}{2 \Delta x^2} A^2 (Q_{i+1,j}^n - 2Q_{ij}^n + Q_{i-1,j}^n) + \frac{\Delta t^2}{2 \Delta y^2} B^2 (Q_{i,j+1}^n - 2Q_{ij}^n + Q_{i,j-1}^n) \\ &\quad + \frac{\Delta t^2}{8 \Delta x \Delta y} (AB + BA) [(Q_{i+1,j+1}^n - Q_{i-1,j+1}^n) - (Q_{i+1,j-1}^n - Q_{i-1,j-1}^n)]. \end{aligned} \quad (19.8)$$

This has a nine-point stencil involving all nine of the points in Figure 19.1(b) in the update for Q_{ij} . In Section 19.3.1 we will see that this same method can be viewed as a finite volume method for updating the cell average Q_{ij} shown in Figure 19.1(a), resulting from defining numerical fluxes at the four edges of the cell in a natural way, based on the nine nearby cell values.

19.2 Finite Volume Methods and Approaches to Discretization

As in one space dimension, the two-dimensional Lax–Wendroff method suffers from problems with numerical dispersion, leading to phase errors and to unphysical oscillations in problems with discontinuities or steep gradients. By reinterpreting this method as a finite volume method and introducing upwind biasing and flux limiting, we will see that it can be greatly improved. We will also write the resulting methods in a more general form that applies also to nonlinear conservation laws and to variable-coefficient problems, by using the waves resulting from solving Riemann problems at each cell edge as the basis for upwinding and limiting.

This goal is pursued starting in the next chapter, where we begin to focus on this approach to developing multidimensional high-resolution methods. This is certainly not the only approach, however, and a wide variety of other multidimensional algorithms have been developed and successfully used. The remainder of this chapter is devoted to introducing some general notions of multidimensional finite volume methods.

Three of the most popular general approaches to obtaining multidimensional methods are:

- *Fully discrete flux-differencing methods.* In Section 19.3 we will see that the Lax-Wendroff method can be rewritten as a flux-differencing method. A numerical flux at each edge of the grid cell is defined, based on the data at the beginning of the time step. Differencing these fluxes gives the update to the cell average over a time step. To obtain better than first-order accuracy it is necessary to use the Taylor series expansion developed above in defining these fluxes. To obtain high-resolution nonoscillatory results it is also necessary to introduce limiters. There are many ways in which this can be done, and one particular method of this form is developed in Chapters 20 and 21.
- *Semidiscrete methods with Runge–Kutta time stepping.* Rather than using the Taylor series to replace time derivatives by spatial derivatives, we can focus on obtaining good accuracy of the flux at one instant in time and then use a Runge–Kutta method to perform the time stepping. This approach was introduced in one space dimension in Section 10.4. The two-dimensional extension is briefly discussed in Section 19.4.
- *Dimensional splitting.* By far the simplest approach to obtaining a multidimensional method is to apply a fractional-step method to split a multidimensional problem up into a sequence of one-dimensional problems. To solve $q_t + f(q)_x + g(q)_y = 0$, for example, we might alternate between solving $q_t + f(q)_x = 0$ and $q_t + g(q)_y = 0$, similarly to the way fractional-step methods are used for handling source terms, as discussed in Chapter 17. This approach, which is often surprisingly effective, is discussed in Section 19.5.

We will concentrate on methods that use one-dimensional Riemann solvers as a basic tool in the determination of interface fluxes. This is possible because the local problem at the edge of a grid cell is essentially one-dimensional in the direction normal to the edge. In order to obtain better than first-order accuracy it is necessary to bring in more multidimensional information, as is clear from the Taylor series expansion (19.5), but this can be done in various ways while still only using one-dimensional Riemann solutions.

One particular approach is developed starting in Chapter 20, in the form of the wave-propagation algorithms developed in [282], [283], and [257]. These algorithms take the form of fully discrete flux-differencing methods and are implemented in CLAWPACK in such a way that they can be applied to a wide variety of hyperbolic problems. Some related methods of this form can be found, for example, in [22], [24], [80], [100], [389].

Other methods have been developed that are based on a full decomposition of the data into multidimensional waves, rather than relying on one-dimensional Riemann solvers. For some examples, see [3], [52], [105], [106], [134], [135], [199], [342], [414], [415], [428].

19.3 Fully Discrete Flux-Differencing Methods

In deriving the two-dimensional conservation law $q_t + f(q)_x + g(q)_y = 0$ in Section 18.1, we considered an arbitrary region Ω . Now consider the special case where Ω is a rectangular grid cell of the form $C_{ij} = [x_{i-1/2}, x_{i+1/2}] \times [y_{j-1/2}, y_{j+1/2}]$, as shown in Figure 19.1, where $x_{i+1/2} - x_{i-1/2} = \Delta x$ and $y_{j+1/2} - y_{j-1/2} = \Delta y$. In this special case, the formula (18.7) simplifies, since the normal vector always points in either the x - or the y -direction. The normal flux is given by $f(q)$ along the left and right edges and by $g(q)$ along the top and

bottom. Integrating around the edges as required in (18.7) then gives

$$\frac{d}{dt} \iint_{C_{ij}} q(x, y, t) dx dy = \int_{y_{j-1/2}}^{y_{j+1/2}} f(q(x_{i+1/2}, y, t)) dy - \int_{y_{j-1/2}}^{y_{j+1/2}} f(q(x_{i-1/2}, y, t)) dy + \int_{x_{i-1/2}}^{x_{i+1/2}} g(q(x, y_{j+1/2}, t)) dx - \int_{x_{i-1/2}}^{x_{i+1/2}} g(q(x, y_{j-1/2}, t)) dx. \quad (19.9)$$

Copyright © 2002, Cambridge University Press.
All rights reserved. May not be reproduced in any form without permission from the publisher, except fair uses permitted under U.S. or applicable copyright law.

If we integrate this expression from t_n to t_{n+1} and divide by the cell area $\Delta x \Delta y$, we are led to a fully discrete flux-differencing method of the form

$$Q_{ij}^{n+1} = Q_{ij}^n - \frac{\Delta t}{\Delta x} [F_{i+1/2,j}^n - F_{i-1/2,j}^n] - \frac{\Delta t}{\Delta y} [G_{i,j+1/2}^n - G_{i,j-1/2}^n], \quad (19.10)$$

where

$$F_{i-1/2,j}^n \approx \frac{1}{\Delta t \Delta y} \int_{t_n}^{t_{n+1}} \int_{y_{j-1/2}}^{y_{j+1/2}} f(q(x_{i-1/2}, y, t)) dy dt, \\ G_{i,j-1/2}^n \approx \frac{1}{\Delta t \Delta x} \int_{t_n}^{t_{n+1}} \int_{x_{i-1/2}}^{x_{i+1/2}} g(q(x, y_{j-1/2}, t)) dx dt. \quad (19.11)$$

The numerical fluxes F^n and G^n at each edge are typically computed from the data Q^n at the initial time. (As in one dimension, these methods can also be extended to nonconservative hyperbolic systems, see Section 19.3.3.)

For the linear system $q_t + Aq_x + Bq_y = 0$ we can obtain approximations to these interface fluxes by using the Taylor expansion (19.5), which can be rewritten as

$$q(x, y, t_n + \Delta t) = q - \Delta t \left(Aq - \frac{\Delta t}{2} A^2 q_x - \frac{\Delta t}{2} AB q_y \right)_x - \Delta t \left(Bq - \frac{\Delta t}{2} B^2 q_y - \frac{\Delta t}{2} BA q_x \right)_y + \dots \quad (19.12)$$

This suggests that we need

$$F_{i-1/2,j} \approx Aq(x_{i-1/2}, y_j, t_n) - \frac{\Delta t}{2} A^2 q_x(x_{i-1/2}, y_j, t_n) - \frac{\Delta t}{2} AB q_y(x_{i-1/2}, y_j, t_n), \\ G_{i,j-1/2} \approx Bq(x_i, y_{j-1/2}, t_n) - \frac{\Delta t}{2} B^2 q_y(x_i, y_{j-1/2}, t_n) - \frac{\Delta t}{2} BA q_x(x_i, y_{j-1/2}, t_n). \quad (19.13)$$

It can be shown that for this problem these expressions agree with the integrals in (19.11) to $\mathcal{O}(\Delta t^2)$.

19.3.1 Flux-Differencing Form of the Lax–Wendroff Method

The Lax–Wendroff method (19.8) for the constant-coefficient linear system $q_t + Aq_x + Bq_y = 0$ can be interpreted as a method of the form (19.10), where the fluxes are given by

$$\begin{aligned} F_{i-1/2,j} &= \frac{1}{2}A(Q_{i-1,j} + Q_{ij}) - \frac{\Delta t}{2\Delta x}A^2(Q_{ij} - Q_{i-1,j}) \\ &\quad - \frac{\Delta t}{8\Delta y}AB[(Q_{i,j+1} - Q_{ij}) + (Q_{i-1,j+1} - Q_{i-1,j}) \\ &\quad + (Q_{ij} - Q_{i,j-1}) + (Q_{i-1,j} - Q_{i-1,j-1})], \\ G_{i,j-1/2} &= \frac{1}{2}B(Q_{i,j-1} + Q_{ij}) - \frac{\Delta t}{2\Delta y}B^2(Q_{ij} - Q_{i,j-1}) \\ &\quad - \frac{\Delta t}{8\Delta x}BA[(Q_{i+1,j} - Q_{ij}) + (Q_{i+1,j-1} - Q_{i,j-1}) \\ &\quad + (Q_{ij} - Q_{i-1,j}) + (Q_{i,j-1} - Q_{i-1,j-1})]. \end{aligned} \quad (19.14)$$

These fluxes relate directly to (19.13). Note in particular that the expression ABq_y in (19.13), for example, is approximated by

$$\begin{aligned} ABq_y(x_{i-1/2}, y_j, t_n) &\approx \frac{1}{4\Delta y}[AB(Q_{i,j+1} - Q_{ij}) + AB(Q_{i-1,j+1} - Q_{i-1,j}) \\ &\quad + AB(Q_{ij} - Q_{i,j-1}) + AB(Q_{i-1,j} - Q_{i-1,j-1})]. \end{aligned} \quad (19.15)$$

In Chapters 20 and 21 we will see how this method can be greatly improved by introducing an upwind bias and flux limiting into the formulas.

19.3.2 Godunov's Method

For a general conservation law, the simplest flux-differencing method of the form (19.10) is Godunov's method. A natural two-dimensional generalization of the method developed in Section 15.1 is obtained by simply solving the normal Riemann problem at each cell edge to find the value Q^ψ that propagates with speed 0, and then evaluating the appropriate flux function at this value to obtain the numerical flux at this edge. This gives

$$\begin{aligned} F_{i-1/2,j} &= f(Q_{i-1/2,j}^\psi), \\ G_{i,j-1/2} &= g(Q_{i,j-1/2}^\psi), \end{aligned} \quad (19.16)$$

where $Q_{i-1/2,j}^\psi$ is obtained by solving the Riemann problem for $q_t + f(q)_x = 0$ with data $Q_{i-1,j}$ and Q_{ij} , while $Q_{i,j-1/2}^\psi$ is obtained by solving the Riemann problem for $q_t + g(q)_y = 0$ with data $Q_{i,j-1}$ and Q_{ij} . As in one dimension, approximate Riemann solvers can be used in place of the exact Riemann solution.

For a linear system of equations with $f(q) = Aq$ and $g(q) = Bq$, we denote the eigenvector matrices for A and B by R^x and R^y , respectively, and the eigenvalue matrices by Λ^x and Λ^y , as in Section 18.5. We can then define matrices A^\pm and B^\pm analogous to

(4.45) by

$$A^\pm = R^x(\Lambda^x)^\pm(R^x)^{-1}, \quad B^\pm = R^y(\Lambda^y)^\pm(R^y)^{-1}. \quad (19.17)$$

In terms of this notation, we find that the Godunov fluxes for a linear problem are the natural generalization of the one-dimensional flux (4.56),

$$\begin{aligned} F_{i-1/2,j} &= A^+ Q_{i-1,j} + A^- Q_{ij}, \\ G_{i,j-1/2} &= B^+ Q_{i,j-1} + B^- Q_{ij}. \end{aligned} \quad (19.18)$$

This amounts to using only the first terms in the fluxes (19.13), and an upwind approximation to these. Of course, this method is only first-order accurate and moreover is typically stable only for Courant number up to 1/2 in two dimensions. This is illustrated for the advection equation in Section 20.4.

19.3.3 Fluctuation Form

As in one space dimension, we will develop finite volume methods in a more general form than the flux-differencing formula (19.10), so that they are also applicable to hyperbolic equations that are not in conservation form. To make this extension we will rewrite the method as follows, motivated by the one-dimensional method (15.62):

$$\begin{aligned} Q_{ij}^{n+1} &= Q_{ij} - \frac{\Delta t}{\Delta x} (\mathcal{A}^+ \Delta Q_{i-1/2,j} + \mathcal{A}^- \Delta Q_{i+1/2,j}) \\ &\quad - \frac{\Delta t}{\Delta y} (\mathcal{B}^+ \Delta Q_{i,j-1/2} + \mathcal{B}^- \Delta Q_{i,j+1/2}) \\ &\quad - \frac{\Delta t}{\Delta x} (\tilde{F}_{i+1/2,j} - \tilde{F}_{i-1/2,j}) - \frac{\Delta t}{\Delta y} (\tilde{G}_{i,j+1/2} - \tilde{G}_{i,j-1/2}). \end{aligned} \quad (19.19)$$

The term $\mathcal{A}^+ \Delta Q_{i-1/2,j}$, for example, represents the first-order Godunov update to the cell value Q_{ij} resulting from the Riemann problem at the edge $(i - 1/2, j)$. The other three similar terms are the Godunov updates resulting from the Riemann problems at the other three edges. For the linear system discussed above, these fluctuations are simply given by

$$\begin{aligned} \mathcal{A}^\pm \Delta Q_{i-1/2,j} &= A^\pm (Q_{ij} - Q_{i-1,j}), \\ \mathcal{B}^\pm \Delta Q_{i,j-1/2} &= B^\pm (Q_{ij} - Q_{i,j-1}). \end{aligned} \quad (19.20)$$

For Godunov's method we take $\tilde{F} = \tilde{G} = 0$ everywhere. Later the fluxes \tilde{F} and \tilde{G} will be used for correction terms, both those arising from introducing slopes as in one dimension to model the A^2 and B^2 terms in (19.13), and also new ones modeling the cross-derivative terms involving AB and BA in (19.13).

For a general nonlinear conservation law where the Godunov fluxes are defined by (19.16), we can set

$$\begin{aligned}\mathcal{A}^+ \Delta Q_{i-1/2,j} &= f(Q_{ij}) - f(Q_{i-1/2,j}^\psi), \\ \mathcal{A}^- \Delta Q_{i-1/2,j} &= f(Q_{i-1/2,j}^\psi) - f(Q_{i-1,j}), \\ \mathcal{B}^+ \Delta Q_{i,j-1/2} &= g(Q_{ij}) - g(Q_{i,j-1/2}^\psi), \\ \mathcal{B}^- \Delta Q_{i,j-1/2} &= g(Q_{i,j-1/2}^\psi) - g(Q_{i,j-1}).\end{aligned}\tag{19.21}$$

Godunov's method results from using these formulas in (19.19) and setting all $\tilde{F} = \tilde{G} = 0$. As in one dimension, the fluctuations $\mathcal{A}^\pm \Delta Q$ and $\mathcal{B}^\pm \Delta Q$ can also be computed in terms of the waves and speeds arising in the Riemann solution, using formulas analogous to (4.42).

19.4 Semidiscrete Methods with Runge–Kutta Time Stepping

As we see already from the Lax–Wendroff method with fluxes (19.14), obtaining even second-order accuracy with a flux-differencing method based on the time-integrated fluxes (19.11) can lead to complicated formulas. As discussed in Section 10.3 for one-dimensional problems, this Taylor series approach is not easily extended to obtain higher-order methods. For this reason, another popular approach is to proceed as in Section 10.4 and use the expression (19.9) to derive ordinary differential equations for the evolution of the cell averages

$$Q_{ij}(t) = \iint_{C_{ij}} q(x, y, t) dx dy.\tag{19.22}$$

This is accomplished by defining numerical flux functions

$$\begin{aligned}F_{i-1/2,j}(Q(t)) &\approx \frac{1}{\Delta y} \int_{y_{j-1/2}}^{y_{j+1/2}} f(q(x_{i-1/2}, y, t)) dy, \\ G_{i,j-1/2}(Q(t)) &\approx \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} g(q(x, y_{j-1/2}, t)) dx\end{aligned}\tag{19.23}$$

by some procedure based on the nearby cell averages at this instant in time. Then the system of ODEs

$$\begin{aligned}\frac{d}{dt} Q_{ij}(t) &= -\frac{1}{\Delta x} [F_{i+1/2,j}(Q(t)) - F_{i-1/2,j}(Q(t))] \\ &\quad - \frac{1}{\Delta y} [G_{i,j+1/2}(Q(t)) - G_{i,j-1/2}(Q(t))]\end{aligned}\tag{19.24}$$

is solved by an ODE method, typically a multistage Runge–Kutta method. In order to achieve high-order accuracy it is still necessary to use information from several grid cells nearby in defining the fluxes (19.23), typically by some multidimensional interpolation method. However, since we do not attempt to also approximate the time-derivative terms,

this may be relatively simple. To obtain a high-resolution method, it is necessary to include some form of upwinding and/or limiting in the process of approximating the flux, e.g., by a multidimensional version of the ENO method described in Section 10.4.4.

19.5 Dimensional Splitting

The easiest way to extend one-dimensional numerical methods to more space dimensions is to use *dimensional splitting*, an application of the fractional-step procedure discussed in Chapter 17. A multidimensional problem is simply split into a sequence of one-dimensional problems. This is easy to apply on a Cartesian grid aligned with the coordinate axes as shown in Figure 19.1.

For example, the two-dimensional linear problem

$$q_t + Aq_x + Bq_y = 0$$

can be split into

$$x\text{-sweeps : } q_t + Aq_x = 0, \quad (19.25)$$

$$y\text{-sweeps : } q_t + Bq_y = 0. \quad (19.26)$$

In the x -sweeps we start with cell averages Q_{ij}^n at time t_n and solve one-dimensional problems $q_t + Aq_x = 0$ along each row of cells \mathcal{C}_{ij} with j fixed, updating Q_{ij}^n to Q_{ij}^* :

$$Q_{ij}^* = Q_{ij}^n - \frac{\Delta t}{\Delta x} (F_{i+1/2,j}^n - F_{i-1/2,j}^n), \quad (19.27)$$

where $F_{i-1/2,j}^n$ is an appropriate numerical flux for the one-dimensional problem between cells $\mathcal{C}_{i-1,j}$ and \mathcal{C}_{ij} . In the y -sweeps we then use the Q_{ij}^* values as data for solving $q_t + Bq_y = 0$ along each column of cells with i fixed, which results in Q_{ij}^{n+1} :

$$Q_{ij}^{n+1} = Q_{ij}^* - \frac{\Delta t}{\Delta x} (G_{i,j+1/2}^* - G_{i,j-1/2}^*). \quad (19.28)$$

Note that there will generally be a splitting error (see Section 17.3) unless the operators $\mathcal{A} = A\partial_x$ and $\mathcal{B} = B\partial_y$ commute, i.e., unless $AB = BA$. Only in the case where the multidimensional problem decouples into scalar advection equations can we use dimensional splitting with no splitting error. Even in this case we must be careful with boundary conditions (see Section 17.9).

However, the splitting error is often no worse than the errors introduced by the numerical methods in each sweep, and dimensional splitting can be a very effective approach. It gives a simple and relatively inexpensive way to extend one-dimensional high-resolution methods to two or three dimensions.

Note that with the dimensional-splitting approach we do not explicitly model the cross-derivative terms involving q_{xy} in the Taylor series expansion (19.5). In each sweep we only model second derivatives in each coordinate direction, q_{xx} and q_{yy} , which appear in the one-dimensional algorithm. The q_{xy} term arises automatically through the fractional-step

procedure. The intermediate solution q^* resulting from x -sweeps involves terms modeling Aq_x . In the y -sweeps we compute terms modeling Bq_y^* , which thus model $B(Aq_x)_y$.

Instead of this *Godunov splitting*, one might instead use the *Strang splitting*

$$\begin{aligned} Q_{ij}^* &= Q_{ij}^n - \frac{\Delta t}{2 \Delta x} (F_{i+1/2,j}^n - F_{i-1/2,j}^n), \\ Q_{ij}^{**} &= Q_{ij}^* - \frac{\Delta t}{\Delta x} (G_{i,j+1/2}^* - G_{i,j-1/2}^*), \\ Q_{ij}^{n+1} &= Q_{ij}^n - \frac{\Delta t}{2 \Delta x} (F_{i+1/2,j}^{**} - F_{i-1/2,j}^{**}), \end{aligned} \quad (19.29)$$

as discussed in Section 17.4. With the Strang splitting we also obtain terms modeling $A(Bq_y)_x$ from the second x -sweep, which are also needed in the Taylor series expansion. Only in the constant-coefficient case with $AB = BA$ does the Godunov splitting give a fully second-order accurate method. However, in practice there is often very little difference in results obtained with the two approaches, as is also the case for other fractional-step methods as discussed in Section 17.5.

In fact, if the Strang splitting is implemented as in (19.29), it may give worse results than the Godunov splitting, because the x -sweeps are taken with time step $\Delta t/2$ and hence will typically have Courant number less than 1/2. This introduces more numerical smearing as well as more work. If instead the Strang splitting is implemented by simply alternating the order in which x -sweeps and y -sweeps are performed, then this is avoided (see Section 17.4). However, this is somewhat harder to implement in connection with variable-size time steps, and in CLAWPACK the form (19.29) is implemented as the Strang splitting, though the Godunov splitting is generally recommended instead.

19.5.1 CLAWPACK Implementation

In order to apply the fractional-step approach, we need to be able to solve each of the one-dimensional equations (19.25) and (19.26), and hence must have two different Riemann solvers available. For many physical systems the equations and solution procedure are essentially the same in each direction. For example, the acoustics equations (18.25) in an isotropic medium have exactly the same form when restricted to plane waves in x or y , but with the roles of u and v reversed. The same is true for the shallow water equations (18.38) and the Euler equations (18.45). For this reason it is often simplest to write a single Riemann solver with a flag indicating the desired direction. This convention is used in CLAWPACK, where a single subroutine `rpn2` must be provided that solves the Riemann problem normal to edges of grid cells along one slice of the domain. The flag `ixy` indicates whether the slice is in the x -direction or the y -direction. The other parameters of this routine are identical to the parameters appearing in the one-dimensional Riemann solver `rp1`.

The two-dimensional CLAWPACK code also allows a second Riemann solver `rpt2` to be provided. This must solve a different sort of Riemann problem in the transverse direction, as described in Section 21.3, and is not used in the dimensional-splitting algorithm.

Dimensional splitting is invoked in CLAWPACK by setting `method(3) = -1` for the Godunov splitting, which is generally recommended, or `method(3) = -2` for Strang

splitting. Positive values of `method(3)` instead invoke the unsplit methods described in later chapters. (See Section 21.3.)

Exercise

- 19.1. The system given in Exercise 18.3 is hyperbolic in x and y separately, and so we can apply dimensional splitting to attempt to solve this nonhyperbolic system. Implement this in CLAWPACK, and analyze what happens.

Multidimensional Scalar Equations

In this chapter we will develop high-resolution methods for scalar hyperbolic equations in two space dimensions. We begin by considering the constant-coefficient advection equation and show how the waves obtained by solving the Riemann problem at each cell interface can be naturally used to define high-resolution fluxes. We then extend these methods to variable-coefficient advection and nonlinear scalar conservation laws. In the next chapter they are extended further to hyperbolic systems of equations.

We first consider the scalar advection equation

$$q_t + u q_x + v q_y = 0, \quad (20.1)$$

with u and v constant. In the figures illustrating these methods we will generally assume $u > 0$ and $v > 0$, and this case will sometimes be assumed when we wish to be specific, but most of the formulas will be presented in a manner that applies for flow in any direction. The notation u^\pm and v^\pm meaning the positive or negative part of the velocity will frequently be used, with the definition (4.40).

The true solution for this equation is simply $q(x, y, t) = \hat{q}(x - ut, y - vt)$, but for our present purposes the Taylor series expansion is more illuminating:

$$\begin{aligned} q(x, y, t_{n+1}) &= q(x, y, t_n) + \Delta t q_t(x, y, t_n) + \frac{1}{2}(\Delta t)^2 q_{tt}(x, y, t_n) + \dots \\ &= q(x, y, t_n) - u \Delta t q_x - v \Delta t q_y \\ &\quad + \frac{1}{2}(\Delta t)^2 [u^2 q_{xx} + vu q_{xy} + uv q_{yx} + v^2 q_{yy}] + \dots \end{aligned} \quad (20.2)$$

This comes from (19.5) with $A = u$ and $B = v$, and will be useful in identifying terms arising in finite volume approximations to the advection equation.

20.1 The Donor-Cell Upwind Method for Advection

The simplest finite volume method for the advection equation is the first-order upwind method, which takes the general form

$$\begin{aligned} Q_{ij}^{n+1} &= Q_{ij} - \frac{\Delta t}{\Delta x} [u^+(Q_{ij} - Q_{i-1,j}) + u^-(Q_{i+1,j} - Q_{ij})] \\ &\quad - \frac{\Delta t}{\Delta y} [v^+(Q_{ij} - Q_{i,j-1}) + v^-(Q_{i,j+1} - Q_{ij})]. \end{aligned} \quad (20.3)$$

This uses an upwind approximation to the derivatives q_x and q_y in the $\mathcal{O}(\Delta t)$ terms of the Taylor series expansion (20.2). This method has the form (19.19) with $\tilde{F} = \tilde{G} = 0$ and fluctuations

$$\begin{aligned}\mathcal{A}^\pm \Delta Q_{i-1/2,j} &= u^\pm(Q_{ij} - Q_{i-1,j}), \\ \mathcal{B}^\pm \Delta Q_{i,j-1/2} &= v^\pm(Q_{ij} - Q_{i,j-1}),\end{aligned}\tag{20.4}$$

which are a special case of (19.20).

This upwind method agrees with Godunov's method as described in Section 19.3.2 for this scalar equation. The fluxes for the method (20.3) are

$$\begin{aligned}F_{i-1/2,j} &= u^+Q_{i-1,j} + u^-Q_{ij}, \\ G_{i,j-1/2} &= v^+Q_{i,j-1} + v^-Q_{ij},\end{aligned}\tag{20.5}$$

which agree with the Godunov fluxes (19.18). In this case each Riemann solution consists of a single wave carrying the jump in Q between the neighboring two grid cells, propagating at speed u horizontally or at speed v vertically depending on the orientation of the two cells. The value of Q^\vee at each interface depends on whether the relevant velocity is positive or negative.

The first-order accurate method (20.3) for the advection equation is often called the *donor-cell upwind* (DCU) method. Each flux in (20.5) approximates the amount of q flowing normal to the edge, assuming that the only contribution to this flux is from the adjacent cell on the upwind side (the donor cell). This is indicated schematically in Figure 20.1(a) for the case $u, v > 0$.

Note that the updated value Q_{ij}^{n+1} depends on only the three values Q_{ij} , $Q_{i-1,j}$, and $Q_{i,j-1}$. This is clearly not correct, since the flow is really at an angle to the grid, as indicated in Figure 20.1(b), and the value $Q_{i-1,j-1}$ should also affect Q_{ij}^{n+1} . The CFL condition (Section 4.4) suggests that this may cause stability problems, and indeed this method does not have the best possible stability properties. It will be shown in Section 20.4 that this

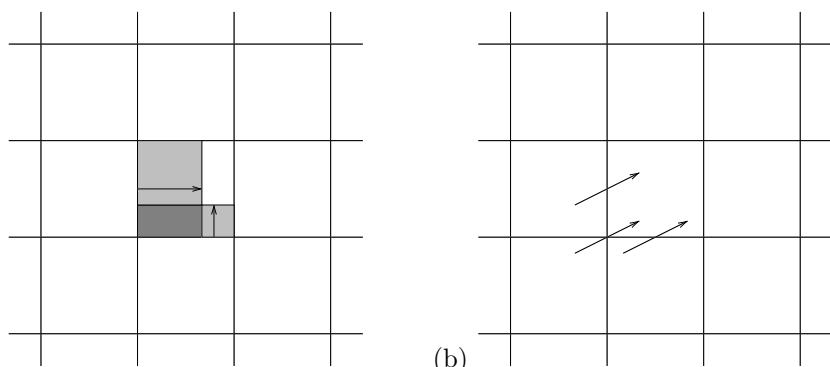


Fig. 20.1. (a) Waves moving normal to the cell interfaces give the updates for the DCU method. (b) The true velocity (u, v) is at an angle to the grid, and information from cell $(i-1, j-1)$ should also affect the new value in cell (i, j) . This corner coupling is missing in the DCU method.

method is stable only for Δt small enough that

$$\left| \frac{u \Delta t}{\Delta x} \right| + \left| \frac{v \Delta t}{\Delta y} \right| \leq 1. \quad (20.6)$$

An improved upwind method is developed in Section 20.2, which takes account of the flow direction more fully and has the stability bound

$$\max \left(\left| \frac{u \Delta t}{\Delta x} \right|, \left| \frac{v \Delta t}{\Delta y} \right| \right) \leq 1. \quad (20.7)$$

This is better than (20.6) whenever u and v are both nonzero, i.e., when flow is at an angle to the grid.

20.2 The Corner-Transport Upwind Method for Advection

For the advection equation, a better first-order accurate upwind method can be derived by taking the reconstruct–evolve–average approach of Algorithm 4.1, extended in the obvious way to two space dimensions:

- View the cell averages at time t_n as defining a piecewise constant function $\tilde{q}^n(x, y, t_n)$ with constant value Q_{ij}^n in cell C_{ij} ,
- Evolve the advection equation exactly with this data over time Δt ,
- Average the resulting solution $\tilde{q}^n(x, y, t_{n+1})$ back onto the grid.

For the constant-coefficient advection equation this is easily done, since

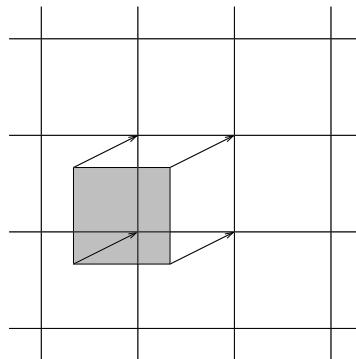
$$\tilde{q}^n(x, y, t_{n+1}) = \tilde{q}^n(x - u \Delta t, y - v \Delta t, t_n).$$

The exact solution is the same piecewise constant function, simply shifted by $(u \Delta t, v \Delta t)$. So we find that

$$\begin{aligned} Q_{ij}^{n+1} &= \frac{1}{\Delta x \Delta y} \int_{x_{i-1/2}}^{x_{i+1/2}} \int_{y_{j-1/2}}^{y_{j+1/2}} \tilde{q}^n(x - u \Delta t, y - v \Delta t, t_n) dx dy \\ &= \frac{1}{\Delta x \Delta y} \int_{x_{i-1/2}-u \Delta t}^{x_{i+1/2}-u \Delta t} \int_{y_{j-1/2}-v \Delta t}^{y_{j+1/2}-v \Delta t} \tilde{q}^n(x, y, t_n) dx dy. \end{aligned} \quad (20.8)$$

The new cell average Q_{ij}^{n+1} is given by the cell average of $\tilde{q}^n(x, y, t_n)$ over the shaded region shown in Figure 20.2. Since $\tilde{q}^n(x, y, t_n)$ is constant in each grid cell, this reduces to a simple convex combination of four cell values:

$$\begin{aligned} Q_{ij}^{n+1} &= \frac{1}{\Delta x \Delta y} [(\Delta x - u \Delta t)(\Delta y - v \Delta t)Q_{ij}^n + (\Delta x - u \Delta t)(v \Delta t)Q_{i,j-1}^n \\ &\quad + (\Delta y - v \Delta t)(u \Delta t)Q_{i-1,j}^n + (u \Delta t)(v \Delta t)Q_{i-1,j-1}^n]. \end{aligned} \quad (20.9)$$



Copyright © 2002, Cambridge University Press.
All rights reserved. May not be reproduced in any form without permission from the publisher, except fair uses permitted under U.S. or applicable copyright law.

Fig. 20.2. The corner-transport upwind method is obtained by shifting the piecewise constant data by distance $(u \Delta t, v \Delta t)$ and averaging back on the grid. Alternatively, the new value Q_{ij}^{n+1} is determined by averaging the piecewise constant function over the shaded region shown in the figure.

This can be rearranged to yield

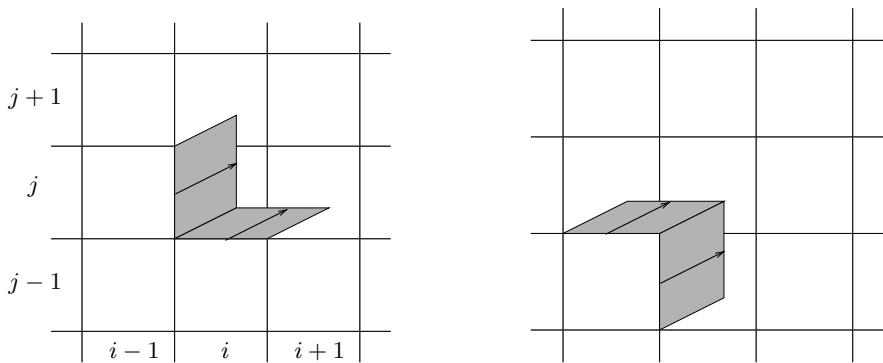
$$\begin{aligned} Q_{ij}^{n+1} = & Q_{ij} - \frac{u \Delta t}{\Delta x} (Q_{ij} - Q_{i-1,j}) - \frac{v \Delta t}{\Delta y} (Q_{ij} - Q_{i,j-1}) \\ & + \frac{1}{2} (\Delta t)^2 \left\{ \frac{u}{\Delta x} \left[\frac{v}{\Delta y} (Q_{ij} - Q_{i,j-1}) - \frac{v}{\Delta y} (Q_{i-1,j} - Q_{i-1,j-1}) \right] \right. \\ & \left. + \frac{v}{\Delta y} \left[\frac{u}{\Delta x} (Q_{ij} - Q_{i-1,j}) - \frac{u}{\Delta x} (Q_{i,j-1} - Q_{i-1,j-1}) \right] \right\}. \quad (20.10) \end{aligned}$$

The top line of this expression corresponds to the donor-cell upwind method. The additional terms can be arranged in several different ways. They have been displayed here in a manner that relates directly to the Taylor series expansion (20.2). We see that the final term in (20.10) models the cross-derivative terms $uvq_{yx} + vuq_{xy}$ in the $\mathcal{O}((\Delta t)^2)$ term of that expansion.

The method (20.10) is often called *corner-transport upwind* (CTU) method (following Colella [80]), since it includes the proper transport across the corner from cell $C_{i-1,j-1}$ to C_{ij} . It is still only first-order accurate, for two reasons: It is missing approximations to the q_{xx} and q_{yy} terms in (20.2), and the approximations to uq_x and vq_y terms are only first-order one-sided approximations. Both of these deficiencies can be addressed by introducing slopes in the x - and y -directions separately, just as we did in one dimension. Consequently a high-resolution version of this algorithm is easy to construct, as we will do in Section 20.6. See [25], [470] for discussion of some related algorithms.

20.3 Wave-Propagation Implementation of the CTU Method

Before discussing high-resolution corrections, we will develop a different implementation of the CTU method that will be much easier to extend to variable-coefficient advection and to other hyperbolic systems. Figure 20.3 shows the basis for a wave-propagation view of this method, in which all waves propagate at velocity (u, v) in the correct physical direction. To be specific we will continue to assume $u > 0$ and $v > 0$ in the figures and formulas in this section. More general formulas are given in the next section.

Copyright © 2002, Cambridge University Press.
All rights reserved. May not be reproduced in any form without permission from the publisher, except fair uses permitted under U.S. or applicable copyright law.Fig. 20.3. (a) Transverse propagation affecting the fluxes $\tilde{F}_{i+1/2,j}$ and $\tilde{G}_{i,j+1/2}$. (b) Transverse propagation affecting the fluxes $\tilde{F}_{i-1/2,j}$ and $\tilde{G}_{i,j-1/2}$.

From the interface between cells $\mathcal{C}_{i-1,j}$ and \mathcal{C}_{ij} , for example, there is a wave that propagates into cells \mathcal{C}_{ij} and $\mathcal{C}_{i,j+1}$. The jump across this wave is $Q_{ij} - Q_{i-1,j}$, and this increment affects both cell averages Q_{ij} and $Q_{i,j+1}$. As Figure 20.3 shows, there are four distinct waves that affect Q_{ij}^n . The effect of each wave is to modify the cell average by the jump across the wave multiplied by the fraction of the cell covered by the wave. These fractions are easily worked out, noting that the small triangular portion of each wave moving transversely into neighboring cells has area $\frac{1}{2}(u \Delta t)(v \Delta t) = \frac{1}{2}uv(\Delta t)^2$. The wave from interface $(i-1/2, j)$, for example, modifies Q_{ij} by

$$\left(\frac{u \Delta t \Delta y - \frac{1}{2}uv(\Delta t)^2}{\Delta x \Delta y} \right) (Q_{ij} - Q_{i-1,j}) \quad (20.11)$$

and modifies $Q_{i,j+1}$ by

$$\left(\frac{\frac{1}{2}uv(\Delta t)^2}{\Delta x \Delta y} \right) (Q_{ij} - Q_{i-1,j}). \quad (20.12)$$

Note that the update (20.11) is present in the formula (20.10), split into two parts. The latter part, corresponding to the triangular piece, is grouped with three other term corresponding to the triangular pieces of the other three waves shown in Figure 20.3 that affect this cell.

Thus we can view (20.10) as consisting of the DCU method (the first line), in which waves simply move normal to the cell interfaces as shown in Figure 20.1, combined with a set of corrections for the transverse propagation of the waves. These corrections can be viewed as fluxes through edges of the cell. The triangular region in cell $(i, j+1)$ of Figure 20.3(a) has been transferred from cell (i, j) and hence corresponds to a flux through edge $(i, j+1/2)$. This transfer can be represented by a flux $\tilde{G}_{i,j+1/2}$. Taking this viewpoint, we find that we can rewrite (20.10) in the form (19.19) by defining $\mathcal{A}^\pm \Delta Q$ and $\mathcal{B}^\pm \Delta Q$ as in (20.4) and the

correction fluxes as

$$\begin{aligned}\tilde{F}_{i-1/2,j} &= -\frac{1}{2} \frac{\Delta t}{\Delta y} uv(Q_{i-1,j} - Q_{i-1,j-1}), \\ \tilde{F}_{i+1/2,j} &= -\frac{1}{2} \frac{\Delta t}{\Delta y} uv(Q_{ij} - Q_{i,j-1}), \\ \tilde{G}_{i,j-1/2} &= -\frac{1}{2} \frac{\Delta t}{\Delta x} uv(Q_{i,j-1} - Q_{i-1,j-1}), \\ \tilde{G}_{i,j+1/2} &= -\frac{1}{2} \frac{\Delta t}{\Delta x} uv(Q_{ij} - Q_{i-1,j}).\end{aligned}\tag{20.13}$$

Copyright © 2002, Cambridge University Press.
All rights reserved. May not be reproduced in any form without permission from the publisher, except fair uses permitted under U.S. or applicable copyright law.

The CTU method has better stability properties than DCU. From the interpretation of the algorithm given at the beginning of Section 20.2, we expect the method to be stable for any time step for which the piecewise constant function does not shift more than one grid cell in the time step. This gives the stability bound (20.7), and the method is stable for Courant numbers up to 1. This can be shown formally in the 1-norm by extending the proof of Section 8.3.4 to two dimensions using the convex combination (20.9). Stability in the 2-norm is demonstrated using von Neumann analysis in the next section.

20.4 von Neumann Stability Analysis

For constant-coefficient linear equations, von Neumann analysis is often the easiest way to determine stability bounds, as discussed in Section 8.3.3 in one space dimension. As two-dimensional examples, in this section we consider the DCU method (20.3) and the CTU method (20.10). Similar analysis can also be performed for the Lax–Wendroff method or the wave-propagation version (introduced in Section 20.6) provided that no limiter function is applied. (Applying limiters makes the method nonlinear and von Neumann analysis can no longer be used.) See [282] for stability analyses of these cases and three-dimensional generalizations, and also [202].

To be specific we will assume $u, v > 0$, although similar analysis applies to other choices of signs. Then the DCU method (20.3) becomes

$$Q_{IJ}^{n+1} = Q_{IJ}^n - v^x (Q_{IJ}^n - Q_{I-1,J}^n) - v^y (Q_{IJ}^n - Q_{I,J-1}^n),\tag{20.14}$$

where $v^x = u \Delta t / \Delta x$ and $v^y = v \Delta t / \Delta y$. (We use I, J as the grid indices in this section, so that $i = \sqrt{-1}$ can be used in the complex exponential.) As in one dimension, Fourier analysis decouples the constant-coefficient linear difference equation into separate equations for each mode, so it suffices to consider data consisting of a single arbitrary Fourier mode

$$Q_{IJ}^n = e^{i(\xi I \Delta x + \eta J \Delta y)},\tag{20.15}$$

where ξ and η are the wave numbers in x and y . Inserting this into (20.14) gives

$$Q_{IJ}^{n+1} = g(\xi, \eta, \Delta x, \Delta y, \Delta t) Q_{IJ}^n\tag{20.16}$$

with amplification factor

$$g(\xi, \eta, \Delta x, \Delta y, \Delta t) = (1 - v^x - v^y) + v^x e^{-i\xi \Delta x} + v^y e^{-i\eta \Delta y}.\tag{20.17}$$

The method is stable in the 2-norm provided that $|g| \leq 1$ for all choices of ξ and η . Values of g lie a distance at most $v^x + v^y$ from the point $1 - v^x - v^y$ in the complex plane, and hence the method is stable for $0 \leq v^x + v^y \leq 1$. By considering other choices for the sign of u and v we find that in general the stability limit (20.6) is required for the DCU method.

We now turn to the CTU method, which has the form (20.10) when $u, v > 0$. With our current notation this becomes

$$\begin{aligned} Q_{IJ}^{n+1} = & Q_{IJ}^n - v^x(Q_{IJ}^n - Q_{I-1,J}^n) - v^y(Q_{IJ}^n - Q_{I,J-1}^n) \\ & + \frac{1}{2}v^xv^y[(Q_{IJ}^n - Q_{I,J-1}^n) - (Q_{I-1,J}^n - Q_{I-1,J-1}^n) \\ & + (Q_{IJ}^n - Q_{I-1,J}^n) - (Q_{I,J-1}^n - Q_{I-1,J-1}^n)]. \end{aligned} \quad (20.18)$$

Inserting the Fourier mode (20.15) into this again gives an expression of the form (20.16) with amplification factor

$$\begin{aligned} g(\xi, \eta, \Delta x, \Delta y, \Delta t) = & 1 - v^x(1 - e^{-i\xi \Delta x}) - v^y(1 - e^{-i\eta \Delta y}) \\ & + \frac{1}{2}v^xv^y[(1 - e^{-i\eta \Delta y}) - e^{-i\xi \Delta x}(1 - e^{-i\eta \Delta y}) \\ & + (1 - e^{-i\xi \Delta x}) - e^{-i\eta \Delta y}(1 - e^{-i\xi \Delta x})] \\ = & [1 - v^x(1 - e^{-i\xi \Delta x})][1 - v^y(1 - e^{-i\eta \Delta y})]. \end{aligned} \quad (20.19)$$

Now g is the product of two one-dimensional terms. The method is stable if and only if both terms lie in the unit circle for all choices of ξ and η , and so the method is stable provided $\max(v^x, v^y) \leq 1$. By considering other choices for the sign of u and v we find that in general the stability limit (20.7) is required for the CTU method.

20.5 The CTU Method for Variable-Coefficient Advection

The formulas (20.13) are for the advection equation in the special case $u, v > 0$. For different directions of flow the fluxes must be properly specified to reflect the propagation directions. In this section we will give the general formulas based on a simple wave-propagation procedure. We also now consider the more general context where we allow the velocities to vary in space, since this is equally easy to handle with the wave-propagation approach.

Here we consider the *color-equation* form of the advection equation,

$$q_t + u(x, y)q_x + v(x, y)q_y = 0, \quad (20.20)$$

as discussed in Section 9.3. The conservative form of the two-dimensional advection equation can be solved by extensions of the approach developed in Section 9.5.2.

We assume that the velocities are specified at cell edges (see Section 9.5 for the one-dimensional case) with $u_{i-1/2,j}$ specified at the edge between cells $(i-1, j)$ and (i, j) and $v_{i,j-1/2}$ at the edge between cells $(i, j-1)$ and (i, j) . Only the normal velocity is needed at each edge in order to determine the normal flux through that portion of the cell boundary. Ideally these should be averages of the true normal velocity along the corresponding edge of the cell (see Section 20.8).

The Riemann problem at each interface leads to a single wave with speed given by the edge velocity. In the x -direction we have

$$\begin{aligned}\mathcal{W}_{i-1/2,j} &= Q_{ij} - Q_{i-1,j}, \\ s_{i-1/2,j} &= u_{i-1/2,j},\end{aligned}\tag{20.21}$$

and in the y -direction

$$\begin{aligned}\mathcal{W}_{i,j-1/2} &= Q_{ij} - Q_{i,j-1}, \\ s_{i,j-1/2} &= v_{i,j-1/2}.\end{aligned}\tag{20.22}$$

The fluctuations needed for the DCU algorithm are the natural generalizations of (20.4),

$$\begin{aligned}\mathcal{A}^\pm \Delta Q_{i-1/2,j} &= s_{i-1/2,j}^\pm \mathcal{W}_{i-1/2,j} = u_{i-1/2,j}^\pm (Q_{ij} - Q_{i-1,j}), \\ \mathcal{B}^\pm \Delta Q_{i,j-1/2} &= s_{i,j-1/2}^\pm \mathcal{W}_{i,j-1/2} = v_{i,j-1/2}^\pm (Q_{ij} - Q_{i,j-1}),\end{aligned}\tag{20.23}$$

To compute the correction fluxes needed for the CTU method, we view each wave as potentially propagating transversely into each of the neighboring cells (see Figure 20.4). Rather than giving a single expression for each correction flux, we will build up these fluxes by adding in any transverse terms arising from each Riemann problem as it is solved.

At the beginning of each time step we set

$$\tilde{F}_{i-1/2,j} := 0 \quad \text{and} \quad \tilde{G}_{i,j-1/2} := 0 \quad \forall i, j.$$

After solving each Riemann problem in the x -direction, at interface $(i - 1/2, j)$, we set

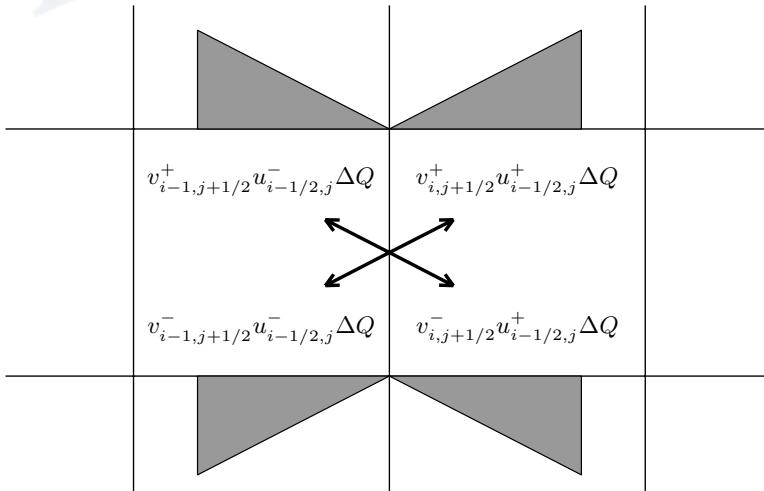


Fig. 20.4. The jump $\Delta Q_{i-1/2,j}$ may propagate in any of four directions, depending on the velocities, and so four neighboring \tilde{G} -fluxes may need to be updated.

$\mathcal{A}^\pm \Delta Q_{i-1/2,j}$ as in (20.23) and then update the nearby correction fluxes by

$$\begin{aligned}\tilde{G}_{i-1,j-1/2} &:= \tilde{G}_{i-1,j-1/2} - \frac{1}{2} \frac{\Delta t}{\Delta x} v_{i-1,j-1/2}^- u_{i-1/2,j}^- (\mathcal{Q}_{ij} - \mathcal{Q}_{i-1,j}), \\ \tilde{G}_{i-1,j+1/2} &:= \tilde{G}_{i-1,j+1/2} - \frac{1}{2} \frac{\Delta t}{\Delta x} v_{i-1,j+1/2}^+ u_{i-1/2,j}^- (\mathcal{Q}_{ij} - \mathcal{Q}_{i-1,j}), \\ \tilde{G}_{i,j-1/2} &:= \tilde{G}_{i,j-1/2} - \frac{1}{2} \frac{\Delta t}{\Delta x} v_{i,j-1/2}^- u_{i-1/2,j}^+ (\mathcal{Q}_{ij} - \mathcal{Q}_{i-1,j}), \\ \tilde{G}_{i,j+1/2} &:= \tilde{G}_{i,j+1/2} - \frac{1}{2} \frac{\Delta t}{\Delta x} v_{i,j+1/2}^+ u_{i-1/2,j}^+ (\mathcal{Q}_{ij} - \mathcal{Q}_{i-1,j}).\end{aligned}\tag{20.24}$$

This takes into account all possible triangular regions. Normally three out of these four updates will be zero, as in the case where u and v are constant. At least two will always be zero, since $u_{i-1/2,j}$ can't be both positive and negative. But if, for example, $u_{i-1/2,j} > 0$ and $v_{i,j+1/2} > 0$ while $v_{i,j-1/2} < 0$, then both $\tilde{G}_{i,j+1/2}$ and $\tilde{G}_{i,j-1/2}$ will be updated, since the wave is evidently flowing transversely into both the cell above and the cell below in this case.

It may also happen that a single interface flux, say $\tilde{G}_{i,j+1/2}$, will be updated by more than one flux correction arising from different Riemann problems, for example if $v_{i,j+1/2} > 0$ and $u_{i-1/2,j} > 0$ while $u_{i+1/2,j} < 0$.

The unsplit algorithms are implemented in CLAWPACK using this same approach. The correction fluxes are all initialized to zero. A Riemann problem is solved at each cell interface, and in addition to determining the fluctuations, the appropriate nearby correction fluxes are updated.

This is implemented by means of a second *transverse Riemann solver* that takes the fluctuation $\mathcal{A}^+ \Delta Q_{i-1/2,j} = u_{i-1/2,j}^+ (\mathcal{Q}_{ij} - \mathcal{Q}_{i-1,j})$, for example, and produces a down-going transverse fluctuation

$$\mathcal{B}^- \mathcal{A}^+ \Delta Q_{i-1/2,j} = v_{i,j-1/2}^- u_{i-1/2,j}^+ (\mathcal{Q}_{ij} - \mathcal{Q}_{i-1,j})\tag{20.25}$$

and an up-going transverse fluctuation

$$\mathcal{B}^+ \mathcal{A}^+ \Delta Q_{i-1/2,j} = v_{i,j+1/2}^+ u_{i-1/2,j}^+ (\mathcal{Q}_{ij} - \mathcal{Q}_{i-1,j}).\tag{20.26}$$

These are used to update the correction fluxes $\tilde{G}_{i,j-1/2}$ and $\tilde{G}_{i,j+1/2}$, respectively, as in (20.24). The left-going fluctuation $\mathcal{A}^- \Delta Q_{i-1/2,j} = u_{i-1/2,j}^- (\mathcal{Q}_{ij} - \mathcal{Q}_{i-1,j})$ results in the transverse fluctuations

$$\mathcal{B}^- \mathcal{A}^- \Delta Q_{i-1/2,j} = v_{i,j-1/2}^- u_{i-1/2,j}^- (\mathcal{Q}_{ij} - \mathcal{Q}_{i-1,j})\tag{20.27}$$

and

$$\mathcal{B}^+ \mathcal{A}^- \Delta Q_{i-1/2,j} = v_{i,j+1/2}^+ u_{i-1/2,j}^- (\mathcal{Q}_{ij} - \mathcal{Q}_{i-1,j}).\tag{20.28}$$

that are used to update $\tilde{G}_{i-1,j-1/2}$ and $\tilde{G}_{i-1,j+1/2}$, respectively. This approach generalizes quite naturally to hyperbolic systems of equations and is presented in Section 21.2. The form of the Riemann solvers required in CLAWPACK is discussed in more detail in Section 21.3.

A similar approach is taken when sweeping in the y -direction. After solving the Riemann problem at interface $(i, j - 1/2)$, we set $\mathcal{B}^\pm \Delta Q_{i,j-1/2}$ as in (20.23) and then update the nearby correction fluxes by

$$\begin{aligned}\tilde{F}_{i-1/2,j-1} &:= \tilde{F}_{i-1/2,j-1} - \frac{1}{2} \frac{\Delta t}{\Delta y} u_{i-1/2,j-1}^- v_{i,j-1/2}^-(Q_{ij} - Q_{i,j-1}), \\ \tilde{F}_{i+1/2,j-1} &:= \tilde{F}_{i+1/2,j-1} - \frac{1}{2} \frac{\Delta t}{\Delta y} u_{i+1/2,j-1}^+ v_{i,j-1/2}^-(Q_{ij} - Q_{i,j-1}), \\ \tilde{F}_{i-1/2,j} &:= \tilde{F}_{i-1/2,j} - \frac{1}{2} \frac{\Delta t}{\Delta y} u_{i-1/2,j}^- v_{i,j-1/2}^+(Q_{ij} - Q_{i,j-1}), \\ \tilde{F}_{i+1/2,j} &:= \tilde{F}_{i+1/2,j} - \frac{1}{2} \frac{\Delta t}{\Delta y} u_{i+1/2,j}^+ v_{i,j-1/2}^+(Q_{ij} - Q_{i,j-1}).\end{aligned}\tag{20.29}$$

20.6 High-Resolution Correction Terms

As noted at the end of Section 20.2, the CTU method fails to be second-order accurate because it is based on first-order accurate approximations to the q_x and q_y terms in the Taylor series expansion, and is missing the q_{xx} and q_{yy} terms altogether. Both of these problems can be fixed by adding in additional correction fluxes based entirely on the one-dimensional theory. In each direction we wish to replace the first-order upwind approximation by a Lax–Wendroff approximation in that direction. (In two dimensions we also need the cross-derivative terms in the Taylor series expansion (20.2), but these have already been included via the transverse terms in the CTU method.) To improve the method we make the following updates to the correction fluxes already defined:

$$\begin{aligned}\tilde{F}_{i-1/2,j} &:= \tilde{F}_{i-1/2,j} + \frac{1}{2} |u_{i-1/2,j}| \left(1 - \frac{\Delta t}{\Delta x} |u_{i-1/2,j}| \right) \tilde{\mathcal{W}}_{i-1/2,j}, \\ \tilde{G}_{i,j-1/2} &:= \tilde{G}_{i,j-1/2} + \frac{1}{2} |v_{i,j-1/2}| \left(1 - \frac{\Delta t}{\Delta y} |v_{i,j-1/2}| \right) \tilde{\mathcal{W}}_{i,j-1/2}.\end{aligned}\tag{20.30}$$

These have exactly the same form as the one-dimensional correction flux (6.56). In the present case there is only a single wave $\mathcal{W}_{i-1/2,j} = Q_{ij} - Q_{i-1,j}$, and as usual $\tilde{\mathcal{W}}_{i-1/2,j}$ represents a limited version of this wave, obtained by comparing this wave with the wave in the upwind direction. If $u_{i-1/2,j} > 0$ and $v_{i,j-1/2} < 0$, for example, then $\mathcal{W}_{i-1/2,j}$ is compared to $\mathcal{W}_{i-3/2,j}$ while $\mathcal{W}_{i,j-1/2}$ is compared to $\mathcal{W}_{i,j+1/2}$.

20.7 Relation to the Lax–Wendroff Method

Suppose we apply the method just derived to the constant-coefficient advection equation (20.1) with no limiters. We might suspect this should reduce to the Lax–Wendroff method for the advection equation, since this is what happens in this situation in one dimension. Indeed, the pure x - and y -derivatives will be approximated as in the Lax–Wendroff method, but the cross-derivative terms are not. Instead, combining the previous expressions yields a

flux-differencing method of the form (19.10) with fluxes

$$\begin{aligned} F_{i-1/2,j} &= \frac{1}{2}u(Q_{i-1,j} + Q_{ij}) - \frac{\Delta t}{2\Delta x}u^2(Q_{ij} - Q_{i-1,j}) \\ &\quad - \frac{\Delta t}{2\Delta y}[u^-v^-(Q_{i,j+1} - Q_{ij}) + u^+v^-(Q_{i-1,j+1} - Q_{i-1,j}) \\ &\quad + u^-v^+(Q_{ij} - Q_{i,j-1}) + u^+v^+(Q_{i-1,j} - Q_{i-1,j-1})], \end{aligned} \quad (20.31)$$

$$\begin{aligned} G_{i,j-1/2} &= \frac{1}{2}v(Q_{i,j-1} + Q_{ij}) - \frac{\Delta t}{2\Delta y}v^2(Q_{ij} - Q_{i,j-1}) \\ &\quad - \frac{\Delta t}{2\Delta x}[v^-u^-(Q_{i+1,j} - Q_{ij}) + v^+u^-(Q_{i+1,j-1} - Q_{i,j-1}) \\ &\quad + v^-u^+(Q_{ij} - Q_{i-1,j}) + v^+u^+(Q_{i,j-1} - Q_{i-1,j-1})]. \end{aligned}$$

Compare this with the Lax–Wendroff method (19.14) for the case $A = u$, $B = v$. Instead of approximating the cross-derivative term with a simple average of four nearby fluxes, as is done in (19.15) for Lax–Wendroff, the wave-propagation algorithm uses

$$\begin{aligned} uvq_y(x_{i-1/2}, y_j) &\approx \frac{1}{\Delta y}[u^-v^-(Q_{i,j+1} - Q_{ij}) + u^+v^-(Q_{i-1,j+1} - Q_{i-1,j}) \\ &\quad + u^-v^+(Q_{ij} - Q_{i,j-1}) + u^+v^+(Q_{i-1,j} - Q_{i-1,j-1})]. \end{aligned} \quad (20.32)$$

In this constant-coefficient case only one of these four terms will be nonzero. Rather than averaging four nearby approximations to q_y , only one is used, taken from the upwind direction.

This leads to an improvement in the stability of the method. The Lax–Wendroff method is generally stable only if

$$\frac{\Delta t}{\Delta x}\sqrt{u^2 + v^2} \leq 1, \quad (20.33)$$

whereas the wave-propagation version is stable up to Courant number 1 in the sense of (20.7). If $u = v$, then this is better by a factor of $\sqrt{2}$. In Chapter 21 we will see that similar improvements can be made in the Lax–Wendroff method for systems of equations, by generalizing (20.32) to systems using the matrices A^\pm and B^\pm .

20.8 Divergence-Free Velocity Fields

Note that the conservative advection equation

$$q_t + (u(x, y)q)_x + (v(x, y)q)_y = 0 \quad (20.34)$$

and the color equation

$$q_t + u(x, y)q_x + v(x, y)q_y = 0 \quad (20.35)$$

are mathematically equivalent if the velocity field is divergence-free,

$$u_x(x, y) + v_y(x, y) = 0, \quad (20.36)$$

a case that arises in many applications. In this case (20.35) is often called the *advective form* of the equation, while (20.34) is the *conservative form*.

The constraint (20.36) holds, for example, for two-dimensional models of incompressible flow and more generally for any flow in which the net flux through the boundary of any arbitrary region Ω should be zero. For then we have

$$0 = \int_{\partial\Omega} \vec{n}(s) \cdot \vec{u}(s) ds = \iint_{\Omega} \vec{\nabla} \cdot \vec{u}(x, y) dx dy, \quad (20.37)$$

where, as in Section 18.1, $\vec{n} \cdot \vec{u}$ is the normal velocity.

If $q(x, y, t)$ measures the density of a conserved tracer in a divergence-free flow, then we expect the integral of q to be conserved and generally hope to achieve this numerically even when discretizing the color equation. Note that numerical conservation would be guaranteed if the conservative equations (20.34) were used instead, but there are other potential disadvantages in using this form.

The methods developed in Sections 20.5 and 20.6 will be conservative on the color equation provided that the edge velocities satisfy

$$\frac{1}{\Delta x} (u_{i+1/2,j} - u_{i-1/2,j}) + \frac{1}{\Delta y} (v_{i,j+1/2} - v_{i,j-1/2}) = 0, \quad (20.38)$$

as will be verified below. This is a natural discrete version of (20.36) across a grid cell. More fundamental is the integral interpretation of this condition. Since $\Delta y u_{i\pm 1/2,j}$ and $\Delta x v_{i,j\pm 1/2}$ are supposed to approximate integrals of the normal velocity along the four sides of the grid cell, we see that multiplying (20.38) by $\Delta x \Delta y$ gives a discrete form of the requirement (20.37). In particular, the discrete divergence-free condition (20.38) will be satisfied if we determine the edge velocities by computing exact averages of the normal velocities,

$$u_{i-1/2,j} = \frac{1}{\Delta y} \int_{y_{j-1/2}}^{y_{j+1/2}} u(x_{i-1/2}, y) dy, \\ v_{i,j-1/2} = -\frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} v(x, y_{j-1/2}) dx. \quad (20.39)$$

In this case (20.38) follows immediately as a special case of (20.37) for $\Omega = \mathcal{C}_{ij}$, the (i, j) grid cell.

Unfortunately the integrals in (20.39) may be hard to evaluate exactly. If the velocities are smooth, then simply evaluating the normal velocity at the midpoint of each edge will give values that are second-order accurate, but probably will not exactly satisfy the condition (20.38). In Section 20.8.1 we will see an approach to specifying these values using a stream function that is often quite simple to implement.

First we verify that the condition (20.38) does lead to discrete conservation when the color equation is solved using the method developed above. Recall that the wave-propagation method for the color equation can be written in the form (19.19) with the fluctuations (20.23) and correction fluxes \tilde{F} and \tilde{G} arising from both transverse propagation and high-resolution corrections, if these are included. Since these corrections are implemented by differencing the fluxes \tilde{F} and \tilde{G} , they will maintain conservation. Thus we only need to worry about the

fluctuations in verifying that the full method is conservative, and it is enough to consider the DCU method.

$$\begin{aligned} Q_{ij}^{n+1} = & Q_{ij} - \frac{\Delta t}{\Delta x} [u_{i-1/2,j}^+ (Q_{ij} - Q_{i-1,j}) + u_{i+1/2,j}^- (Q_{i+1,j} - Q_{ij})] \\ & - \frac{\Delta t}{\Delta y} [v_{i,j-1/2}^+ (Q_{ij} - Q_{i,j-1}) + v_{i,j+1/2}^- (Q_{i,j+1} - Q_{ij})]. \end{aligned} \quad (20.40)$$

Summing this equation over all i and j and rearranging the sum on the right to collect together all terms involving $Q_{ij} = Q_{ij}^n$, we obtain

$$\sum_{i,j} Q_{ij}^{n+1} = \sum_{i,j} Q_{ij}^n \left[1 + \frac{\Delta t}{\Delta x} (u_{i+1/2,j} - u_{i-1/2,j}) + \frac{\Delta t}{\Delta y} (v_{i,j+1/2} - v_{i,j-1/2}) \right]. \quad (20.41)$$

We see that the method is conservative provided that (20.38) is satisfied, in the sense that $\sum Q_{ij}^{n+1} = \sum Q_{ij}^n$ up to boundary fluxes.

20.8.1 Stream-Function Specification of Velocities

It is often easiest to define a two-dimensional divergence-free velocity field in terms of a *streamfunction* $\psi(x, y)$. Any continuous and piecewise differential scalar function $\psi(x, y)$ can be used to define a velocity field via

$$\begin{aligned} u(x, y) &= \psi_y(x, y), \\ v(x, y) &= -\psi_x(x, y). \end{aligned} \quad (20.42)$$

This velocity field will be divergence-free, since

$$u_x + v_y = \psi_{yx} - \psi_{xy} = 0.$$

Note that the velocity field $\vec{u} = (u, v)$ is orthogonal to $\vec{\nabla}\psi = (\psi_x, \psi_y)$, and hence contour lines of ψ in the x - y plane are *streamlines* of the flow, and are simply particle paths of the flow in the case we are considering, where ψ and hence \vec{u} is independent of t .

If we know the stream function ψ for a velocity field, then it is easy to compute the edge velocities (20.39) by integrating (20.42), yielding the simple formulas

$$\begin{aligned} u_{i-1/2,j} &= \frac{1}{\Delta y} [\psi(x_{i-1/2}, y_{j+1/2}) - \psi(x_{i-1/2}, y_{j-1/2})], \\ v_{i,j-1/2} &= -\frac{1}{\Delta x} [\psi(x_{i+1/2}, y_{j-1/2}) - \psi(x_{i-1/2}, y_{j-1/2})]. \end{aligned} \quad (20.43)$$

We simply difference ψ between two corners to determine the total flow normal to that edge. More generally, differencing ψ between any two points in the plane gives the total flow normal to the line between those points, a fact that is useful in defining edge velocities on more general curvilinear grids (see Section 23.5.2).

The expressions (20.43) might also be interpreted as centered approximations to the derivatives in (20.42), but since they are exactly equal to the integrals (20.39), the discrete divergence-free condition (20.38) will be satisfied. This fact can also be verified

directly from the formulas (20.43), since differencing the edge velocities as in (20.38) using the expressions (20.43) leads to a complete cancellation of the four corner values of ψ .

20.8.2 Solid-Body Rotation

As an example with circular streamlines, consider the stream function

$$\psi(x, y) = x^2 + y^2. \quad (20.44)$$

The resulting velocity field

$$u(x, y) = 2y, \quad v(x, y) = -2x \quad (20.45)$$

corresponds to solid-body rotation. This is a nice test problem for two-dimensional advection algorithms, since the true solution is easily found for any initial data. In particular, the solution at time $t = N\pi$ agrees with the initial data for any integer N , since the flow has then made N complete rotations.

Example 20.1. Figure 20.5 shows the results of solid-body rotation on a 80×80 grid with data $q = 0$ except in a square region where $q = 1$ and a circular region where q is cone-shaped, growing to a value 1 at the center:

$$q(x, y, 0) = \begin{cases} 1 & \text{if } 0.1 < x < 0.6 \text{ and } -0.25 < y < 0.25, \\ 1 - r/0.35 & \text{if } r \equiv \sqrt{(x + 0.45)^2 + y^2} < 0.35, \\ 0 & \text{otherwise.} \end{cases} \quad (20.46)$$

The results are not perfect, of course. The discontinuity in q is smeared out, and the peak of the cone is chopped off. However, this unsplit high-resolution method (using the MC limiter) gives much better results than would be obtained with more classical methods. For example, Figure 20.6 shows what we would obtain with the first-order CTU method or the second-order method with no limiter. This figure also shows results obtained using dimensional splitting with one-dimensional high-resolution methods, which compare very well with the unsplit results of Figure 20.5.

20.9 Nonlinear Scalar Conservation Laws

The methods developed in Sections 20.5 and 20.6 extend easily to nonlinear scalar conservation laws $q_t + f(q)_x + g(q)_y = 0$. The one-dimensional Riemann problem normal to each cell edge is solved as in one dimension, resulting in waves, speeds, and fluctuations.

In the x -direction we have

$$\begin{aligned} W_{i-1/2,j} &= Q_{ij} - Q_{i-1,j}, \\ s_{i-1/2,j} &= \begin{cases} [f(Q_{ij}) - f(Q_{i-1,j})]/(Q_{ij} - Q_{i-1,j}) & \text{if } Q_{i-1,j} \neq Q_{ij}, \\ f'(Q_{ij}) & \text{if } Q_{i-1,j} = Q_{ij}, \end{cases} \end{aligned} \quad (20.47)$$

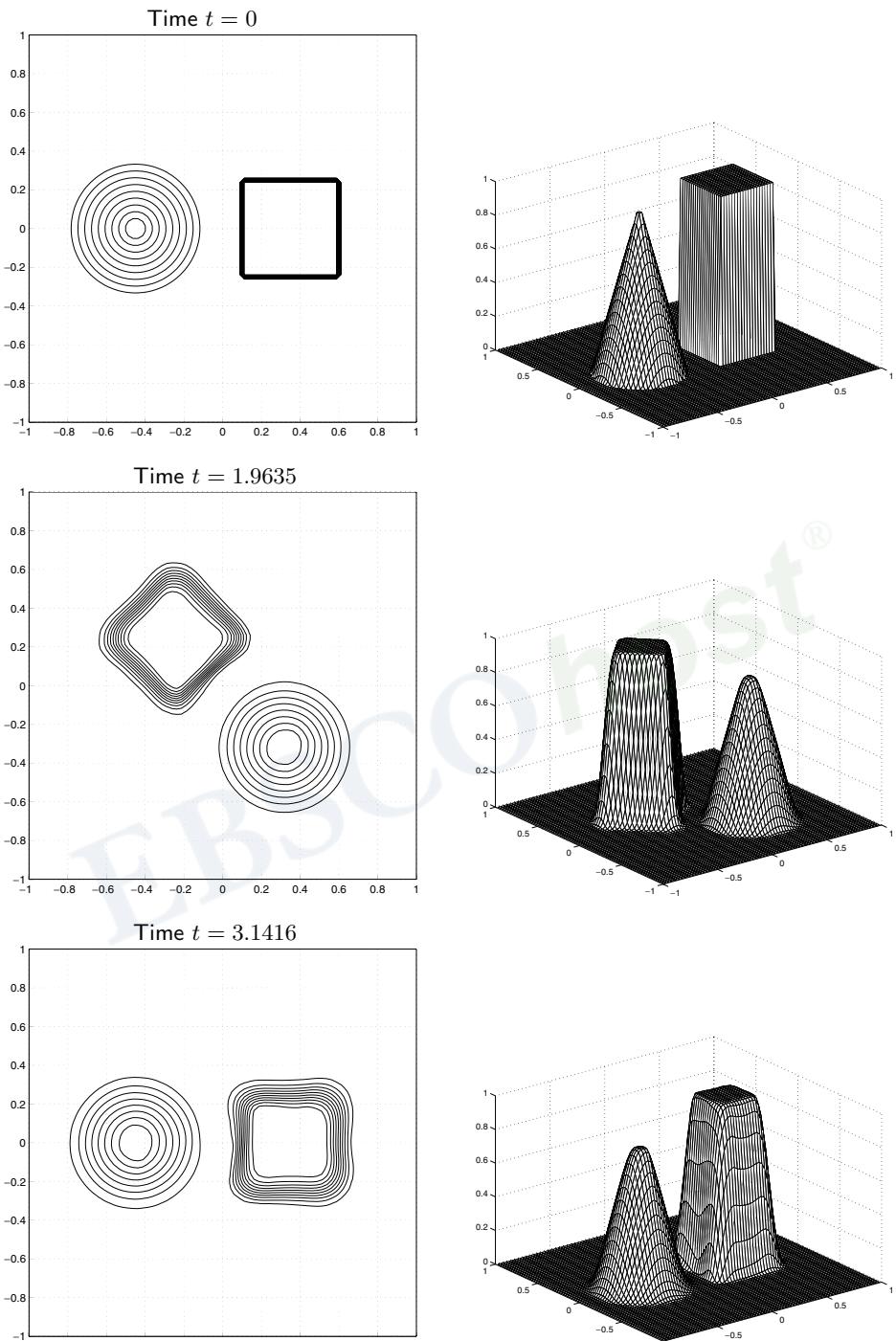


Fig. 20.5. Solid-body rotation from Example 20.1. The solution computed on an 80×80 grid is shown at three different times. Top: $t = 0$; middle: $t = 5\pi/8$; bottom: $t = \pi$. At each time the solution is shown as a contour plot (left) and a mesh plot (right). Contour lines are at the values $q = 0.05, 0.15, 0.25, \dots, 0.95$. [claw/book/chap20/rotate]

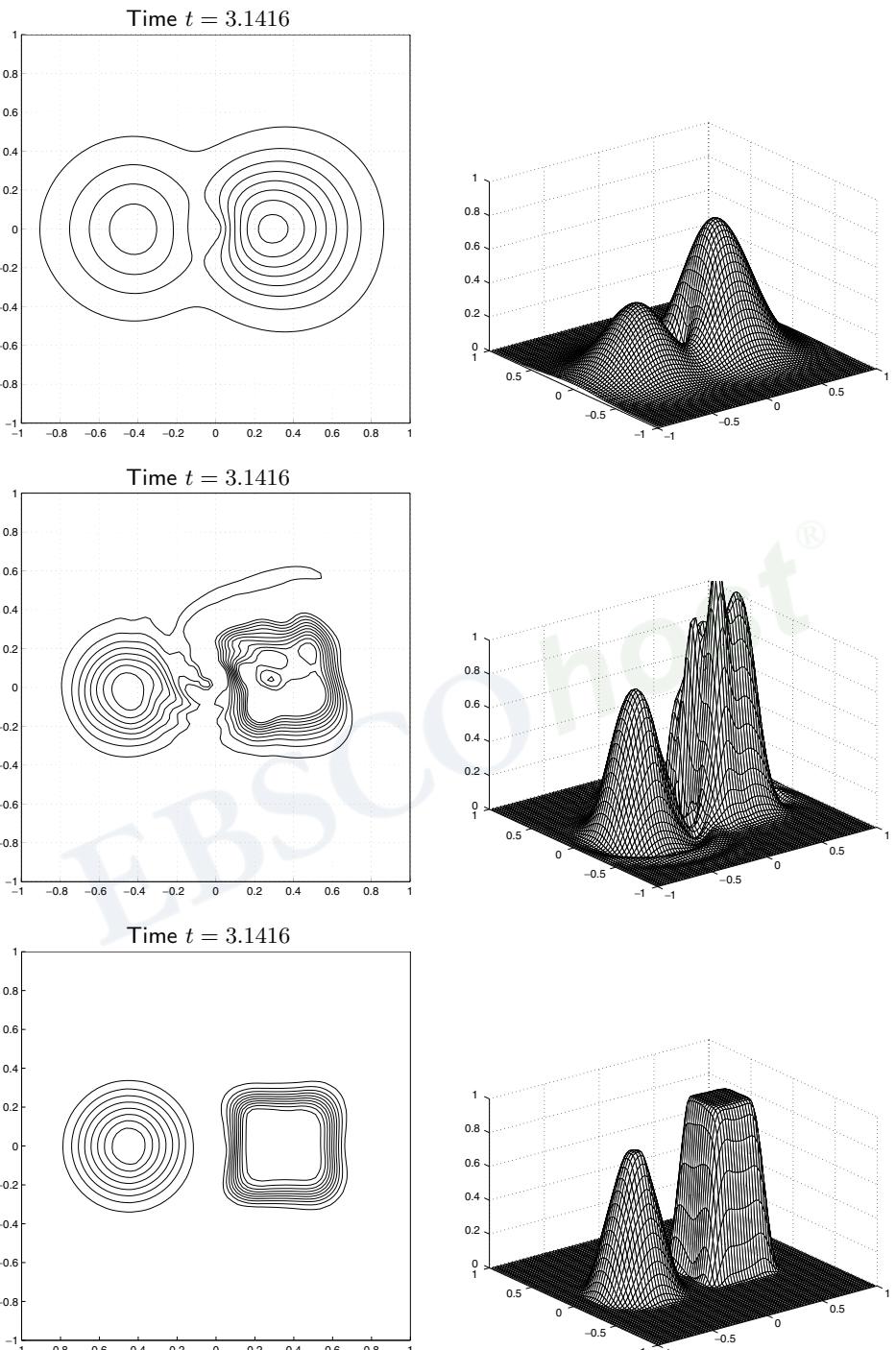


Fig. 20.6. Solid-body rotation from Example 20.1. Results obtained with the first-order CTU method (top), the second-order “Lax–Wendroff” method with no limiter (middle), and dimensional splitting with the high-resolution method in each direction (bottom). Each is shown only at the final time $t = \pi$.
[claw/book/chap20/rotate]

and in the y -direction

$$\mathcal{W}_{i,j-1/2} = Q_{ij} - Q_{i,j-1},$$

$$s_{i,j-1/2} = \begin{cases} [g(Q_{ij}) - g(Q_{i,j-1})]/(Q_{ij} - Q_{i,j-1}) & \text{if } Q_{i,j-1} \neq Q_{ij}, \\ g'(Q_{ij}) & \text{if } Q_{i,j-1} = Q_{ij}. \end{cases} \quad (20.48)$$

The fluctuations can be defined simply as

$$\mathcal{A}^\pm \Delta Q_{i-1/2,j} = s_{i-1/2,j}^\pm \mathcal{W}_{i-1/2,j},$$

$$\mathcal{B}^\pm \Delta Q_{i,j-1/2} = s_{i,j-1/2}^\pm \mathcal{W}_{i,j-1/2}, \quad (20.49)$$

except in the case of transonic rarefactions, where these must be modified as in Section 12.3. The wave and speed can be used to compute second-order correction terms as in (20.30),

$$\tilde{F}_{i-1/2,j} := \tilde{F}_{i-1/2,j} + \frac{1}{2} |s_{i-1/2,j}| \left(1 - \frac{\Delta t}{\Delta x} |s_{i-1/2,j}| \right) \tilde{\mathcal{W}}_{i-1/2,j},$$

$$\tilde{G}_{i,j-1/2} := \tilde{G}_{i,j-1/2} + \frac{1}{2} |s_{i,j-1/2}| \left(1 - \frac{\Delta t}{\Delta y} |s_{i,j-1/2}| \right) \tilde{\mathcal{W}}_{i,j-1/2}. \quad (20.50)$$

The only subtle point is the determination of transverse velocities for the CTU terms corresponding to the correction fluxes developed in Section 20.5 for the advection equation. We no longer have velocities specified at nearby cell edges. Instead, the transverse velocity must be determined by approximating $g'(q)$ based on the data $Q_{i-1,j}$ and Q_{ij} (or other data nearby). One natural approach that generalizes quite easily to systems of equations, as we will see in the next chapter, is to choose the transverse velocity to be

$$\hat{v}_{i-1/2,j} = \begin{cases} [g(Q_{ij}) - g(Q_{i-1,j})]/(Q_{ij} - Q_{i-1,j}) & \text{if } Q_{i-1,j} \neq Q_{ij}, \\ g'(Q_{ij}) & \text{if } Q_{i-1,j} = Q_{ij}. \end{cases} \quad (20.51)$$

We then set

$$\tilde{G}_{i-1,j-1/2} := \tilde{G}_{i-1,j-1/2} - \frac{1}{2} \frac{\Delta t}{\Delta x} \hat{v}_{i-1/2,j}^- s_{i-1/2,j}^-(Q_{ij} - Q_{i-1,j}),$$

$$\tilde{G}_{i-1,j+1/2} := \tilde{G}_{i-1,j+1/2} - \frac{1}{2} \frac{\Delta t}{\Delta x} \hat{v}_{i-1/2,j}^+ s_{i-1/2,j}^+(Q_{ij} - Q_{i-1,j}),$$

$$\tilde{G}_{i,j-1/2} := \tilde{G}_{i,j-1/2} - \frac{1}{2} \frac{\Delta t}{\Delta x} \hat{v}_{i-1/2,j}^- s_{i-1/2,j}^+(Q_{ij} - Q_{i-1,j}),$$

$$\tilde{G}_{i,j+1/2} := \tilde{G}_{i,j+1/2} - \frac{1}{2} \frac{\Delta t}{\Delta x} \hat{v}_{i-1/2,j}^+ s_{i-1/2,j}^+(Q_{ij} - Q_{i-1,j}). \quad (20.52)$$

Note that this is somewhat different from (20.24) in that a single transverse velocity $\hat{v}_{i-1/2,j}$ is used based on the data $Q_{i-1,j}$ and Q_{ij} rather than the four edge values $v_{i,j\pm 1/2}$ and $v_{i-1,j\pm 1/2}$ appearing in (20.24). Similarly, a transverse velocity $\hat{u}_{i,j-1/2}$ is defined in the course of solving Riemann problems in the y -direction and is used to update nearby \tilde{F} -fluxes.

20.9.1 Burgers Equation

The inviscid Burgers equation (11.13) can be generalized to two space dimensions as

$$u_t + n^x \left(\frac{1}{2} u^2 \right)_x + n^y \left(\frac{1}{2} u^2 \right)_y = 0 \quad (20.53)$$

where $\vec{n} = (n^x, n^y)$ is an arbitrary unit vector. For $\vec{n} = (1, 0)$ or $(0, 1)$, this is just the one-dimensional Burgers equation in x or y respectively. More generally this can be reduced to a one-dimensional Burgers equation at angle $\theta = \tan^{-1}(n^y/n^x)$ to the x -axis. If we introduce new coordinates ξ in this direction and η in the orthogonal direction, then (20.53) reduces to

$$u_t + \left(\frac{1}{2} u^2 \right)_\xi = 0. \quad (20.54)$$

Along each slice in the ξ -direction, we can solve this one-dimensional equation to obtain the solution $u(x, y, t)$ along this slice.

Figure 20.7 shows some sample results, using the same initial data (20.46) as for the solid-body rotation example shown in Figure 20.5. Two different angles, $\theta = 0$ and $\theta = \pi/4$, are illustrated. These were computed on a 300×300 grid using the high-resolution wave-propagation algorithm with the MC limiter.

20.10 Convergence

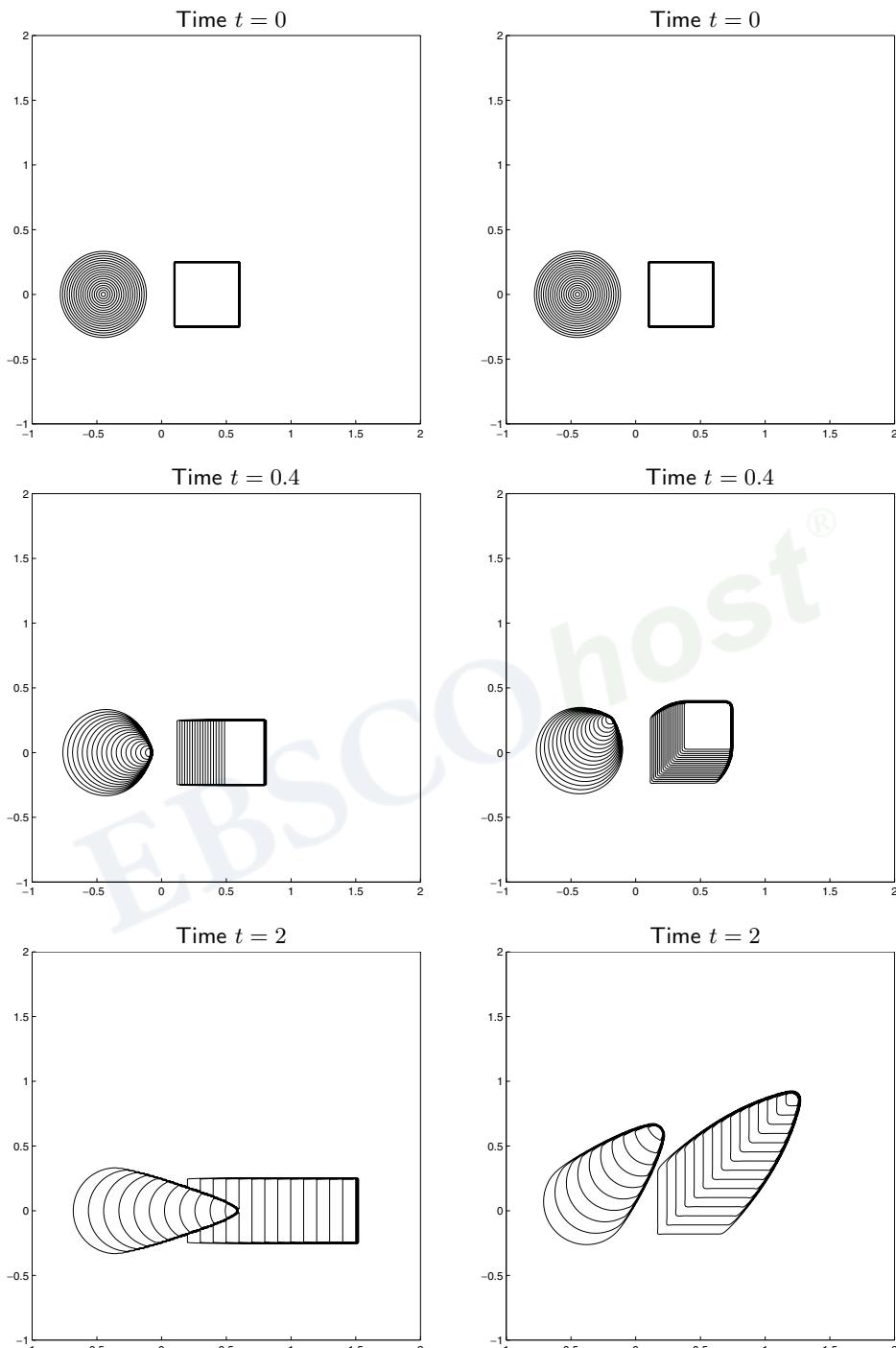
Convergence theory for multidimensional numerical methods is even more difficult than for one-dimensional problems. For scalar problems several results are known, however, and some of these are summarized in this section. See [156], [245] for more detailed discussions.

20.10.1 Convergence of Dimensional Splitting

In considering the convergence of dimensionally split methods, the first natural question is whether convergence occurs when the exact solution operator is used in each one-dimensional sweep (19.25) and (19.26). For nonlinear conservation laws this was shown by Crandall & Majda [95], even when the solution contains shock waves. Let $\mathcal{S}(t)$ represent the true solution operator of the full equation $q_t + f(q)_x + g(q)_y = 0$ over time t , so $\mathcal{S}(t)\vec{q}$ is the (unique) entropy-satisfying solution at time t , $(\mathcal{S}(t)\vec{q})(x, t) = q(x, y, t)$. Similarly, let $\mathcal{S}^x(t)$ and $\mathcal{S}^y(t)$ be the solution operators for the one-dimensional problems $q_t + f(q)_x = 0$ and $q_t + g(q)_y = 0$, respectively. Then convergence in the 1-norm of both the Godunov and Strang splitting is guaranteed by the following theorem.

Theorem 20.1 (Crandall & Majda [95]). *If the exact solution operator is used in each step of the fractional-step procedure, then the method converges to the weak solution of the two-dimensional scalar conservation law, i.e.,*

$$\|\mathcal{S}(T)\vec{q} - [\mathcal{S}^y(\Delta t)\mathcal{S}^x(\Delta t)]^n \vec{q}\|_1 \rightarrow 0$$



Copyright © 2002, Cambridge University Press.
All rights reserved. May not be reproduced in any form without permission from the publisher, except fair uses permitted under U.S. or applicable copyright law.

Fig. 20.7. Solution of the two-dimensional Burgers equation (20.53) at angle θ to the x -axis.
Left column: $\theta = 0$. Right column: $\theta = \pi/4$. Contour lines are at $u = 0.05 : 0.05 : 0.95$.
[claw/book/chap20/burgers]

and

$$\|\mathcal{S}(T)\tilde{q} - [\mathcal{S}^x(\Delta t/2)\mathcal{S}^y(\Delta t)\mathcal{S}^x(\Delta t/2)]^n \tilde{q}\|_1 \rightarrow 0$$

as $\Delta t \rightarrow 0$ and $n \rightarrow \infty$ with $n \Delta t = T$.

This shows that there is hope that numerical methods based on these splittings will also converge to the solution of the two-dimensional problem. If we use monotone methods (see Section 12.12) for each one-dimensional problem, then this can in fact be shown:

Theorem 20.2 (Crandall & Majda [95]). *If the exact solution operators $\mathcal{S}^x(\Delta t)$ and $\mathcal{S}^y(\Delta t)$ in the above theorem are replaced by monotone methods for the one-dimensional conservation laws, then the results still hold.*

20.10.2 Total Variation in Two Dimensions

For high-resolution methods on scalar problems, a basic tool in one dimension is the total variation. In two dimensions this is of more limited use, as we now explore.

The true solution to a scalar conservation law is still total variation diminishing (TVD) in two dimensions, where the total variation is now defined as

$$\begin{aligned} \text{TV}(q) &= \limsup_{\epsilon \rightarrow 0} \frac{1}{\epsilon} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |q(x + \epsilon, y) - q(x, y)| dx dy \\ &\quad + \limsup_{\epsilon \rightarrow 0} \frac{1}{\epsilon} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |q(x, y + \epsilon) - u(x, y)| dx dy. \end{aligned} \quad (20.55)$$

We can define the total variation of a discrete grid function analogously by

$$\text{TV}(Q) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} [\Delta y |Q_{i+1,j} - Q_{ij}| + \Delta x |Q_{i,j+1} - Q_{ij}|]. \quad (20.56)$$

In one space dimension we found that requiring a method to be TVD for scalar problems was a very useful requirement in developing high-resolution methods, because of the following two facts:

1. It is possible to derive methods that are TVD in general and also second-order accurate on smooth solutions (at least away from extrema).
2. A TVD method guarantees approximations that all lie in some compact set (as the grid is refined), and hence convergence can be proved for nonlinear problems.

This meant that we could derive high-resolution methods that resolve discontinuities without spurious oscillations while also giving good accuracy on smooth solutions and being provably convergent.

In two dimensions we might hope to do the same. Since the true solution is still TVD and we again wish to avoid spurious oscillations, we might try to require that the numerical method be TVD. With the variation defined as in (20.56), this would guarantee that all approximate solutions lie in an appropriate compact set and allow us to prove convergence,

just as in one dimension. It is then natural to look for conditions similar to Harten's conditions of Theorem 6.1 that might guarantee the solution is TVD and also be loose enough to allow second-order accuracy. Unfortunately, an attempt to do this resulted instead in the following negative result.

Theorem 20.3 (Goodman & LeVeque [159]). *Except in certain trivial cases, any method that is TVD in two space dimensions is at most first-order accurate.*

This does not mean, however, that it is impossible to achieve high-resolution results in two dimensions. In fact the method described in Section 20.9 works very well in practice and gives results that are typically as sharp and accurate as one would expect based on one-dimensional experience.

Also, dimensional splitting often works very well when one-dimensional high-resolution methods are applied in each direction separately. Note that if the second-order Strang splitting is used, then this method is "second-order accurate" to the extent that the one-dimensional high-resolution method is. Moreover, in each sweep limiters are applied that keep the one-dimensional variation along that row of cells from increasing, and thus they do a good job of insuring that no spurious oscillations arise. The problem is that this is not enough to prove that the two-dimensional variation defined by (20.56) does not increase. A rather pathological example constructed in [159] shows that the two-dimensional variation may in fact increase. In practice this is not generally an issue, however, and Theorem 20.3 simply means that the TVD notion is not as useful for proving convergence of high-resolution methods in two dimensions as it is in one dimension.

A number of other techniques have instead been introduced for proving convergence of numerical methods for nonlinear scalar conservation laws in more than one dimension. One approach that has been quite successful is to use the theory of measure-valued solutions of conservation laws introduced by DiPerna [110]. This requires a weaker condition than uniformly bounded variation of the approximate solutions. Szepessy [430] used this to prove convergence of a finite element method, and Coquel & Le Floch [89], [90] applied a similar approach to finite volume methods, as did Kröner & Rokyta [247]. See [74], [75], [76], [246], [251], [473], [485] for some other work on convergence and error estimates for multidimensional scalar problems.

Exercises

20.1. Consider the advection equation $q_t + q_x + q_y = 0$ with initial data

$$Q_{ij}^0 = \begin{cases} 1 & \text{if } i + j \leq 0, \\ 0 & \text{if } i + j > 0 \end{cases}$$

for the Cauchy problem ($-\infty < i, j < \infty$). Suppose $\Delta t = \Delta x = \Delta y$, so that the Courant number is 1. Determine the solution Q_{ij}^1 and Q_{ij}^2 after 1 and 2 time steps when each of the following algorithms is applied:

- (a) The DCU algorithm of Section 20.1. Observe that there is an exponentially growing oscillation and the method is unstable at this Courant number.

- Copyright © 2002, Cambridge University Press.
All rights reserved. May not be reproduced in any form without permission from the publisher, except fair uses permitted under U.S. or applicable copyright law.
- (b) The CTU algorithm of Section 20.2. Observe that the method is stable and produces the exact solution at this Courant number.
 - 20.2. Consider the Cauchy problem for the constant-coefficient advection equation with $u, v > 0$. Suppose we apply dimensional splitting using the Godunov splitting defined by (19.27) and (19.28) with the one-dimensional first-order upwind algorithm in each sweep. Eliminate Q^* to determine how Q^{n+1} is defined in terms of Q^n , and show that this is equivalent to the CTU algorithm of Section 20.2. Would the same be true for the variable-coefficient advection equation?
 - 20.3. Verify that (20.41) follows from (20.40).
 - 20.4. Show that computing the discrete divergence (20.38) using the edge velocities (20.43) leads to a complete cancellation of the four corner values of ψ , verifying that use of the stream function gives a divergence-free discrete velocity field.
 - 20.5. Use CLAWPACK to solve the solid-body rotation problem of Section 20.8.2 on the domain $[-1, 1] \times [0, 1]$ with $\hat{q}(x, y) \equiv 0$ and boundary conditions

$$q(x, 0, t) = \begin{cases} 0 & \text{if } x < -0.8 \text{ or } -0.2 < x < 0, \\ 1 & \text{if } -0.8 \leq x \leq -0.2, \end{cases}$$

and extrapolation boundary conditions along the remainder of the boundary. Observe that the solution should reach a steady state after $t = \pi$ with the profile specified along the inflow boundary being reproduced at the outflow boundary. Compare how well the various methods implemented in CLAWPACK perform on this problem. You might also try other choices of the inflow boundary conditions, e.g., a smooth function of x .

- 20.6. In one space dimension the true solution to the variable-coefficient color equation $q_t + u(x)q_x = 0$ is TVD. Is the same true in two dimensions for the equation $q_t + u(x, y)q_x + v(x, y)q_y = 0$ with respect to the total variation (20.56)? Hint: Consider the special case $q_t + u(y)q_x = 0$.

Multidimensional Systems

In this chapter the high-resolution wave-propagation algorithms developed in Chapter 20 for scalar problems are extended to hyperbolic systems. We start with constant-coefficient linear systems, where the essential ingredients are most easily seen. A Riemann problem is first solved normal to each cell edge (a simple eigendecomposition in the linear case). The resulting waves are used to update cell averages on either side. The addition of correction terms using wave limiters (just as in one dimension) gives high-resolution terms modeling the pure x - and y -derivative terms in the Taylor series expansion (19.5). The cross-derivative terms are handled by simple extension of the corner-transport upwind (CTU) idea presented for the advection equation in Sections 20.2 through 20.5. In general this requires solving a second set of Riemann problems transverse to the interface. For a linear system this means performing a second eigendecomposition using the coefficient matrix in the transverse direction. Extending the methods to variable-coefficient or nonlinear systems is then easy, using ideas that are already familiar from one space dimension. The solutions (or approximate solutions) to the more general Riemann problems are used in place of the eigendecompositions, and the method is implemented in a wave-propagation form that applies very generally.

21.1 Constant-Coefficient Linear Systems

We again consider the constant-coefficient linear system $q_t + Aq_x + Bq_y = 0$ discussed in Chapter 19, where in particular the Lax–Wendroff and Godunov methods for this system were presented. The numerical fluxes for these two methods are given by (19.14) and (19.18) respectively. Our goal is to create a high-resolution version of the Lax–Wendroff method that incorporates upwinding and limiting. In addition to upwinding the first-order term (to rewrite it as the Godunov flux plus a high-resolution correction), we also wish to upwind the cross-derivative terms, as motivated by the CTU method for advection presented in Section 20.2. This can be accomplished by replacing the approximation (19.15) to ABq_y used in the Lax–Wendroff method with

$$\begin{aligned} ABq_y(x_{i-1/2}, y_j) \approx & \frac{1}{\Delta y} [A^- B^-(Q_{i,j+1} - Q_{ij}) + A^+ B^-(Q_{i-1,j+1} - Q_{i-1,j}) \\ & + A^- B^+(Q_{ij} - Q_{i,j-1}) + A^+ B^+(Q_{i-1,j} - Q_{i-1,j-1})], \end{aligned} \quad (21.1)$$

generalizing the expression (20.28) for the CTU method on the advection equation. A similar approximation is used for the BAq_x term in the G -fluxes. Here A^\pm and B^\pm are defined as usual by (19.17). Note the important fact that

$$A^-B^- + A^+B^- + A^-B^+ + A^+B^+ = (A^- + B^+)(A^- + B^+) = AB \quad (21.2)$$

for arbitrary matrices A and B , which ensures that we are still using a consistent approximation to the cross-derivative terms. Rather than multiplying each of the four jumps in Q by $\frac{1}{4}AB$ as in the Lax–Wendroff method, the product AB is split into four unequal pieces based on upwinding.

In practice we do not compute these matrices or the matrix products indicated above. Instead these terms in the flux are computed by solving Riemann problems and accumulating contributions to the fluxes based on the direction of wave propagation, exactly as was done for the advection equation in Section 20.2. This approach makes it easy to extend the method to variable-coefficient or nonlinear problems.

In spite of the fact that we do not actually compute the fluxes in terms of these matrices, it is useful to display them in this form for comparison with the Lax–Wendroff fluxes (19.14). We have

$$\begin{aligned} F_{i-1/2,j} = & A^+Q_{i-1,j} + A^-Q_{ij} + \frac{1}{2} \sum_{p=1}^m |\lambda^{xp}| \left(1 - \frac{\Delta t}{\Delta x} |\lambda^{xp}| \right) \tilde{\mathcal{W}}_{i-1/2,j}^p \\ & - \frac{\Delta t}{2\Delta y} [A^-B^-(Q_{i,j+1} - Q_{ij}) + A^+B^-(Q_{i-1,j+1} - Q_{i-1,j}) \\ & + A^-B^+(Q_{ij} - Q_{i,j-1}) + A^+B^+(Q_{i-1,j} - Q_{i-1,j-1})], \end{aligned} \quad (21.3)$$

and a similar expression for $G_{i,j-1/2}$. Here $\mathcal{W}_{i-1/2,j}^p = \alpha_{i-1/2,j}^p r^{xp}$ is the p th wave in the Riemann solution, with $\alpha_{i-1/2,j}^p = (R^x)^{-1}(Q_{ij} - Q_{i-1,j})$. The limited version $\tilde{\mathcal{W}}_{i-1/2,j}^p$ is obtained by comparing this wave with $\tilde{\mathcal{W}}_{I-1/2,j}^p$, where

$$I = \begin{cases} i-1 & \text{if } \lambda^{xp} > 0, \\ i+1 & \text{if } \lambda^{xp} < 0. \end{cases}$$

If no limiter is used, then, as in one dimension,

$$\sum_{p=1}^m |\lambda^{xp}| \left(1 - \frac{\Delta t}{\Delta x} |\lambda^{xp}| \right) \mathcal{W}_{i-1/2,j}^p = |A| \left(I - \frac{\Delta t}{\Delta x} |A| \right) (Q_{ij} - Q_{i-1,j})$$

and

$$\begin{aligned} A^+Q_{i-1,j} + A^-Q_{ij} + \frac{1}{2} \sum_{p=1}^m |\lambda^{xp}| \left(1 - \frac{\Delta t}{\Delta x} |\lambda^{xp}| \right) \mathcal{W}_{i-1/2,j}^p \\ = A^+Q_{i-1,j} + A^-Q_{ij} + \frac{1}{2} |A| \left(I - \frac{\Delta t}{\Delta x} |A| \right) (Q_{ij} - Q_{i-1,j}) \\ = \frac{1}{2} A(Q_{i-1,j} + Q_{ij}) - \frac{1}{2} \frac{\Delta t}{\Delta x} A^2 (Q_{ij} - Q_{i-1,j}), \end{aligned} \quad (21.4)$$

which agrees with the corresponding terms in the Lax–Wendroff flux (19.14). On the other hand, if all waves are fully limited so that $\tilde{\mathcal{W}}_{i-1/2,j}^p = 0$, then these terms in the flux reduce to the Godunov flux (19.18).

Copyright © 2002, Cambridge University Press.
All rights reserved. May not be reproduced in any form without permission from the publisher, except fair uses permitted under U.S. or applicable copyright law.

21.2 The Wave-Propagation Approach to Accumulating Fluxes

To implement the method described above, we take an approach very similar to what was done in Sections 20.5 and 20.6 for the advection equation. We use the form (19.19), which means we need fluctuations and correction fluxes. We present the algorithm for computing each of these in a framework that easily extends to nonlinear systems of equations by using approximate Riemann solvers:

1. Initialize $\tilde{F}_{i-1/2,j} = 0$ and $\tilde{G}_{i,j-1/2} = 0$ at each interface.
2. Sweep through the grid, solving each Riemann problem in x . At the interface between cells $C_{i-1,j}$ and C_{ij} we use data $Q_{i-1,j}$ and Q_{ij} to compute waves $\mathcal{W}_{i-1/2,j}^p$ and speeds $s_{i-1/2,j}^p$. We also compute fluctuations $\mathcal{A}^- \Delta Q_{i-1/2,j}$ and $\mathcal{A}^+ \Delta Q_{i-1/2,j}$ exactly as in one space dimension. For the constant-coefficient linear case the \mathcal{W} and s will be eigenvectors and eigenvalues of A and we will have

$$\begin{aligned}\mathcal{A}^- \Delta Q_{i-1/2,j} &= \sum_{p=1}^m (s_{i-1/2,j}^p)^- \mathcal{W}_{i-1/2,j}^p = A^- \Delta Q_{i-1/2,j}, \\ \mathcal{A}^+ \Delta Q_{i-1/2,j} &= \sum_{p=1}^m (s_{i-1/2,j}^p)^+ \mathcal{W}_{i-1/2,j}^p = A^+ \Delta Q_{i-1/2,j}.\end{aligned}\quad (21.5)$$

3. The waves are limited to obtain $\tilde{\mathcal{W}}_{i-1/2,j}^p$ and these are used to update the correction fluxes at this interface:

$$\tilde{F}_{i-1/2,j} := F_{i-1/2,j} + \frac{1}{2} \sum_{p=1}^m |s_{i-1/2,j}^p| \left(1 - \frac{\Delta t}{\Delta x} |s_{i-1/2,j}^p| \right) \tilde{\mathcal{W}}_{i-1/2,j}^p. \quad (21.6)$$

4. The right-going fluctuation $\mathcal{A}^+ \Delta Q_{i-1/2,j}$ is used to compute an up-going transverse fluctuation $\mathcal{B}^+ \mathcal{A}^+ \Delta Q_{i-1/2,j}$ and a down-going transverse fluctuation $\mathcal{B}^- \mathcal{A}^+ \Delta Q_{i-1/2,j}$ by solving a *transverse Riemann problem*. We have seen an example of this for the advection equation $q_t + u q_x + v q_y = 0$ in Section 20.5, where $\mathcal{A}^+ \Delta Q_{i-1/2,j} = u_{i-1/2,j}^+ (Q_{ij} - Q_{i-1,j})$ and the transverse fluctuations are defined by (20.25) and (20.26),

$$\mathcal{B}^\pm \mathcal{A}^+ \Delta Q_{i-1/2,j} = v_{i,j \pm 1/2}^\pm u_{i-1/2,j}^+ (Q_{ij} - Q_{i-1,j}). \quad (21.7)$$

In general the symbols $\mathcal{B}^+ \mathcal{A}^+ \Delta Q$ and $\mathcal{B}^- \mathcal{A}^+ \Delta Q$ each represent a single m -vector obtained by some decomposition of the fluctuation $\mathcal{A}^+ \Delta Q$. The notation is motivated by the linear case, in which case we want

$$\mathcal{B}^\pm \mathcal{A}^+ \Delta Q_{i-1/2,j} = B^\pm A^+ (Q_{ij} - Q_{i-1,j}). \quad (21.8)$$

In the linear system case these are computed by decomposing the fluctuation $\mathcal{A}^+ \Delta Q_{i-1/2,j}$ into eigenvectors of B ,

$$\mathcal{A}^+ \Delta Q_{i-1/2,j} = \sum_{p=1}^m \beta^p r^{yp},$$

and then setting

$$\mathcal{B}^\pm \mathcal{A}^+ \Delta Q_{i-1/2,j} = \sum_{p=1}^m (\lambda^{yp})^\pm \beta^p r^{yp}. \quad (21.9)$$

This *wave decomposition* of $\mathcal{A}^+ \Delta Q_{i-1/2,j}$ can be viewed as solving a second Riemann problem in the transverse direction, even though it is not based on left and right states as we normally interpret a Riemann solver. The net contribution of all right-going waves is split up into up-going and down-going parts based on the eigenvectors corresponding to plane waves in the y -direction.

5. These fluctuations $\mathcal{B}^\pm \mathcal{A}^+ \Delta Q_{i-1/2,j}$ are used to update the correction fluxes above and below cell \mathcal{C}_{ij} :

$$\begin{aligned} \tilde{G}_{i,j+1/2} &:= \tilde{G}_{i,j+1/2} - \frac{\Delta t}{2 \Delta x} \mathcal{B}^+ \mathcal{A}^+ \Delta Q_{i-1/2,j}, \\ \tilde{G}_{i,j-1/2} &:= \tilde{G}_{i,j-1/2} - \frac{\Delta t}{2 \Delta x} \mathcal{B}^- \mathcal{A}^+ \Delta Q_{i-1/2,j}. \end{aligned} \quad (21.10)$$

6. In a similar manner, the left-going fluctuation $\mathcal{A}^- \Delta Q_{i-1/2,j}$ is split into transverse fluctuations $\mathcal{B}^\pm \mathcal{A}^- \Delta Q_{i-1/2,j}$, which are then used to update the fluxes above and below cell $\mathcal{C}_{i-1,j}$:

$$\begin{aligned} \tilde{G}_{i-1,j+1/2} &:= \tilde{G}_{i-1,j+1/2} - \frac{\Delta t}{2 \Delta x} \mathcal{B}^+ \mathcal{A}^- \Delta Q_{i-1/2,j}, \\ \tilde{G}_{i-1,j-1/2} &:= \tilde{G}_{i-1,j-1/2} - \frac{\Delta t}{2 \Delta x} \mathcal{B}^- \mathcal{A}^- \Delta Q_{i-1/2,j}. \end{aligned} \quad (21.11)$$

Note that these updates to nearby \tilde{G} fluxes are exactly analogous to what was done in (20.24) for the scalar advection equation.

7. Steps 2–6 are now repeated for each Riemann problem in y , at interfaces between cells $\mathcal{C}_{i,j-1}$ and \mathcal{C}_{ij} . The resulting waves $\mathcal{W}_{i,j-1/2}$ are limited by comparisons in the y -direction and used to update $\tilde{G}_{i,j-1/2}$. In solving these Riemann problems we also compute fluctuations $\mathcal{B}^\pm \Delta Q_{i,j-1/2}$, which are then split transversely into $\mathcal{A}^\pm \mathcal{B}^+ \Delta Q_{i,j-1/2}$ and $\mathcal{A}^\pm \mathcal{B}^- \Delta Q_{i,j-1/2}$. These four transverse fluctuations are used to modify four nearby \tilde{F} fluxes, as was done in (20.25) for the advection equation.
8. Finally, the updating formula (19.19) is applied to advance by time Δt ,

$$\begin{aligned} Q_{ij}^{n+1} &= Q_{ij} - \frac{\Delta t}{\Delta x} (\mathcal{A}^+ \Delta Q_{i-1/2,j} + \mathcal{A}^- \Delta Q_{i+1/2,j}) \\ &\quad - \frac{\Delta t}{\Delta y} (\mathcal{B}^+ \Delta Q_{i,j-1/2} + \mathcal{B}^- \Delta Q_{i,j+1/2}) \\ &\quad - \frac{\Delta t}{\Delta x} (\tilde{F}_{i+1/2,j} - \tilde{F}_{i-1/2,j}) - \frac{\Delta t}{\Delta y} (\tilde{G}_{i,j+1/2} - \tilde{G}_{i,j-1/2}). \end{aligned} \quad (21.12)$$

21.3 CLAWPACK Implementation

This wave-propagation algorithm is implemented in CLAWPACK by assuming that the user has provided two Riemann solvers. One, called `rpn2`, solves the Riemann problem normal to any cell interface and is similar to the one-dimensional Riemann solver `rp1`. For dimensional splitting only this one Riemann solver is needed. The new Riemann solver needed for the unsplit algorithm is called `rpt2` and solves Riemann problems of the sort just described in the transverse direction.

Each Riemann solver must be capable of solving the appropriate Riemann problem in either the x -direction or the y -direction, depending on which edge of the cell we are working on. The computations are organized by first taking sweeps in the x -direction along each row of the grid and then sweeps in the y -direction along each column. In each sweep a one-dimensional slice of the data is passed into the Riemann solver `rpn2`, so that `q1` and `qr` in this routine are exactly analogous to `q1` and `qr` in the one-dimensional Riemann solver `rp1`. A flag `ixy` is also passed in to indicate whether this is an x -slice (if `ixy=1`) or a y -slice (if `ixy=2`) of the data. The corresponding one-dimensional slice of the auxiliary array is also passed in.

The subroutine returns vectors of fluctuations (`amdq`, `apdq`) and waves and speeds (`wave`, `s`) obtained by solving the one-dimensional Riemann problem at each interface along the slice, as in `rp1`. In order to use the dimensional-splitting methods described in Section 19.5, only this Riemann solver `rpn2` is required.

To perform the transverse Riemann solves required in the multidimensional wave-propagation algorithms, each of the fluctuations `amdq` ($= \mathcal{A}^- \Delta Q$) and `apdq` ($= \mathcal{A}^+ \Delta Q$) must be passed into the transverse solver `rpt2`, so this routine is called twice. Within the subroutine this parameter is called `asdq` ($= \mathcal{A}^* \Delta Q$), and a parameter `imp` indicates which fluctuation this is (`imp=1` if `asdq = \mathcal{A}^- \Delta Q`, and `imp=2` if `asdq = \mathcal{A}^+ \Delta Q`). For many problems the subroutine's action may be independent of the value of `imp`. In particular, for a constant-coefficient linear system the vector $\mathcal{A}^* \Delta Q$ is simply decomposed into eigenvectors of B in either case. For a variable-coefficient problem, however, the matrix B may be different to the left and right of the interface, and so the decomposition may depend on which direction the fluctuation is propagating.

The routine `rpt2` returns `bmasdq` ($= \mathcal{B}^- \mathcal{A}^* \Delta Q$) and `bpasdq` ($= \mathcal{B}^+ \mathcal{A}^* \Delta Q$), the splitting of this fluctuation in the transverse direction. These terms are used to update the correction fluxes \tilde{G} nearby.

The same routine is used during the y -sweeps to split $\mathcal{B}^\pm \Delta Q$ into $\mathcal{A}^\pm \mathcal{B}^\pm \Delta Q$, so when `ixy=2` it is important to realize that the input parameter `asdq` represents either $\mathcal{B}^- \Delta Q$ or $\mathcal{B}^+ \Delta Q$, while the outputs `bmasdq` and `bpasdq` now represent $\mathcal{A}^\pm \mathcal{B}^\pm \Delta Q$. Similarly, in `rpn2` the parameter `asdq` represents $\mathcal{A}^* \Delta Q$ in the x -sweeps, as described above, and represents $\mathcal{B}^* \Delta q$ in the y -sweeps. It may be easiest to simply remember that in these routines “a” always refers to the normal direction and “b” to the transverse direction.

For many systems of equations the Riemann solver for the x -sweeps and y -sweeps take a very similar form, especially if the equations are isotropic and have exactly the same form in any direction (as is the case for many physical systems such as acoustics, shallow water, or gas dynamics). Then the cases `ixy=1` and `ixy=2` may be distinguished only by which component of q represents the normal velocity and which is the transverse velocity. This is

the reason that a single Riemann solver `rpn2` with a flag `ixy` is required rather than separate Riemann solvers in the x - and y -directions.

The parameter values `method(2)` and `method(3)` determine what method is used in CLAWPACK. If `method(2)=1` then the first-order updates are used but the second-order corrections based on limited waves are not used, i.e., step 3 in the above algorithm is skipped. If `method(3)=0` then no transverse propagation is done, i.e., steps 4–6 are skipped. If `method(3)=1` then these steps are performed. If `method(3)=2` then an additional improvement is made to the algorithm, in which the correction terms from step 3 are also split in the transverse direction. This is accomplished by applying the transverse solver `rpt2` to the vectors

$$\mathcal{A}^- \Delta Q_{i-1/2,j} + \sum_{p=1}^m |s_{i-1/2,j}^p| \left(1 - \frac{\Delta t}{\Delta x} |s_{i-1/2,j}^p| \right) \tilde{W}_{i-1/2,j}^p$$

and

$$\mathcal{A}^+ \Delta Q_{i-1/2,j} - \sum_{p=1}^m |s_{i-1/2,j}^p| \left(1 - \frac{\Delta t}{\Delta x} |s_{i-1/2,j}^p| \right) \tilde{W}_{i-1/2,j}^p$$

instead of to $\mathcal{A}^- \Delta Q_{i-1/2,j}$ and $\mathcal{A}^+ \Delta Q_{i-1/2,j}$. The rationale for this is explained in [283].

If `method(3)<0` is specified, then dimensional splitting is used instead of this unsplit algorithm, as has already been described in Section 19.5.1.

See the CLAWPACK *User Guide* and sample programs for more description of the normal and transverse Riemann solvers. As an example we consider the acoustics equations in the next section. The CLAWPACK Riemann solver for this system may be found in [`claw/book/chap21/acoustics`].

21.4 Acoustics

As an example, consider the two-dimensional acoustics equations (18.24) with no background flow ($u_0 = v_0 = 0$). In this case the eigenvectors of A and B are given in (18.31) and (18.32), and the eigenvalues of each are $\lambda^{x1} = -c_0$, $\lambda^{x2} = 0$, and $\lambda^{x3} = c_0$.

The Riemann solver `rpn2` must solve the Riemann problem $q_t + A q_x = 0$ in the x -direction when `ixy=1` or $q_t + B q_y = 0$ in the y -direction when `ixy=2`.

If `ixy=1`, then we decompose $\Delta Q_{i-1/2,j} = Q_{ij} - Q_{i-1,j}$ as

$$\Delta Q = \alpha^1 r^{x1} + \alpha^2 r^{x2} + \alpha^3 r^{x3}, \quad (21.13)$$

where the eigenvectors are given in (18.31). For clarity the subscript $i - 1/2, j$ has been dropped from ΔQ and also from the coefficients α , which are different at each interface of course.

Solving the linear system (21.13) for α yields

$$\begin{aligned} \alpha^1 &= \frac{-\Delta Q^1 + Z_0 \Delta Q^2}{2Z_0}, \\ \alpha^2 &= \Delta Q^3, \\ \alpha^3 &= \frac{\Delta Q^1 + Z_0 \Delta Q^2}{2Z_0}. \end{aligned} \quad (21.14)$$

The waves are then given by

$$\mathcal{W}^1 = \alpha^1 \begin{bmatrix} -Z_0 \\ 1 \\ 0 \end{bmatrix}, \quad \mathcal{W}^2 = \alpha^2 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad \mathcal{W}^3 = \alpha^3 \begin{bmatrix} Z_0 \\ 1 \\ 0 \end{bmatrix}, \quad (21.15)$$

and the corresponding wave speeds are $s^1 = -c_0$, $s^2 = 0$, and $s^3 = c_0$.

The fluctuations are then given by

$$\mathcal{A}^- \Delta Q = s^1 \mathcal{W}^1, \quad \mathcal{A}^+ \Delta Q = s^3 \mathcal{W}^3.$$

Note that the 2-wave makes no contribution to these fluctuations or to the second-order correction terms, where the contribution is also weighted by s^2 , so the implementation can be made slightly more efficient by propagating only the 1-wave and 3-wave. (This is done in [claw/book/chap21/acoustics], where `mwaves=2` is used.) If there were a nonzero background flow (u_0, v_0) , then the wave speeds would be $s^1 = u_0 - c_0$, $s^2 = u_0$, and $s^3 = u_0 + c_0$. In this case it would be necessary to use all three waves and consider the sign of each s^p in computing the fluctuations.

If $ixy=2$ then we are sweeping in the y -direction. We then need to decompose $\Delta Q = \Delta Q_{i,j-1/2} = Q_{ij} - Q_{i,j-1}$ as

$$\Delta Q = \alpha^1 r^{y1} + \alpha^2 r^{y2} + \alpha^3 r^{y3},$$

where the eigenvectors are given in (18.32). This yields

$$\begin{aligned} \alpha^1 &= \frac{-\Delta Q^1 + Z_0 \Delta Q^3}{2Z_0}, \\ \alpha^2 &= \Delta Q^2, \\ \alpha^3 &= \frac{\Delta Q^1 + Z_0 \Delta Q^3}{2Z_0}. \end{aligned} \quad (21.16)$$

The waves are

$$\mathcal{W}^1 = \alpha^1 \begin{bmatrix} -Z_0 \\ 0 \\ 1 \end{bmatrix}, \quad \mathcal{W}^2 = \alpha^2 \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \mathcal{W}^3 = \alpha^3 \begin{bmatrix} Z_0 \\ 0 \\ 1 \end{bmatrix}, \quad (21.17)$$

and the corresponding wave speeds are $s^1 = -c_0$, $s^2 = 0$, and $s^3 = c_0$. The fluctuations are then

$$\mathcal{B}^- \Delta Q = s^1 \mathcal{W}^1, \quad \mathcal{B}^+ \Delta Q = s^3 \mathcal{W}^3,$$

but recall that in the CLAWPACK Riemann solver these are again denoted by `amdq` and `apdq`.

Note that these formulas are essentially the same for each value of ixy , except that the roles of the second and third components of Q are switched, depending on which velocity u or v is the velocity normal to the interface. In the CLAWPACK Riemann solver

EBSCO : eBook Collection (EBSCONHOST) - printed on 2/28/2018 4:06 PM via FLORIDA INST OF TECHNOLOGY

AN: 112638 ; LeVeque, Randall J.; Finite Volume Methods for Hyperbolic Problems

Account: flit.main.ehost

[claw/book/chap21/acoustics/rpn2ac.f], this is easily accomplished by using indices

$$\text{mu} = \begin{cases} 2 & \text{if } \text{ixy} = 1, \\ 3 & \text{if } \text{ixy} = 2, \end{cases} \quad \text{mv} = \begin{cases} 3 & \text{if } \text{ixy} = 1, \\ 2 & \text{if } \text{ixy} = 2 \end{cases} \quad (21.18)$$

for the normal (mu) and transverse (mv) components of Q .

In fact, these formulas are easily generalized to solve a Riemann problem at any angle to the x and y axes. This is discussed in Section 23.6, where acoustics on a general quadrilateral grid is discussed.

The transverse Riemann solver `rpt2` must take a fluctuation $\mathcal{A}^* \Delta Q$ and split it into $\mathcal{B}^- \mathcal{A}^* \Delta Q$ and $\mathcal{B}^+ \mathcal{A}^* \Delta Q$, or take a fluctuation $\mathcal{B}^* \Delta q$ and split it into $\mathcal{A}^- \mathcal{B}^* \Delta Q$ and $\mathcal{A}^+ \mathcal{B}^* \Delta Q$. This requires another splitting into eigenvectors of these matrices and multiplication by the corresponding eigenvalues. This is described in the next section for the more general problem of acoustics in heterogeneous media. For the constant-coefficient case, see also the simpler transverse Riemann solver [claw/book/chap21/acoustics/rpt2ac.f].

21.5 Acoustics in Heterogeneous Media

In Section 21.4 the normal and transverse Riemann solvers for acoustics in a homogeneous material were discussed. In this section we extend this to the case of a heterogeneous material, where the density $\rho(x, y)$ and the bulk modulus $K(x, y)$ may vary in space. The one-dimensional case has been studied in Section 9.6, and here we develop the two-dimensional generalization.

As in one dimension, the linear hyperbolic system can be solved in the nonconservative form

$$q_t + A(x, y)q_x + B(x, y)q_y = 0, \quad (21.19)$$

where

$$q = \begin{bmatrix} p \\ u \\ v \end{bmatrix}, \quad A = \begin{bmatrix} 0 & K(x, y) & 0 \\ 1/\rho(x, y) & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 & K(x, y) \\ 0 & 0 & 0 \\ 1/\rho(x, y) & 0 & 0 \end{bmatrix}. \quad (21.20)$$

This equation is not in conservation form, but can still be handled by high-resolution methods if we use the fluctuation form. The solution to the Riemann problem normal to each cell interface is computed exactly as in the one-dimensional case of Section 9.6. Let ρ_{ij} and c_{ij} be the density and sound speed in the (i, j) cell, where $c_{ij} = \sqrt{K_{ij}/\rho_{ij}}$. Then the Riemann problem at the $(i - 1/2, j)$ edge, for example, gives

$$\mathcal{W}^1 = \alpha^1 \begin{bmatrix} -Z_{i-1,j} \\ 1 \\ 0 \end{bmatrix}, \quad \mathcal{W}^2 = \alpha^2 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad \mathcal{W}^3 = \alpha^3 \begin{bmatrix} Z_{ij} \\ 1 \\ 0 \end{bmatrix},$$

where $Z_{ij} = \rho_{ij}c_{ij}$ is the impedance in the (i, j) cell, and

$$\begin{aligned}\alpha^1 &= \frac{-\Delta Q^1 + Z_{ij} \Delta Q^2}{Z_{i-1,j} + Z_{ij}}, \\ \alpha^2 &= \Delta Q^3, \\ \alpha^3 &= \frac{\Delta Q^1 + Z_{i-1,j} \Delta Q^2}{Z_{i-1,j} + Z_{ij}}.\end{aligned}\quad (21.21)$$

The subscript $i - 1/2, j$ has been omitted from α and ΔQ here for clarity.

These reduce to (21.14) in the case of constant impedance. As usual, the fluctuations $\mathcal{A}^- \Delta Q$ and $\mathcal{A}^+ \Delta Q$ are given by the product of the waves and wave speeds,

$$\mathcal{A}^- \Delta Q_{i-1/2,j} = s_{i-1/2,j}^1 \mathcal{W}_{i-1/2,j}^1, \quad \mathcal{A}^+ \Delta Q_{i-1/2,j} = s_{i-1/2,j}^3 \mathcal{W}_{i-1/2,j}^3,$$

where $s_{i-1/2,j}^1 = -c_{i-1,j}$ and $s_{i-1/2,j}^3 = c_{ij}$ are the appropriate wave speeds.

21.5.1 Transverse Propagation

The right-going fluctuation $\mathcal{A}^+ \Delta Q$ is split into up-going and down-going fluctuations $\mathcal{B}^+ \mathcal{A}^+ \Delta Q$ and $\mathcal{B}^- \mathcal{A}^+ \Delta Q$ that modify the fluxes $\tilde{G}_{i,j+1/2}$ and $\tilde{G}_{i,j-1/2}$ above and below the cell (i, j) , respectively. To compute the down-going fluctuation $\mathcal{B}^- \mathcal{A}^+ \Delta Q$, for example, we need to decompose the vector $\mathcal{A}^+ \Delta Q$ into eigenvectors corresponding to up-going and down-going waves arising from the interface at $(i, j - 1/2)$,

$$\mathcal{A}^+ \Delta Q_{i-1/2,j} = \beta^1 \begin{bmatrix} -Z_{i,j-1} \\ 0 \\ 1 \end{bmatrix} + \beta^2 \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} + \beta^3 \begin{bmatrix} Z_{ij} \\ 0 \\ 1 \end{bmatrix}, \quad (21.22)$$

with speeds $-c_{i,j-1}, 0, c_{ij}$ respectively. Solving this linear system gives

$$\beta^1 = \frac{-(\mathcal{A}^+ \Delta Q_{i-1/2,j})^1 + (\mathcal{A}^+ \Delta Q_{i-1/2,j})^3 Z_{ij}}{Z_{i,j-1} + Z_{ij}}, \quad (21.23)$$

where $(\mathcal{A}^+ \Delta Q_{i-1/2,j})^p$ is the p th element of the vector $\mathcal{A}^+ \Delta Q_{i-1/2,j}$. The coefficient β^1 is the only one needed to compute the down-going fluctuation, which is obtained by multiplying the first wave in (21.22) by the speed of this down-going wave,

$$\mathcal{B}^- \mathcal{A}^+ \Delta Q_{i-1/2,j} = -c_{i,j-1} \beta^1 \begin{bmatrix} -Z_{i,j-1} \\ 0 \\ 1 \end{bmatrix}. \quad (21.24)$$

To compute the up-going fluctuation $\mathcal{B}^+ \mathcal{A}^+ \Delta Q$, we instead decompose the vector $\mathcal{A}^+ \Delta Q$ into eigenvectors corresponding to up-going and down-going waves arising from the

interface at $(i, j + 1/2)$,

$$\mathcal{A}^+ \Delta Q_{i-1/2,j} = \beta^1 \begin{bmatrix} -Z_{ij} \\ 0 \\ 1 \end{bmatrix} + \beta^2 \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} + \beta^3 \begin{bmatrix} Z_{i,j+1} \\ 0 \\ 1 \end{bmatrix}, \quad (21.25)$$

with speeds $-c_{ij}$, 0, $c_{i,j+1}$ respectively. Solving this linear system gives

$$\beta^3 = \frac{(\mathcal{A}^+ \Delta Q_{i-1/2,j})^1 + (\mathcal{A}^+ \Delta Q_{i-1/2,j})^3 Z_{i,j+1}}{Z_{ij} + Z_{i,j+1}}. \quad (21.26)$$

The coefficient β^3 is the only one needed to compute the up-going fluctuation, which is obtained by multiplying the third wave in (21.25) by the speed of this up-going wave,

$$\mathcal{B}^+ \mathcal{A}^+ \Delta Q_{i-1/2,j} = c_{i,j+1} \beta^3 \begin{bmatrix} Z_{i,j+1} \\ 0 \\ 1 \end{bmatrix}. \quad (21.27)$$

The left-going fluctuation $\mathcal{A}^- \Delta Q_{i-1/2,j}$ must similarly be decomposed in two different ways to compute the transverse fluctuations $\mathcal{B}^- \mathcal{A}^- \Delta Q_{i-1/2,j}$ and $\mathcal{B}^+ \mathcal{A}^- \Delta Q_{i-1/2,j}$. The formulas are quite similar with i replaced by $i - 1$ in the sound speeds c and impedances Z above. See [claw/book/chap21/corner/rpt2acv.f].

Example 21.1. Figure 21.1(a) shows a heterogeneous medium with piecewise constant density and bulk modulus. Figure 21.2 shows the calculation of an acoustic pulse propagating in this medium. The pulse is initially a square rightward-propagating plane-wave pulse in pressure, as indicated in Figure 21.1(b). The pressure perturbation is nonzero only for

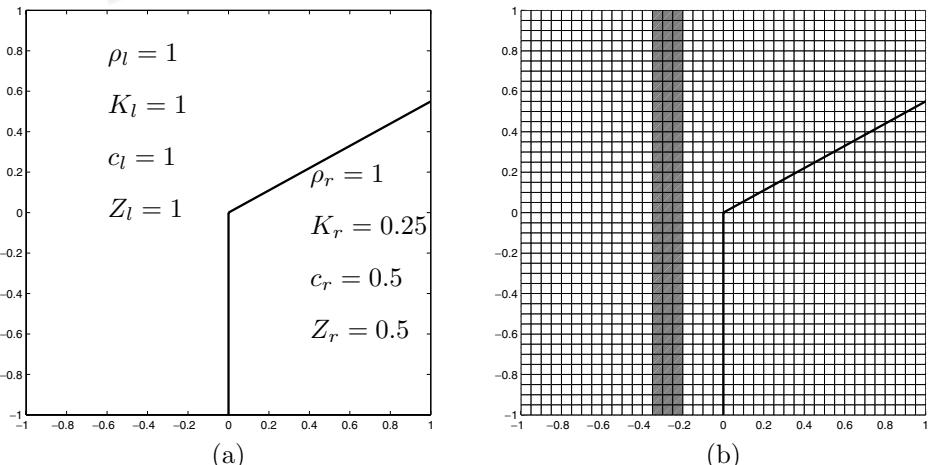
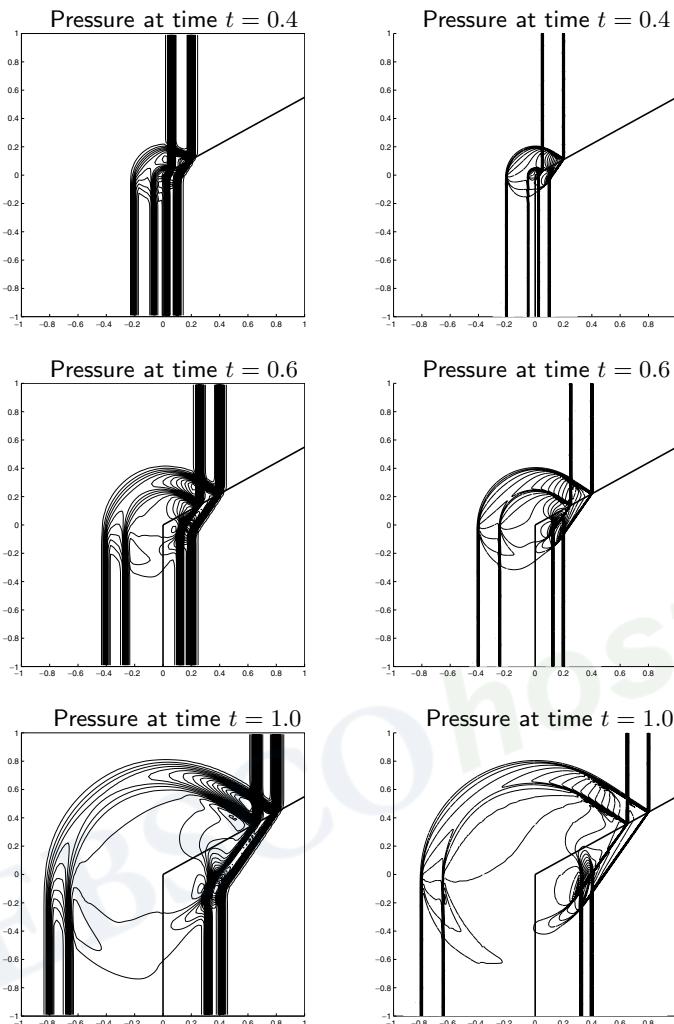


Fig. 21.1. (a) Piecewise-constant heterogeneous material for Example 21.1. (b) Illustration of how the interface cuts through a Cartesian grid, and the initial pressure pulse.



Copyright © 2002, Cambridge University Press.
All rights reserved. May not be reproduced in any form without permission from the publisher, except fair uses permitted under U.S. or applicable copyright law.

Fig. 21.2. Contours of pressure for an acoustic pulse propagating in the material shown in Figure 21.1, at three different times (from bottom to top). The calculation on the left was done on a 100×100 uniform Cartesian grid. [claw/book/chap21/corner] The calculation on the right is highly resolved using adaptive mesh refinement. [claw/book/chap21/corner/amr]

$-0.35 < x < -0.2$. When the pulse hits the interface, it is partially reflected and partially transmitted. As the pulse moves up the ramp portion of the interface, observe that the usual law of reflection is satisfied: the angle of incidence of the original pulse is equal to the angle of reflection. The transmitted wave is also oblique to the grid, at an angle determined by Snell's law that depends on the difference in wave speeds between the two media.

Two calculations are shown in Figure 21.2: a coarse-grid calculation on a 100×100 grid on the left, and a highly refined adaptive mesh refinement (AMR) calculation on the right. Fine grids are used only where required near the discontinuities in pressure. To obtain the

same resolution on a uniform grid would require a 960×960 grid. The AMR code used for this computation is also part of CLAWPACK (AMRCLAW), and is described in [32].

These calculations were all performed on a Cartesian grid in spite of the fact that the interface cuts obliquely through the grid cells as illustrated in Figure 21.1(b). Values of the impedance and sound speed in each grid cell are determined by using appropriate averages of the density and bulk modulus in the cells and then computing Z and c from these. We first determine what fraction of the grid cell lies in each of the two materials, the *left* material with density ρ_l and bulk modulus K_l , and the *right* material with density ρ_r and bulk modulus K_r . If w_l, w_r is the fraction lying in each state, then we set

$$\rho_{ij} = w_l \rho_l + w_r \rho_r, \quad K_{ij} = (w_l/K_l + w_r/K_r)^{-1}. \quad (21.28)$$

We use the arithmetic average of the densities and the harmonic average of the bulk moduli, as suggested by the discussion of Section 9.14. We then set

$$c_{ij} = \sqrt{K_{ij}/\rho_{ij}}, \quad Z_{ij} = \rho_{ij} c_{ij}. \quad (21.29)$$

21.6 Transverse Riemann Solvers for Nonlinear Systems

We now consider a nonlinear conservation law $q_t + f(q)_x + g(q)_y = 0$ and will concentrate on the procedure we must perform at the interface between cells $(i-1, j)$ and (i, j) to split fluctuations $\mathcal{A}^\pm \Delta Q_{i-1/2,j}$ into $\mathcal{B}^\pm \mathcal{A}^\pm \Delta Q_{i-1/2,j}$. For the constant-coefficient linear problem we simply multiply by the matrices B^- and B^+ , but for a nonlinear system there is no single matrix B , but rather a Jacobian matrix $g'(q)$ that depends on the data. However, if we solve Riemann problems normal to each edge by using a linearized approximate Riemann solver, as discussed in Section 15.3, then this linear approach is easily extended to the nonlinear case. In solving the Riemann problem $q_t + f(q)_x = 0$ we determined a matrix \hat{A} so that the fluctuations are defined simply by multiplying $Q_{ij} - Q_{i-1,j}$ by \hat{A}^- and \hat{A}^+ . The matrix \hat{A} depends on certain averaged values obtained from the states $Q_{i-1,j}$ and Q_{ij} . To define the transverse Riemann solver we can now simply use these same averaged values to define a matrix \hat{B} that approximates $g'(q)$ near the interface. The transverse Riemann solver then returns

$$\begin{aligned} \mathcal{B}^- \mathcal{A}^* \Delta Q &= \hat{B}^- (\mathcal{A}^* \Delta Q), \\ \mathcal{B}^+ \mathcal{A}^* \Delta Q &= \hat{B}^+ (\mathcal{A}^* \Delta Q). \end{aligned} \quad (21.30)$$

This is illustrated in the next section for the shallow water equations. Examples for the Euler equations can be found on the webpage [\[claw/book/chap21/euler\]](http://claw/book/chap21/euler).

21.7 Shallow Water Equations

The numerical methods developed above for multidimensional acoustics can be extended easily to nonlinear systems such as the shallow water equations. For the dimensional-splitting method we only need a normal Riemann solver (`rpn2` in CLAWPACK). This is essentially identical to the one-dimensional Riemann problem for the shallow water equations with a passive tracer as discussed in Section 13.12.1. In practice an approximate

Riemann solver is typically used, e.g., the Roe solver developed in Section 15.3.3. This is very easily extended to the two-dimensional case. In the x -direction, for example, the velocity v does not affect the nonlinear waves, and so the Roe averages \bar{h} and \hat{u} are computed as in (15.32) and (15.35) respectively,

$$\bar{h} = \frac{1}{2}(h_l + h_r), \quad \hat{u} = \frac{\sqrt{h_l} u_l + \sqrt{h_r} u_r}{\sqrt{h_l} + \sqrt{h_r}}, \quad (21.31)$$

where $h_l = h_{i-1,j}$, $h_r = h_{ij}$, etc. We also need an average value \hat{v} , discussed below. The Roe matrix is then

$$\hat{A} = \begin{bmatrix} 0 & 1 & 0 \\ -\hat{u}^2 + g\bar{h} & 2\hat{u} & 0 \\ -\hat{u}\hat{v} & \hat{v} & \hat{u} \end{bmatrix}, \quad (21.32)$$

with eigenvalues and eigenvectors

$$\begin{aligned} \hat{\lambda}^{x1} &= \hat{u} - \hat{c}, & \hat{\lambda}^{x2} &= \hat{u}, & \hat{\lambda}^{x3} &= \hat{u} + \hat{c}, \\ \hat{r}^{x1} &= \begin{bmatrix} 1 \\ \hat{u} - \hat{c} \\ \hat{v} \end{bmatrix}, & \hat{r}^{x2} &= \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, & \hat{r}^{x3} &= \begin{bmatrix} 1 \\ \hat{u} + \hat{c} \\ \hat{v} \end{bmatrix}, \end{aligned} \quad (21.33)$$

where $\hat{c} = \sqrt{g\bar{h}}$. Entropy fixes and limiters are applied just as in one dimension.

For the average velocity \hat{v} it seems possible to simply use the Roe average

$$\hat{v} = \frac{\sqrt{h_l} v_l + \sqrt{h_r} v_r}{\sqrt{h_l} + \sqrt{h_r}} \quad (21.34)$$

and obtain good results in general. This does not, however, give a matrix \hat{A} that satisfies the usual requirement

$$\hat{A}(Q_r - Q_l) = f(Q_r) - f(Q_l) \quad (21.35)$$

for the Roe matrix. Choosing \bar{h} and \hat{u} as in (21.31) insures that the first two equations of the system (21.35) hold, but the third equation requires

$$-\hat{u}\hat{v}\delta^1 + \hat{v}\delta^2 + \hat{u}\delta^3 = h_r u_r v_r - h_l u_l v_l,$$

where $\delta = Q_r - Q_l$. This equation can be used to define \hat{v} , obtaining

$$\begin{aligned} \hat{v} &= \frac{(h_r u_r v_r - h_l u_l v_l) - \hat{u}(h_r u_r - h_l u_l)}{(h_r u_r - h_l u_l) - \hat{u}(h_r - h_l)} \\ &= \frac{a_l v_l + a_r v_r}{a_l + a_r}, \end{aligned} \quad (21.36)$$

where

$$a_l = h_l(\hat{u} - u_l), \quad a_r = h_r(u_r - \hat{u}).$$

A difficulty with this approach is that the denominator is zero whenever $u_l = u_r$, and this case must be handled separately. In practice the weighting (21.34) has been successfully used.

To use the method developed in Section 21.2, we must also provide a transverse Riemann solver, similar to the one developed in Section 18.4 for the two-dimensional acoustics equations. For the nonlinear shallow water equations we wish to take the right-going flux $\mathcal{A}^+ \Delta Q_{i-1/2,j}$, for example, and split it into an up-going part $\mathcal{B}^+ \mathcal{A}^+ \Delta Q_{i-1/2,j}$ and a down-going part $\mathcal{B}^- \mathcal{A}^+ \Delta Q_{i-1/2,j}$. As developed in Sections 21.1 through 21.6, the basic idea is to split the vector $\mathcal{A}^+ \Delta Q_{i-1/2,j}$ into eigenvectors of a matrix B approximating $g'(q)$. For this nonlinear system the Jacobian varies with q . However, we can use the Roe-averaged quantities \bar{h} , \bar{u} , and \bar{v} to define a natural approximate matrix \hat{B} to use for this decomposition. The eigenvalues and eigenvectors are as in (18.42) but with (h, u, v) replaced by the Roe averages. The formula (21.30) is then used to define the transverse fluctuations.

Normal and transverse Riemann solvers for the shallow water equations can be found in the directory [claw/book/chap21/radialdam]. This approach has been used in the example shown below.

21.7.1 A Radial Dam-Break Problem

Figure 21.3 shows a radial dam-break problem for the two-dimensional shallow water equations. The depth is initially $h = 2$ inside a circular dam and $h = 1$ outside. When the dam is removed, a shock wave travels radially outwards while a rarefaction wave moves inwards. This is similar to the structure of the one-dimensional dam-break Riemann problem. The fluid itself is moving outwards, and is accelerated either abruptly through the shock wave or smoothly through the rarefaction wave. Figure 21.4 shows the time evolution of both the depth and the radial momentum as a function of r , the distance from the origin, over a longer time period. This was computed by solving the one-dimensional equations

$$\begin{aligned} h_t + (hU)_r &= -\frac{hU}{r}, \\ (hU)_t + \left(hU^2 + \frac{1}{2}gh^2 \right)_r &= -\frac{hU^2}{r}, \end{aligned} \quad (21.37)$$

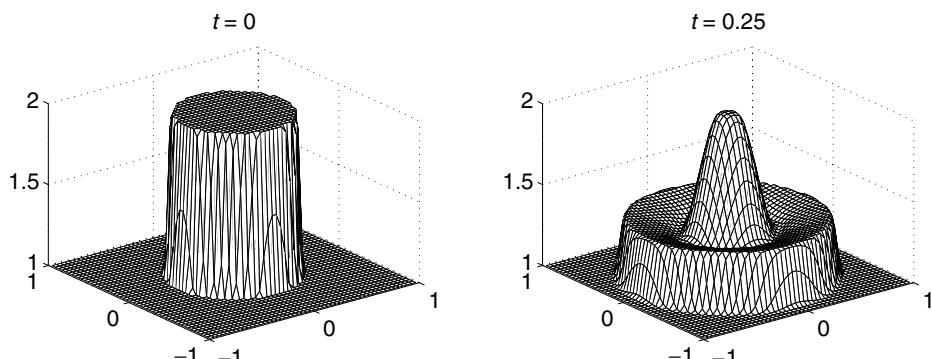


Fig. 21.3. Depth of water h for a radial dam-break problem, as computed on a 50×50 grid.

Copyright © 2002, Cambridge University Press.
All rights reserved. May not be reproduced in any form without permission from the publisher, except fair uses permitted under U.S. or applicable copyright law.

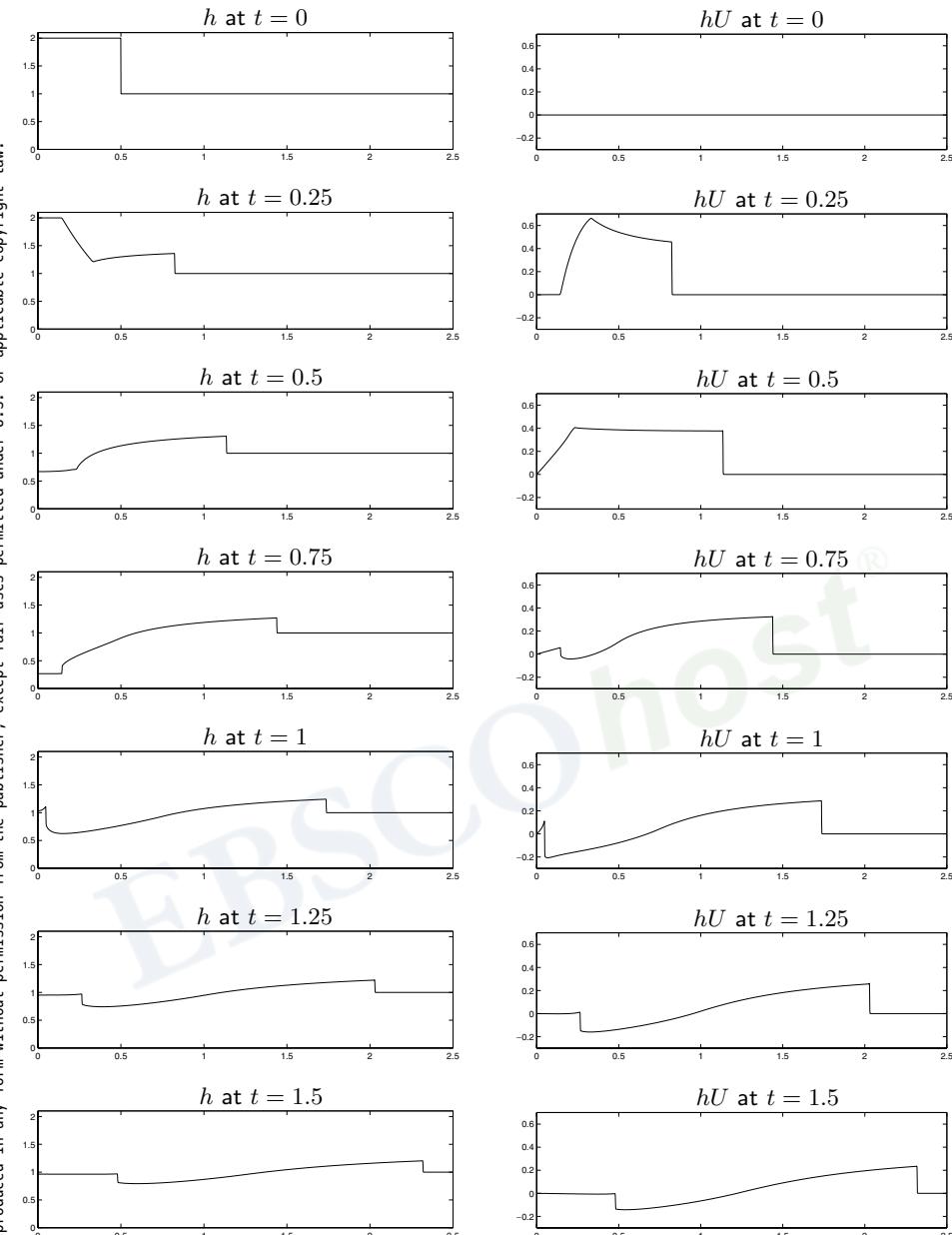


Fig. 21.4. Solution to the radial dam-break problem as a function of r . Left: depth h . Right: radial momentum hU . [claw/book/chap21/radialdam/1drad]

where $U(r, t)$ is the radial velocity. These follow from (18.52) with the hydrostatic pressure (18.39). Note that at time $t = 0.25$ the depth and momentum are no longer constant between the shock and rarefaction wave as in the one-dimensional Riemann problem. This is due to the source terms in (21.37), which physically arise from the fact that the fluid is spreading out and it is impossible to have constant depth and constant nonzero radial velocity.

Once the rarefaction wave hits the origin, all of the available fluid has been accelerated outwards. At this point the depth at the center begins to fall and ultimately falls below $h = 1$. At later times this depression in the water leads to inward acceleration of the fluid, filling this hole back in again. Note that after about $t = 0.75$ there is a region of negative radial momentum. As the fluid starts to flow inward, a second shock wave forms where the converging inward flow is decelerated back to zero velocity. This shock wave is already visible at time $t = 0.75$, and by $t = 1$ it has passed through the origin. At later times the structure has a basic N-wave form. The initially quiescent fluid is accelerated outwards through the leading shock, the velocity falls through a rarefaction wave to an inward velocity, and then the fluid is decelerated back to zero velocity through the second shock. These shocks weaken as they propagate outward, due to the radial spreading, and for large time will be essentially N-waves.

Figure 21.5 shows contour plots of numerical results computed using the unsplit method on a 125×125 grid over the domain $[-2.5, 2.5] \times [-2.5, 2.5]$, along with scatterplots of the computed solution vs. distance from the origin. This is a fairly coarse grid for this problem, the same resolution $\Delta x = \Delta y = 0.04$ as shown in Figure 21.3 but on a larger

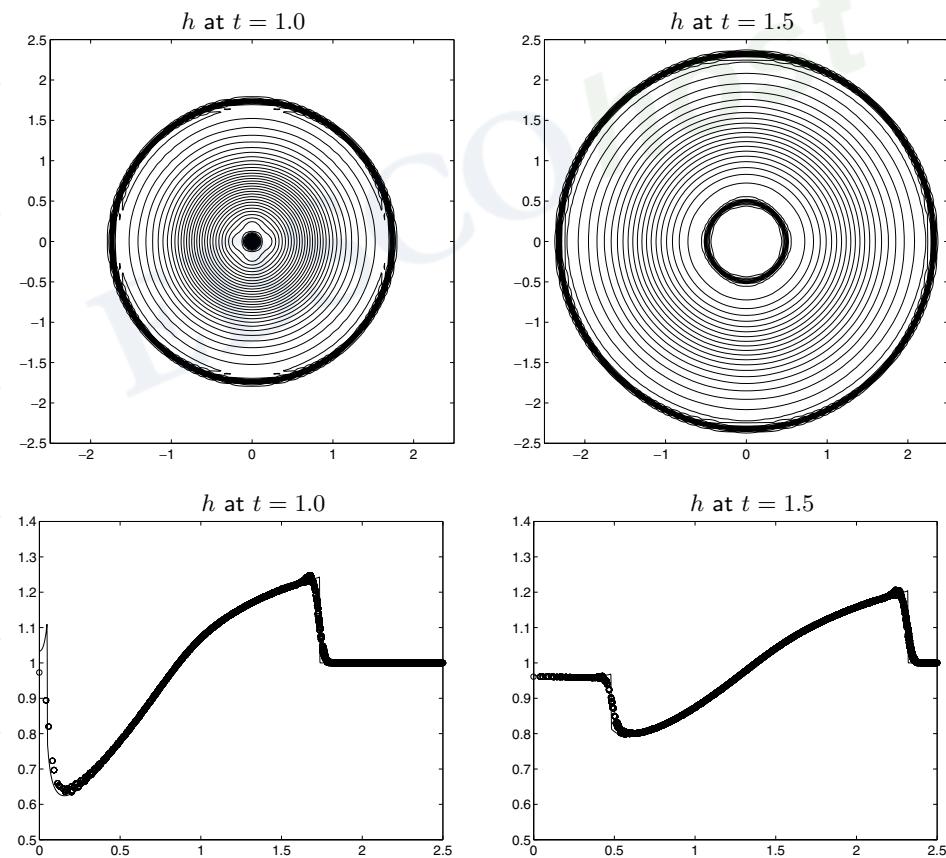


Fig. 21.5. Computed solutions for the radial dam-break problem. Top: contour plots of the depth at two times. Bottom: scatterplots of the depth vs. distance from the origin at the same two times. Contour levels are 0.61 : 0.02 : 1.31. [claw/book/chap21/radialdam]

domain. The depth is shown at two different times. At $t = 1.0$ the shock near the origin cannot be resolved on this grid. The structure looks correct elsewhere, however. At $t = 1.5$ the basic structure is captured well everywhere. Note in particular that the depth near the origin stabilizes at a value $h \approx 0.96$ that is well captured also in the two-dimensional results.

21.8 Boundary Conditions

Boundary conditions in two (or three) space dimensions can be handled in much the same way as in one dimension. The grid on the desired computational domain consists of *interior cells* that we will label by $i = 1, 2, \dots, m_x$ and $j = 1, 2, \dots, m_y$. This grid is extended by introducing a set of *ghost cells* on all sides, for $i = 1 - m_{BC}, \dots, 0$ and $i = m_x + 1, \dots, m_x + m_{BC}$, and for $j = 1 - m_{BC}, \dots, 0$ and $j = m_y + 1, \dots, m_y + m_{BC}$. Figure 21.6 shows a portion of such an extended grid for the case $m_{BC} = 2$. At the beginning of each time step the ghost-cell values are filled, based on data in the interior cells and the given boundary conditions, and then the algorithm of choice is applied over the extended domain. How many rows of ghost cells are needed depends on the stencil of the algorithm. The high-resolution algorithms presented in Chapters 20 and 21 generally require two rows of ghost cells as shown in the figure. This allows us to solve a Riemann problem at the boundary of the original domain and also one additional Riemann problem outside the domain. The waves from this Riemann problem do not enter the domain and so do not affect the solution directly, but are used to limit the waves arising from the original boundary. In the discussion below we assume $m_{BC} = 2$, but it should be clear how to extend each of the boundary conditions for larger values of m_{BC} .

The manner in which ghost cells are filled depends on the nature of the given boundary conditions. Periodic boundary conditions, for example, are easy to apply (as in the one-dimensional case of Section 7.1) by simply copying data from the opposite side of the grid. Below we will consider other standard cases, extending the approaches that were developed

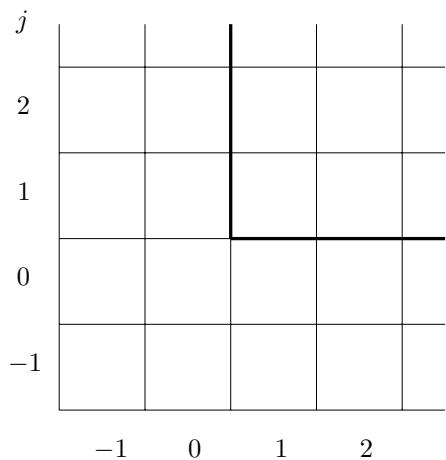


Fig. 21.6. The lower left corner of a typical computational domain is shown as the dark line. Interior grid cells are labeled with $i = 1, 2, \dots$ and $j = 1, 2, \dots$. The domain is extended with $m_{BC} = 2$ rows of ghost cells on each side.

in Chapter 7 to multidimensional problems. In particular, we consider solid-wall boundary conditions and the use of extrapolation at outflow boundaries. All of these are implemented in the default CLAWPACK routine [`claw/clawpack/2d/lib/bc2.f`].

21.8.1 Dimensional Splitting

We will primarily concentrate on the techniques needed when unsplit methods of the form (19.10) or (19.19) are used, in which the data Q^n is used to compute all fluctuations and fluxes in both the x - and y -directions. If a dimensional-splitting method is used, as described in Section 19.5, then additional issues arise. If we sweep first in the x -direction to obtain Q^* from Q^n , and then sweep in the y -direction to obtain Q^{n+1} from Q^* (using the Godunov splitting), we will need to specify boundary conditions for Q^* that may differ from the physical boundary conditions originally given for q . Some discussion of this point was given in Section 17.9 in the context of fractional-step methods for source terms. For dimensional splitting, appropriate ghost-cell values for Q^* can often be obtained by first extending Q^n to all ghost cells and then sweeping over the rows of ghost cells along the top and bottom of the grid ($j = -1, 0$ and $j = m_y + 1, m_y + 2$) as well as over the rows of interior cells ($j = 1, 2, \dots, m_y$). This modifies the ghost-cell values by solving the same one-dimensional equation as in the interior, and gives Q^* -values in these cells that can now be used as the ghost-cell values in the y -sweeps.

Note that if we wish to use the Strang splitting (19.29), then the ghost-cell values along the left and right edges ($i = -1, 0, m_x + 1, m_x + 2$) must also be updated to Q^* so that we can apply y -sweeps to these rows of cells as well as in the interior. Then we will have Q^{**} -values in these ghost cells, which are needed in taking the final x -sweep to obtain Q^{n+1} from Q^{**} . To do this requires that we have twice as many ghost cells (at least along the left and right boundaries), so that Q^* can be obtained in the ones where Q^{**} is ultimately needed.

21.8.2 Unsplit Wave-Propagation Algorithms

Even if unsplit algorithms of the form (19.19) are used, it may be necessary to sweep over the rows of ghost cells as well as the interior cells in order to properly implement the multidimensional algorithms. For example, the implementation of the wave-propagation algorithm discussed in Section 21.2 uses the solution to the Riemann problem between cells $C_{i-1,j}$ and C_{ij} to update the fluxes $\tilde{G}_{i-1,j+1/2}$ and $\tilde{G}_{i,j+1/2}$. Referring to Figure 21.6, we see that when $j = 0$ we must solve the Riemann problems in this row of ghost cells in order to obtain proper values of the fluxes $\tilde{G}_{i,1/2}$. These in turn are used to update the interior values Q_{i1} .

21.8.3 Solid Walls

In many of the problems we have considered (e.g., acoustics, shallow water equations, gas dynamics), the solution variables include velocity or momentum components in each spatial dimension. A common boundary condition is that the velocity normal to a wall should be zero, corresponding to a solid wall that fluid cannot pass through. As in one dimension, we

can implement this boundary condition by a suitable extension of the solution to the ghost cells. Consider the left edge of the domain, for example, where the x -component of the velocity should vanish. We first extrapolate all components in a symmetric manner, setting

$$Q_{0,j} = Q_{1,j}, \quad Q_{-1,j} = Q_{2,j} \quad \text{for } j = 1, 2, \dots, m_y. \quad (21.38)$$

We then negate the component of Q_{ij} (for $i = -1, 0$) that corresponds to the x -component of velocity or momentum. We perform a similar extension at the right boundary. At the bottom boundary we extrapolate

$$Q_{i,0} = Q_{i,1}, \quad Q_{i,-1} = Q_{i,2} \quad \text{for } i = -1, 0, 1, \dots, m_x + 1, m_x + 2, \quad (21.39)$$

and then negate the y -component of the velocity or momentum in these ghost cells. Note that we apply this procedure in the rows of ghost cells (e.g., $i = -1, 0$) as well as in the interior cells in order to insure that all the ghost cells shown in Figure 21.6 are filled, including the four corner cells where i and j both have values 0 or -1 . One should always insure that these corner cells are properly filled, especially if combinations of different boundary conditions are used at the two adjacent sides.

Note that in the procedure just outlined, the tangential component of velocity is simply extrapolated from the interior. For the problems we have considered (acoustics, shallow water, gas dynamics), it really doesn't matter what value the tangential velocity has in the ghost cells, since any jump in this quantity will propagate with zero normal velocity and will not affect the solution in the interior cells. This is due to the symmetry of the extrapolated data, which results in the contact discontinuity in the Riemann solution having zero velocity, properly mimicking a stationary wall. Hence any tangential velocity is allowed by these boundary conditions.

21.8.4 No-Slip Boundary Condition

In many fluid dynamics problems there is another physical boundary condition we might wish to impose at a solid wall. The *no-slip boundary condition* states that the tangential velocity should also vanish at the wall along with the normal velocity, so that fluid adjacent to the wall is stationary. This is expected due to friction between the wall and fluid molecules, which keeps molecules from slipping freely along the wall. However, this friction is present only in viscous fluids, and hyperbolic equations only model inviscid fluids, so we are not able to impose the no-slip condition in these models. If fluid viscosity is introduced, we obtain a parabolic equation (e.g., the Navier–Stokes equations instead of the inviscid Euler equations) that allows (in fact, requires) more boundary conditions to be specified.

If the physical viscosity of the fluid is very small relative to the typical fluid velocity away from the wall (i.e., if the *Reynolds number* is large), then there will often be a thin *boundary layer* adjacent to the wall in which the tangential velocity rapidly approaches the zero velocity of the wall. The thickness of this layer depends on the magnitude of the viscosity ϵ and often vanishes as $\epsilon \rightarrow 0$. As in the case of shock waves, the inviscid hyperbolic equation attempts to model the $\epsilon = 0$ limit.

In some applications this is a suitable approximation. For example, in many aerodynamics problems the thickness of the physical boundary layer on the surface of a body is much

smaller than a computational cell. For this reason the inviscid Euler equations are often used rather than the more expensive Navier–Stokes equations. However, caution must be used, since in some problems the viscous effects at the boundary do have a substantial influence on the global solution, even when the viscosity is very small. This is particularly true if the geometry is such that the boundary layer separates from the wall at some point (as must happen at the trailing edge of a wing, for example). Then the vorticity generated by a no-slip boundary will move away from the wall and perhaps lead to large-scale turbulence that persists even when ϵ is extremely small, but would not be seen in an ideal inviscid fluid. The numerical viscosity that is inherent in any “inviscid” algorithm can lead to similar effects, but may mimic flow at the wrong Reynolds number. A full discussion of these issues is beyond the scope of this book.

21.8.5 Extrapolation and Absorbing Boundary Conditions

For many problems we must use a computational domain that is smaller than the physical domain, particularly if we must cut off an essentially infinite domain at some point to obtain a finite computational domain, as already discussed in Section 7.3.1. We then wish to impose boundary conditions that allow us to compute on this smaller domain and obtain results that agree well with what would be computed on a larger domain. If the computational domain is large enough for the problem of interest, then we expect that there will only be outgoing waves at the boundary of this domain. There should not be substantial incoming waves unless they have a known form (as from some known external source) that can be imposed as part of the boundary conditions, as was done in Section 7.3.2 in one dimension. Here we will assume there should be no incoming waves, in which case our goal is to impose boundary conditions on the computational domain that are *nonreflecting*, or *absorbing*, and that allow any outgoing waves to disappear without generating spurious incoming waves.

It may be that the outgoing waves should interact outside the computational domain in such a way that incoming waves are generated, which should appear at the boundary at a later time. In general we cannot hope to model such processes via the boundary conditions, and this would be an indication that our computational domain is simply not large enough to capture the full problem.

In one space dimension, we saw in Section 7.3.1 that quite effective absorbing boundary conditions can be obtained simply by using zero-order extrapolation. This idea can be extended easily to more dimensions as well. For example, along the left and bottom edge we would set

$$Q_{0j} = Q_{1j}, \quad Q_{-1,j} = Q_{1j} \quad \text{for } j = 1, 2, \dots, m_y, \quad (21.40)$$

and then

$$Q_{i0} = Q_{i1}, \quad Q_{i,-1} = Q_{i1} \quad \text{for } i = -1, 0, \dots, m_x + 2. \quad (21.41)$$

Note that we have filled the corner ghost cells in Figure 21.6 as well as the edge ghost cells. The value obtained in all four of the corner cells is Q_{11} . The same value would be obtained if we reversed the order above and first extrapolated in y and then in x .

This simple approach to absorbing boundary conditions often works very well in multi-dimensional problems. As in one dimension, its success rests on the fact that the Riemann problem at the edge of the computational domain has the same data on either side, resulting in zero-strength waves and in particular no incoming waves.

While surprisingly effective, this approach is unfortunately not quite as effective as in one dimension, except in the special case of plane waves exiting normal to the boundary of the domain. An outgoing wave at some angle to the grid can be viewed as a superposition of various waves moving in the x - and y -directions. Some of these waves should be incoming from the perspective of the computational boundary. This is clear from the fact that solving a Riemann problem in x or y in the midst of such an oblique wave will result in nontrivial waves moving with both positive and negative speeds. Using zero-order extrapolation will result in the loss of some of this information. The fact that there are no incoming waves normal to the boundary results in an incorrect representation of the outgoing oblique wave, which appears computationally as an incoming reflected wave. The strength of this reflection

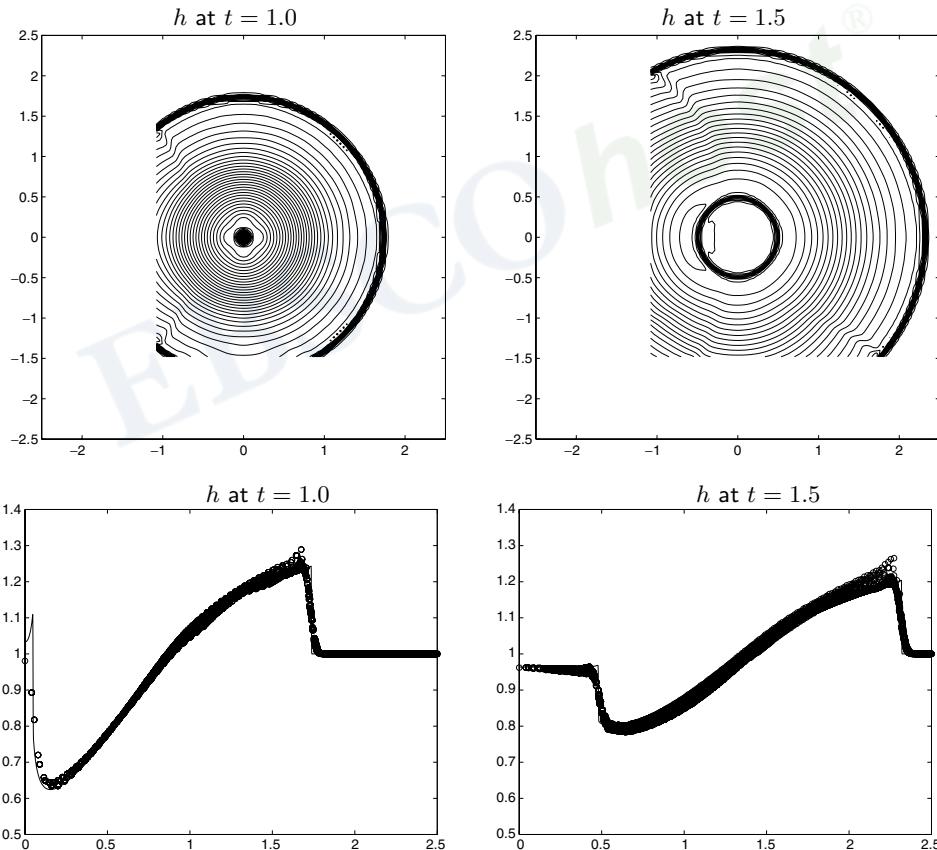


Fig. 21.7. Computed solutions for the radial dam-break problem on a reduced domain $[-1.1, 2.5] \times [-1.5, 2.5]$ with zero-order extrapolation boundary conditions. Top: contour plots of the depth at two times. Bottom: scatterplots of the depth vs. distance from the origin at the same two times. Compare Figure 21.5, where the same resolution has been used on a larger domain. Contour levels are $0.61 : 0.02 : 1.31$. [claw/book/chap21/radialdamabc]

generally depends on the angle of the wave to the boundary and is typically worst in corners. This is illustrated in Example 21.2 below, which shows the effects of this error. Still, these boundary conditions are fairly effective considering their simplicity, and are often good enough in practice.

There is an extensive literature on more sophisticated approaches to specifying absorbing boundary conditions for wave-propagation problems. See, for example, [1], [21], [28], [117], [123], [176], [197], [230].

Example 21.2. As an example, consider the radial dam-break problem for the shallow water equations described in Section 21.7.1. We now solve this same problem on a 90×100 grid in a smaller domain $[-1.1, 2.5] \times [-1.5, 2.5]$ rather than the full domain $[-2.5, 2.5] \times [-2.5, 2.5]$ used to compute the results shown in Figure 21.5. The same mesh size $\Delta x = \Delta y = 0.04$ is used. As seen in the contour plots of Figure 21.7, small-amplitude reflected waves are generated as the shock wave leaves the computational domain. The effect of these errors is also observed in the scatterplots of depth vs. distance from the origin.

Elastic Waves

A brief introduction to one-dimensional elasticity theory and elastic wave propagation was given in Section 2.12. In this chapter we will explore the full three-dimensional elasticity equations in the context of elastic wave propagation, or *elastodynamics*. There are many references available on the basic theory of linear and nonlinear elastodynamics (e.g., [6], [11], [141], [249], [255], [367], [422]), though often not in the first-order hyperbolic form we need. In this chapter the equations, eigenstructure, and Riemann solutions are written out in detail for several different variants of the linear problem.

The notation and terminology for these equations differs widely between different fields of application. Much of the emphasis in the literature is on steady-state problems, or *elastostatics*, in which the goal is to determine the deformation of an object and the internal stresses that result from some applied force. These boundary-value problems are often posed as second-order or fourth-order elliptic equations. We will concentrate instead on the hyperbolic nature of the first-order time-dependent problem, and the eigenstructure of this system. This is important in many wave-propagation applications such as seismic modeling in the earth or the study of ultrasound waves propagating through biological tissue. For small deformations, linear elasticity can generally be used. But even this case can be challenging numerically, since most practical problems involve heterogeneous materials and complicated geometry. High-resolution finite volume methods are well suited to these problems, since interfaces between different materials are handled naturally in the process of solving Riemann problems. This has already been explored in one dimension in Section 9.6. These methods can also be extended to nonlinear elasticity equations by incorporating an appropriate nonlinear Riemann solver, allowing the solution of problems with *finite deformations* (meaning larger than infinitesimal), in which case shock waves can form. For even larger deformations plastic behavior is observed, which can also be modeled with hyperbolic systems in some cases. For some examples of the application of hyperbolic theory and Riemann solvers to elastic and elastic–plastic problems, see for example [8], [30], [55], [83], [85], [164], [273], [327], [360], [406], [454], [455], [456]. Here we restrict our attention to linear elastic problems.

Recall from Section 2.12 that there are generally two basic types of waves that can propagate in an elastic solid, P-waves (pressure waves or primary waves) and S-waves (shear waves or secondary waves). In one dimension the P-waves and S-waves can be modeled separately by disjoint systems of two equations each. In multidimensional problems there is a coupling between these modes and the situation is more complicated. However, we

will see that a plane-wave problem in any of the three coordinate directions leads to a system that decouples into the simple structure seen in Section 2.12.4. This means that finite volume methods based on Riemann solvers in the coordinate directions are easy to apply.

22.1 Derivation of the Elasticity Equations

In this section we will informally derive the full three-dimensional elasticity equations. The clear discussion of Davis and Selvadurai [102] has largely motivated the derivation given here, but other derivations and discussion of elastodynamics can be found in many sources, such as those listed above.

We first generalize the notation of Section 2.12.1 from two to three dimensions. The displacement $\vec{\delta}(x, y, z, t)$ now has three components, and $\nabla \vec{\delta}$ is a 3×3 matrix. The strain tensor ϵ is again defined by

$$\epsilon = \frac{1}{2}[\nabla \vec{\delta} + (\nabla \vec{\delta})^T] = \begin{bmatrix} \epsilon^{11} & \epsilon^{12} & \epsilon^{13} \\ \epsilon^{21} & \epsilon^{22} & \epsilon^{23} \\ \epsilon^{31} & \epsilon^{32} & \epsilon^{33} \end{bmatrix}. \quad (22.1)$$

This symmetric matrix has six distinct elements – three extensional strains and three shear strains – given by

$$\begin{aligned} \epsilon^{11} &= \delta_x^1, & \epsilon^{22} &= \delta_y^2, & \epsilon^{33} &= \delta_z^3 \\ \epsilon^{12} &= \frac{1}{2}(\delta_y^1 + \delta_x^2), & \epsilon^{13} &= \frac{1}{2}(\delta_z^1 + \delta_x^3), & \epsilon^{23} &= \frac{1}{2}(\delta_z^2 + \delta_y^3). \end{aligned} \quad (22.2)$$

The stress tensor σ is also a 3×3 symmetric matrix with six distinct elements,

$$\sigma = \begin{bmatrix} \sigma^{11} & \sigma^{12} & \sigma^{13} \\ \sigma^{21} & \sigma^{22} & \sigma^{23} \\ \sigma^{31} & \sigma^{32} & \sigma^{33} \end{bmatrix}, \quad (22.3)$$

with all elements varying as functions of space and time. This is a tensorial quantity that is written in matrix form corresponding to x - y coordinates. At any point in space this stress tensor represents the internal forces acting at that point. If we introduce a surface through the point with unit normal vector \vec{n} , then the traction (force per unit area) acting on this surface is given by the vector $\sigma \cdot \vec{n}$. In particular, the three columns of σ represent the traction acting on planes normal to the x -, y -, and z -axes respectively. Relative to these planes, the components σ^{11} , σ^{22} , and σ^{33} are the *normal stress* components, while σ^{12} , σ^{13} , and σ^{23} are the *shear stress* components. But it is important to keep in mind that for a plane not aligned with the coordinates, the normal and shear stresses relative to that plane will each in general have values that depend on all components of σ .

We can derive a system of conservation laws governing wave motion as a generalization to the systems (2.91) and (2.98) in the one-dimensional case. These equations have the form

$$\begin{aligned}
 \epsilon_t^{11} - u_x &= 0, \\
 \epsilon_t^{22} - v_y &= 0, \\
 \epsilon_t^{33} - w_z &= 0, \\
 \epsilon_t^{12} - \frac{1}{2}(v_x + u_y) &= 0, \\
 \epsilon_t^{23} - \frac{1}{2}(v_z + w_y) &= 0, \\
 \epsilon_t^{13} - \frac{1}{2}(u_z + w_x) &= 0, \\
 \rho u_t - \sigma_x^{11} - \sigma_y^{12} - \sigma_z^{13} &= 0, \\
 \rho v_t - \sigma_x^{12} - \sigma_y^{22} - \sigma_z^{23} &= 0, \\
 \rho w_t - \sigma_x^{13} - \sigma_y^{23} - \sigma_z^{33} &= 0.
 \end{aligned} \tag{22.4}$$

The first six equations follow directly from the definition of ϵ in terms of the spatial gradient of $\vec{\delta}$, whereas the velocity (u, v, w) is the time derivative of $\vec{\delta}$. (The first of these was derived in (2.92).) The final three equations in (22.4) express the dynamic relationship between the acceleration and the net force resulting from all the stresses.

The equations (22.4) must be completed by specifying a constitutive stress-strain relationship between σ and ϵ . In general this might be nonlinear, but for small deformations a linear stress-strain relation can be assumed, leading to the multidimensional equations of linear elasticity derived in the next subsection.

22.1.1 Linear Elasticity

For small deformations the stress and strain can be related by a generalization of Hooke's law, which has the general form

$$\sigma^{ij} = \sum_{k,l} C^{ijkl} \epsilon^{kl}. \tag{22.5}$$

The tensor C has 81 components, but by symmetry only 21 are independent. We will make a considerable further simplification by assuming that the material is *isotropic*, and hence the material behavior is the same in any direction. In this case the six independent components of σ can be related to those of ϵ by means of a 6×6 matrix, which will be displayed below.

If we apply a small force σ^{11} in the x -direction to an elastic bar, we expect the material to stretch by a linearly proportional small amount,

$$\epsilon^{11} = \frac{1}{E} \sigma^{11}, \tag{22.6}$$

as in Hooke's law. The parameter E is called *Young's modulus*. In general we also expect the bar to contract slightly in the y - and z -directions as it is stretched in x . Since the material is isotropic, we expect the strains ϵ^{22} and ϵ^{33} to be equal to one another and, for small deformations, linear in ϵ^{11} :

$$\epsilon^{22} = \epsilon^{33} = -\nu\epsilon^{11}.$$

The parameter ν is *Poisson's ratio*. For most materials $0 < \nu < 0.5$, although there are strange materials for which $\nu < 0$ (e.g., [252]). Thermodynamics requires $-1 \leq \nu \leq 0.5$. If $\nu = 0.5$, then the material is *incompressible*, a mathematical idealization in that in reality any material can be compressed if sufficient force is applied. The assumption $\nu < 0.5$ is required for a hyperbolic formulation.

Using (22.6), we can write

$$\epsilon^{22} = \epsilon^{33} = -\frac{\nu}{E}\sigma^{11}.$$

Similarly, a force applied in the y - or z -direction will also typically cause strains in all three directions. More generally we can think of applying normal stresses σ^{11} , σ^{22} , and σ^{33} simultaneously, resulting in strains that are a linear combination of those obtained from each stress separately. This leads to the extensional strains

$$\begin{aligned}\epsilon^{11} &= \frac{1}{E}\sigma^{11} - \frac{\nu}{E}\sigma^{22} - \frac{\nu}{E}\sigma^{33}, \\ \epsilon^{22} &= \frac{1}{E}\sigma^{22} - \frac{\nu}{E}\sigma^{11} - \frac{\nu}{E}\sigma^{33}, \\ \epsilon^{33} &= \frac{1}{E}\sigma^{33} - \frac{\nu}{E}\sigma^{11} - \frac{\nu}{E}\sigma^{22}.\end{aligned}\tag{22.7}$$

The shear strains and shear stresses are related to one another by the simpler relations

$$\sigma^{12} = 2\mu\epsilon^{12}, \quad \sigma^{13} = 2\mu\epsilon^{13}, \quad \sigma^{23} = 2\mu\epsilon^{23}.\tag{22.8}$$

where $\mu \geq 0$ is the *shear modulus*. For elastic materials the shear modulus can be determined in terms of E and ν as

$$\mu = \frac{E}{2(1 + \nu)}.\tag{22.9}$$

See [102], for example, for a derivation.

Combining (22.7) and (22.8) gives the desired stress–strain relation

$$\begin{bmatrix} \epsilon^{11} \\ \epsilon^{22} \\ \epsilon^{33} \\ \epsilon^{12} \\ \epsilon^{23} \\ \epsilon^{13} \end{bmatrix} = \begin{bmatrix} 1/E & -\nu/E & -\nu/E & 0 & 0 & 0 \\ -\nu/E & 1/E & -\nu/E & 0 & 0 & 0 \\ -\nu/E & -\nu/E & 1/E & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/2\mu & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/2\mu & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/2\mu \end{bmatrix} \begin{bmatrix} \sigma^{11} \\ \sigma^{22} \\ \sigma^{33} \\ \sigma^{12} \\ \sigma^{23} \\ \sigma^{13} \end{bmatrix}. \quad (22.10)$$

Copyright © 2002, Cambridge University Press.
All rights reserved. May not be reproduced in any form without permission from the publisher, except fair uses permitted under U.S. or applicable copyright law.

We can invert this matrix to instead determine the stress in terms of the strain,

$$\begin{bmatrix} \sigma^{11} \\ \sigma^{22} \\ \sigma^{33} \\ \sigma^{12} \\ \sigma^{23} \\ \sigma^{13} \end{bmatrix} = \begin{bmatrix} \lambda + 2\mu & \lambda & \lambda & 0 & 0 & 0 \\ \lambda & \lambda + 2\mu & \lambda & 0 & 0 & 0 \\ \lambda & \lambda & \lambda + 2\mu & 0 & 0 & 0 \\ 0 & 0 & 0 & 2\mu & 0 & 0 \\ 0 & 0 & 0 & 0 & 2\mu & 0 \\ 0 & 0 & 0 & 0 & 0 & 2\mu \end{bmatrix} \begin{bmatrix} \epsilon^{11} \\ \epsilon^{22} \\ \epsilon^{33} \\ \epsilon^{12} \\ \epsilon^{23} \\ \epsilon^{13} \end{bmatrix}. \quad (22.11)$$

Here we have introduced the parameter λ defined by

$$\lambda = \frac{\nu E}{(1+\nu)(1-2\nu)}. \quad (22.12)$$

This does not have any direct physical interpretation, but is useful in that it appears in the inverse above. The relation (22.9) has also been used to simplify the form of this inverse. The parameter λ should not be confused with an eigenvalue, for which we use the symbol s in this chapter. The parameters λ and μ are often called the *Lamé parameters* for the material. From (22.9) and (22.12) we can also compute E and ν from λ and μ , as

$$E = \frac{\mu(3\lambda + 2\mu)}{\lambda + \mu}, \quad \nu = \frac{1}{2} \left(\frac{\lambda}{\lambda + \mu} \right). \quad (22.13)$$

The relationship (22.11) can be used to convert (22.4) into a closed system of nine equations for the velocities $\vec{u} = (u, v, w)$ and either σ or ϵ , by eliminating the other set of six parameters (analogously to choosing (2.93) or (2.95) in the one-dimensional case). Either way we obtain a hyperbolic linear system of nine equations. The two alternative systems are similarity transformations of one another and have the same eigenvalues, as they must, since they model the same elastic waves.

We will use \vec{u} and σ , as is more common in linear elasticity. Then we need expressions for the time derivatives of the stresses. These may be obtained by using (22.11) to write, for example,

$$\sigma_t^{11} = (\lambda + 2\mu)\epsilon_t^{11} + \lambda\epsilon_t^{22} + \lambda\epsilon_t^{33}$$

and then using the equations of motion (22.4) to evaluate the time derivatives on the right-hand side. We obtain the system

$$\begin{aligned}
 \sigma_t^{11} - (\lambda + 2\mu)u_x - \lambda v_y - \lambda w_z &= 0, \\
 \sigma_t^{22} - \lambda u_x - (\lambda + 2\mu)v_y - \lambda w_z &= 0, \\
 \sigma_t^{33} - \lambda u_x - \lambda v_y - (\lambda + 2\mu)w_z &= 0, \\
 \sigma_t^{12} - \mu(v_x + u_y) &= 0, \\
 \sigma_t^{23} - \mu(v_z + w_y) &= 0, \\
 \sigma_t^{13} - \mu(u_z + w_x) &= 0, \\
 \rho u_t - \sigma_x^{11} - \sigma_y^{12} - \sigma_z^{13} &= 0, \\
 \rho v_t - \sigma_x^{12} - \sigma_y^{22} - \sigma_z^{23} &= 0, \\
 \rho w_t - \sigma_x^{13} - \sigma_y^{23} - \sigma_z^{33} &= 0.
 \end{aligned} \tag{22.14}$$

This can be written as

$$q_t + Aq_x + Bq_y + Cq_z = 0, \tag{22.15}$$

with

$$q = \begin{bmatrix} \sigma^{11} \\ \sigma^{22} \\ \sigma^{33} \\ \sigma^{12} \\ \sigma^{23} \\ \sigma^{13} \\ u \\ v \\ w \end{bmatrix}, \quad A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & -(\lambda + 2\mu) & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\lambda & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\lambda & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mu & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mu \\ -1/\rho & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1/\rho & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1/\rho & 0 & 0 & 0 \end{bmatrix}, \tag{22.16}$$

and similar matrices B and C with the nonzero elements shifted to different locations.

The matrices A , B , and C do not commute, and so these equations are generally coupled in the multidimensional case. This is not surprising, since we expect that elastic waves, like acoustic waves, can propagate equally well in any direction. In fact the eigenvalues of $\tilde{A} = n^x A + n^y B + n^z C$ are the same for any unit vector \vec{n} , and are given by

$$\begin{aligned}
 s^1 &= -c_p, & s^2 &= c_p, & s^3 &= -c_s, & s^4 &= c_s, \\
 s^5 &= -c_s, & s^6 &= c_s, & s^7 &= s^8 &= s^9 &= 0.
 \end{aligned} \tag{22.17}$$

These are not ordered monotonically, but instead the three eigenvalues with modulus 0 are grouped last, since in practice we only need to propagate six waves after solving any

one-dimensional Riemann problem. The 1- and 2-waves are the P-waves, propagating in the directions $\pm\vec{n}$. There are also two sets of S-waves corresponding to the fact that shear motions are in the two-dimensional plane orthogonal to this direction. The wave speeds are given by

$$c_p = \sqrt{\frac{\lambda + 2\mu}{\rho}}, \quad c_s = \sqrt{\frac{\mu}{\rho}}. \quad (22.18)$$

Copyright © 2002, Cambridge University Press.
All rights reserved. May not be reproduced in any form without permission from the publisher, except fair uses permitted under U.S. or applicable copyright law.

The Riemann problem in each coordinate direction is easy to solve. In the x -direction, for example, we have the following eigenvectors of the matrix A of (22.16):

$$r^{1,2} = \begin{bmatrix} \lambda + 2\mu \\ \lambda \\ \lambda \\ 0 \\ 0 \\ 0 \\ \pm c_p \\ 0 \\ 0 \end{bmatrix}, \quad r^{3,4} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \mu \\ 0 \\ 0 \\ 0 \\ \pm c_s \\ 0 \end{bmatrix}, \quad r^{5,6} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \mu \\ 0 \\ 0 \\ \pm c_s \end{bmatrix}. \quad (22.19)$$

These correspond to P-waves in x , shear waves with displacement in the y -direction, and shear waves with displacement in the z -direction, respectively. The other three eigenvectors, corresponding to $\lambda^{7,8,9} = 0$, are given by

$$r^7 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad r^8 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad r^9 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \quad (22.20)$$

These correspond to jumps in σ^{23} , σ^{22} , or σ^{33} alone, each of which causes no wave propagation in x . The matrices B and C have similar eigenvectors, again with the nonzero elements appropriately rearranged.

Note that if we solve a plane-wave problem in which there is only variation in x , then $q_y = q_z = 0$ and the three-dimensional system reduces to $q_t + Aq_x = 0$. In this case the system does decouple into systems of the form discussed in Section 2.12.4. Actually the P-waves given by $r^{1,2}$ in (22.19) carry variation in σ^{22} and σ^{33} as well as in σ^{11} and u . However, this stress in the y - and z -directions exactly balances the stress in x in such a way that the strain is entirely in the x -direction, i.e., $\epsilon^{22} = \epsilon^{33} = 0$, as is clear when the stress components of $r^{1,2}$ are inserted into (22.10). In spite of the fact that the Poisson ratio is typically nonzero, a compressional plane wave in an infinite solid causes no deformation

in the orthogonal directions (as indicated in Figure 2.2). Also, since the second and third columns of A in (22.16) are identically zero, we see that the stresses σ^{22} and σ^{33} cause no dynamic effects and we can drop these variables from the system in deriving the one-dimensional equations (2.93). But note that these one-dimensional equations are based on the assumption of a plane wave in an infinite three-dimensional solid, as discussed further in Section 22.3. Other “one-dimensional” situations lead to different equations. For example, the equations modeling longitudinal waves in thin elastic rod are discussed in Section 22.6.

22.1.2 The Bulk Modulus and Acoustics

The *mean stress* in a solid is defined to be one third the trace of the stress tensor,

$$\frac{1}{3} \operatorname{tr}(\sigma) = \frac{1}{3}(\sigma^{11} + \sigma^{22} + \sigma^{33}). \quad (22.21)$$

This is an invariant of the stress tensor, i.e., it has the same value regardless of the choice of coordinate system used. The trace of the strain tensor is also an invariant, and this value

$$e \equiv \operatorname{tr}(\epsilon) = \epsilon^{11} + \epsilon^{22} + \epsilon^{33}, \quad (22.22)$$

is called the *volumetric strain*. It approximates the relative change in volume in the strained solid. By adding together the three equations of (22.7), we find that

$$e = \frac{1 - 2\nu}{E} \operatorname{tr}(\sigma), \quad (22.23)$$

and hence the mean stress is related to the volumetric strain by

$$\frac{1}{3} \operatorname{tr}(\sigma) = Ke, \quad (22.24)$$

where the *bulk modulus of compressibility* K is defined by

$$K = \frac{E}{3(1 - 2\nu)} = \lambda + \frac{2}{3}\mu. \quad (22.25)$$

Averaging the first three equations of (22.14) gives an evolution equation for the mean stress,

$$\frac{1}{3}(\sigma^{11} + \sigma^{22} + \sigma^{33})_t - K(u_x + v_y + w_z) = 0. \quad (22.26)$$

We can relate the elastodynamics equations to the acoustics equations derived earlier for gas dynamics if we make the additional assumption on that the stress is *hydrostatic*, as it is in a fluid. This means that there is no shear stress, $\sigma^{12} = \sigma^{13} = \sigma^{23} = 0$, and the extensional stress components are all equal and negative,

$$\sigma^{11} = \sigma^{22} = \sigma^{33} \equiv -p. \quad (22.27)$$

The value p is called the *hydrostatic pressure*, and has the opposite sign from the stresses as discussed in Section 2.12.4. In this case the stress tensor (22.3) reduces to $-pI$ where I is the

identity matrix. Rather than working with this tensor we can reduce the equations and deal only with the scalar pressure p , which satisfies $p = -Ke$ by (22.24), since $p = -\frac{1}{3} \operatorname{tr}(\sigma)$. The equation (22.26) becomes an evolution equation for the hydrostatic pressure,

$$p_t + K(u_x + v_y + w_z) = 0. \quad (22.28)$$

Since we now also assume that $\sigma^{12} = \sigma^{13} = \sigma^{23} = 0$, we can drop the middle three equations of (22.14) and the final three become

$$\begin{aligned} \rho u_t + p_x &= 0, \\ \rho v_t + p_y &= 0, \\ \rho w_t + p_z &= 0. \end{aligned} \quad (22.29)$$

We recognize (22.28), (22.29) as defining the three-dimensional acoustics equations from Section 18.6. This system has wave speeds given by the “speed of sound”

$$c = \sqrt{\frac{K}{\rho}} = \sqrt{\frac{\lambda + \frac{2}{3}\mu}{\rho}}. \quad (22.30)$$

Note that this is different than the P-wave speed c_p of (22.18), which is the sound speed actually observed in solids. However, since a fluid does not support shear stresses we should set $\mu = 0$, in which case (22.18) and (22.30) do agree. The acoustics equations are sometimes used as an approximate system of equations for modeling P-waves in solids when shear waves are relatively unimportant, particularly in solids where μ is small compared to λ .

22.2 The Plane-Strain Equations of Two-Dimensional Elasticity

We can reduce the three-dimensional equations (22.14) to two space dimensions by setting $q_z \equiv 0$, for example, if we assume there is no variation in the z -direction. Note that the strain $\epsilon^{33} = \delta_z^3$ must be zero in this case, since it is the z -derivative of the z -displacement. (We discuss below when this assumption is reasonable.) The stress σ^{33} will not generally be zero, but can be determined in terms of σ^{11} and σ^{22} as discussed below. Dropping the equation for σ^{33} from (22.14) along with all z -derivative terms, the remaining eight equations reduce to two decoupled systems of equations,

$$\begin{aligned} \sigma_t^{11} - (\lambda + 2\mu)u_x - \lambda v_y &= 0, \\ \sigma_t^{22} - \lambda u_x - (\lambda + 2\mu)v_y &= 0, \\ \sigma_t^{12} - \mu(v_x + u_y) &= 0, \\ \rho u_t - \sigma_x^{11} - \sigma_y^{12} &= 0, \\ \rho v_t - \sigma_x^{12} - \sigma_y^{22} &= 0, \end{aligned} \quad (22.31)$$

and

$$\begin{aligned}\sigma_t^{23} - \mu w_y &= 0, \\ \sigma_t^{13} - \mu w_x &= 0, \\ \rho w_t - \sigma_x^{13} - \sigma_y^{23} &= 0.\end{aligned}\quad (22.32)$$

The latter system (22.32) models shear waves with motion orthogonal to the $x-y$ plane. Waves modeled by this system have speed c_s , the S-wave speed given in (22.18).

The system (22.31) is the more interesting system, and models both P-waves and S-waves for which the motion is in the $x-y$ plane. The S-waves modeled by this system have material motion orthogonal to the direction the wave is propagating, but still within the $x-y$ plane. This system (22.31) is often called the two-dimensional *plane-strain equations*, since the strain is confined entirely to the $x-y$ plane. This is a reasonable model for plane waves propagating through a three-dimensional elastic body in cases where there is no variation in the z -direction, for example, if the $x-y$ plane is a representative slice through a three-dimensional solid with essentially infinite extent in the z -direction, as might occur in modeling large-scale seismic waves in the earth, for example. If it is correct to assume that there is no variation in the z -direction, then it is also valid to assume that $\epsilon^{33} = 0$. Otherwise, if ϵ^{33} had some nonzero value independent of z , then the displacement δ^3 would have to be of the form $\delta^3 = \epsilon^{33}(z - z_0)$ and grow without bound in z . This is not reasonable for finite-amplitude waves. Of course, as the material is compressed in the x - or y -direction it will try to expand in z (when $\nu \neq 0$), but it will be prevented from doing so by the adjacent material, which is trying equally hard to expand in the other direction. The result is a nonzero stress σ^{33} while ϵ^{33} remains zero. Indeed, setting $\epsilon^{33} = 0$ in the system (22.11) yields

$$\sigma^{11} = (\lambda + 2\mu)\epsilon^{11} + \lambda\epsilon^{22}, \quad (22.33)$$

$$\sigma^{22} = \lambda\epsilon^{11} + (\lambda + 2\mu)\epsilon^{22}, \quad (22.34)$$

$$\sigma^{33} = \lambda\epsilon^{11} + \lambda\epsilon^{22}. \quad (22.35)$$

The first two equations of this set are all that are needed for the two-dimensional system (22.31), but the stress σ^{33} can also be computed from (22.35) if desired. Alternatively we can obtain

$$\sigma^{33} = \nu(\sigma^{11} + \sigma^{22}) \quad (22.36)$$

from the third equation of (22.7) by setting $\epsilon^{33} = 0$.

Note that if we invert the stress-strain relation (22.33)–(22.35) to find ϵ^{11} and ϵ^{22} in terms of σ^{11} and σ^{22} , we find that

$$\begin{aligned}\hat{E}\epsilon^{11} &= \sigma^{11} - \hat{\nu}\sigma^{22}, \\ \hat{E}\epsilon^{22} &= \sigma^{22} - \hat{\nu}\sigma^{11},\end{aligned}\quad (22.37)$$

where

$$\hat{E} = \frac{E}{1 - \nu^2}, \quad \hat{\nu} = \frac{\nu}{1 - \nu}. \quad (22.38)$$

The equations (22.37) have the same form as the three-dimensional stress-strain relations (22.7), but with different effective values for the Young's modulus \hat{E} and Poisson ratio $\hat{\nu}$. These relations can be derived either by inverting the 2×2 system given by (22.33) and (22.34), or from (22.7) by using (22.36).

It is important to note that the plane-strain system (22.31) does *not* in general model elastic waves in a thin plate, in spite of the fact that it might seem natural to view this as a two-dimensional elastic medium. Wave propagation in a plate can be modeled by a two-dimensional hyperbolic system, but (22.31) is not the correct one; see Section 22.5.

We now discuss the eigenstructure of the system (22.31). Rather than displaying the matrices A and B separately in this case, it is more compact and perhaps also more revealing to show the linear combination $\check{A} = n^x A + n^y B$, where \vec{n} is again a unit vector in an arbitrary direction. The matrix \check{A} is then the coefficient matrix for the one-dimensional problem modeling the propagation of plane waves in the \vec{n} -direction. Setting $\vec{n} = (1, 0)$ or $(0, 1)$ below recovers the matrices A and B separately. We have

$$q = \begin{bmatrix} \sigma^{11} \\ \sigma^{22} \\ \sigma^{12} \\ u \\ v \end{bmatrix}, \quad \check{A} = -\begin{bmatrix} 0 & 0 & 0 & n^x(\lambda + 2\mu) & n^y\lambda \\ 0 & 0 & 0 & n^x\lambda & n^y(\lambda + 2\mu) \\ 0 & 0 & 0 & n^y\mu & n^x\mu \\ n^x/\rho & 0 & n^y/\rho & 0 & 0 \\ 0 & n^y/\rho & n^x/\rho & 0 & 0 \end{bmatrix}. \quad (22.39)$$

The eigenvalues of \check{A} are

$$\check{s}^1 = -c_p, \quad \check{s}^2 = c_p, \quad \check{s}^3 = -c_s, \quad \check{s}^4 = c_s, \quad \check{s}^5 = 0, \quad (22.40)$$

where we use s instead of λ to avoid confusion with the Lamé parameter. The P-wave eigenvectors are

$$\check{r}^1 = \begin{bmatrix} \lambda + 2\mu(n^x)^2 \\ \lambda + 2\mu(n^y)^2 \\ 2\mu n^x n^y \\ n^x c_p \\ n^y c_p \end{bmatrix}, \quad \check{r}^2 = \begin{bmatrix} \lambda + 2\mu(n^x)^2 \\ \lambda + 2\mu(n^y)^2 \\ 2\mu n^x n^y \\ -n^x c_p \\ -n^y c_p \end{bmatrix}, \quad (22.41)$$

while the S-wave eigenvectors $r^{3,4}$ and the stationary wave r^5 are

$$\check{r}^3 = \begin{bmatrix} -2n^x n^y \mu \\ 2n^x n^y \mu \\ [(n^x)^2 - (n^y)^2] \mu \\ -n^y c_s \\ n^x c_s \end{bmatrix}, \quad \check{r}^4 = \begin{bmatrix} -2n^x n^y \mu \\ 2n^x n^y \mu \\ [(n^x)^2 - (n^y)^2] \mu \\ n^y c_s \\ -n^x c_s \end{bmatrix}, \quad \check{r}^5 = \begin{bmatrix} (n^y)^2 \\ (n^x)^2 \\ -n^x n^y \\ 0 \\ 0 \end{bmatrix}. \quad (22.42)$$

Observe that P-waves $\check{r}^{1,2}$ have velocity components directed in the $\pm \vec{n}$ -direction, the direction in which the plane wave propagates. The S-waves, on the other hand, have motion

in the orthogonal direction $\pm(-n^y, n^x)$. Note also that a P-wave in the x - or y -direction has $\sigma^{12} = 0$, but that a P-wave propagating in any other direction has $\sigma^{12} \neq 0$. This is because the elements of the stress tensor have been expressed in x - y coordinates, and representing a purely extensional stress in some other direction requires all components of σ to be nonzero.

22.3 One-Dimensional Slices

We can reduce the systems of equations (22.31) and (22.32) even further if we assume that there is no variation in the y -direction. As in our discussion of the plane-strain equations above, this is typically valid if we are considering a one-dimensional slice through an essentially infinite medium in cases where there is variation in only one direction (e.g., a plane wave propagating through the earth). It is not a valid model for waves in a “one-dimensional” thin elastic rod, which is discussed in Section 22.6.

Setting all y -derivatives to zero in (22.31) results in the two decoupled systems

$$\begin{aligned}\sigma_t^{11} - (\lambda + 2\mu)u_x &= 0, \\ \rho u_t - \sigma_x^{11} &= 0\end{aligned}\tag{22.43}$$

and

$$\begin{aligned}\sigma_t^{12} - \mu v_x &= 0, \\ \rho v_t - \sigma_x^{12} &= 0.\end{aligned}\tag{22.44}$$

These are the systems (2.95) and (2.100) introduced in Section 2.12.4. They model P-waves with displacement in x and S-waves with displacement in y , respectively, with wave speeds c_p and c_s given by (22.18).

We have dropped the equation for σ^{22} , which is given by

$$\sigma^{22} = \lambda \epsilon^{11} = \hat{\nu} \sigma^{11}.\tag{22.45}$$

Note that in this case $\sigma^{33} = \sigma^{22}$ by (22.36).

The equations (22.32) reduce to

$$\begin{aligned}\sigma_t^{13} - \mu w_x &= 0, \\ \rho w_t - \sigma_x^{13} &= 0.\end{aligned}\tag{22.46}$$

This system models an independent set of shear waves in which the displacement is in the z -direction rather than in the y -direction (and propagation still in the x -direction).

22.4 Boundary Conditions

Boundary conditions for elastic solids can be imposed in much the same way as for acoustics (see Sections 7.3 and 21.8), but there are now a wider variety of physically meaningful boundary conditions to consider. We will use the two-dimensional plane-strain equations (22.31) for illustration, and consider a point along the left edge of the domain, which we

assume is at $x = 0$. We must then determine ghost-cell values Q_{0j} and $Q_{-1,j}$ based on interior values and the physical boundary conditions. It should be clear how to translate this discussion to other boundaries and to three dimensions.

Periodic or extrapolation boundary conditions are easily imposed, as for other equations, as discussed in Section 21.8. The more interesting cases are where either the motion of the boundary or the traction applied to the boundary is specified. These are discussed in the next two subsections.

22.4.1 Specified Motion

Suppose the velocity of the boundary at $x = 0, y = y_j$ is known. Call the velocity at this point (U, V) for brevity. Then following the discussion of Sections 7.3.4 and 21.8.3, we can impose this by specifying the ghost-cell values as follows:

$$\begin{aligned} \text{for } Q_{0j}: \quad \sigma_{0j}^{11} &= \sigma_{1j}^{11}, & \sigma_{0j}^{12} &= \sigma_{1j}^{12}, & \sigma_{0j}^{22} &= \sigma_{1j}^{22}, \\ u_{0j} &= 2U - u_{1j}, & v_{0j} &= 2V - v_{1j}; \\ \text{for } Q_{-1,j}: \quad \sigma_{-1,j}^{11} &= \sigma_{2j}^{11}, & \sigma_{-1,j}^{12} &= \sigma_{2j}^{12}, & \sigma_{-1,j}^{22} &= \sigma_{2j}^{22}, \\ u_{-1,j} &= 2U - u_{2j}, & v_{-1,j} &= 2V - v_{2j}. \end{aligned} \quad (22.47)$$

When the Riemann problem is solved at $x = 0$ (i.e., at cell interface $i = 1/2$), this choice insures that the intermediate state $Q_{1/2}^\downarrow$ has velocity (U, V) and satisfies the required physical boundary condition. An important special case is $U = 0, V = 0$, in which case the boundary is fixed at this point.

Note that in the case of an elastic solid we must specify both u and v . This differs from two-dimensional acoustics or inviscid fluid dynamics, where only the normal component of velocity is specified as discussed in Sections 21.8.3 and 21.8.4. For an inviscid fluid there can be slip along the boundary, and so the tangential component of velocity cannot be specified. For an elastic solid we must specify both. For a three-dimensional problem we would also have to specify w at the boundary, e.g., $w = 0$ at a fixed boundary. The stresses are not specified and must be free to react as necessary to the imposed motion. This is accomplished by simply reflecting σ from the interior values in (22.47), and the values observed in the resulting Riemann solution $Q_{1/2}^\downarrow$ can be used to obtain the surface traction if this is desired as part of the solution to the problem.

22.4.2 Specified Traction

Often we wish instead to specify the traction at a point on the boundary and compute the resulting motion. At the boundary $x = 0$ this amounts to specifying the values of σ^{11} and σ^{12} , say as $\sigma^{11} = S^{11}$ and $\sigma^{12} = S^{12}$ at the point $x = 0, y = y_j$. In particular, if this is a free boundary (the edge of an elastic solid with no external force applied), then we should set $S^{11} = 0$ and $S^{12} = 0$. This is called a *traction-free boundary*. Note that we have no physical control over σ^{22} at this boundary.

The proper ghost cell values for general S^{11} and S^{12} are then given by:

$$\begin{aligned} \text{for } Q_{0j}: \quad \sigma_{0j}^{11} &= 2S^{11} - \sigma_{1j}^{11}, & \sigma_{0j}^{12} &= 2S^{12} - \sigma_{1j}^{12}, & \sigma_{0j}^{22} &= \sigma_{1j}^{22}, \\ u_{0j} &= u_{1j}, & v_{0j} &= v_{1j}; \\ \text{for } Q_{-1,j}: \quad \sigma_{-1,j}^{11} &= 2S^{11} - \sigma_{2j}^{11}, & \sigma_{-1,j}^{12} &= 2S^{12} - \sigma_{2j}^{12}, & \sigma_{-1,j}^{22} &= \sigma_{2j}^{22}, \\ u_{-1,j} &= u_{2j}, & v_{-1,j} &= v_{2j}. \end{aligned} \quad (22.48)$$

except fair uses permitted under U.S. or applicable copyright law.

For a three-dimensional problem we would also need to specify $\sigma^{13} = S^{13}$, while σ^{23} and σ^{33} would simply be reflected from the interior, as is done for σ^{22} in (22.48).

22.5 The Plane-Stress Equations and Two-Dimensional Plates

In Section 22.2 we derived a two-dimensional hyperbolic system by using the plane-strain assumption that all displacements are confined to the x - y plane. As discussed there, this is valid if we assume the material is essentially infinite in the z -direction and there is no variation of the solution in that direction. We now consider a different situation, in which the three-dimensional domain is a thin plate bounded by the planes $z = \pm h$ for some small h . We wish to derive two-dimensional equations that model waves whose wavelength is long relative to the thickness of the plate. Each surface of the plate is a free boundary and must be traction-free (see Section 22.4.2), so we have

$$\sigma^{13} \equiv 0, \quad \sigma^{23} \equiv 0, \quad \sigma^{33} \equiv 0 \quad (22.49)$$

on the boundary planes $z = \pm h$. To derive a two-dimensional system of equations we will assume that these are identically zero throughout the plate. Then from (22.8) we also have

$$\epsilon^{13} \equiv 0, \quad \epsilon^{23} \equiv 0. \quad (22.50)$$

Note, however, that we cannot assume $\epsilon^{33} = 0$. Instead, from (22.7) with $\sigma^{33} = 0$ we obtain

$$\epsilon^{33} = -\frac{\nu}{E}(\sigma^{11} + \sigma^{22}), \quad (22.51)$$

along with

$$\begin{aligned} \epsilon^{11} &= \frac{1}{E}(\sigma^{11} - \nu\sigma^{22}), \\ \epsilon^{22} &= \frac{1}{E}(-\nu\sigma^{11} + \sigma^{22}). \end{aligned} \quad (22.52)$$

Adding these last two equations together gives

$$\epsilon^{11} + \epsilon^{22} = \frac{1-\nu}{E}(\sigma^{11} + \sigma^{22}). \quad (22.53)$$

Since $\epsilon^{11} + \epsilon^{22} = \delta_x^1 + \delta_y^2$ is the x - y divergence of the displacement, we see that if $\epsilon^{11} + \epsilon^{22} \neq 0$, then there is compression or expansion in the x - y plane, and in this case from (22.51) there must be compensating motion in the z -direction whenever the Poisson

ratio ν is nonzero. No matter how thin the plate may appear, it is still three-dimensional, and stretching or compressing it in the $x-y$ plane leads to motion in z of the same order of magnitude.

To derive two-dimensional equations we will assume, however, that σ^{11} and σ^{22} , and hence ϵ^{33} , are independent of z . Recall that $\epsilon^{33} = \delta_z^3$, so $\epsilon^{33} > 0$ corresponds to the plate bulging out, while $\epsilon^{33} < 0$ corresponds to the plate becoming thinner.

As the plate thickens or thins, the z -velocity w must be nonzero. Moreover, it is clearly not valid to assume that the value of w is independent of z . For example, if the plate is bulging out, then we must have $w > 0$ for $0 < z < h$ and $w < 0$ for $-h < z < 0$. Since $w = \delta_t^3$, we have $w_z = \delta_{zt}^3 = \epsilon_t^{33}$, and the assumption that ϵ^{33} is independent of z means that w varies linearly in z . Although w is not zero, it is close to zero and symmetric about $z = 0$. The plate equations are most rigorously defined by integrating the three-dimensional equations in z from $-h$ to h and the integral of w then reduces to zero, to leading order. This justifies ignoring the effects of w in deriving the two-dimensional plate equations.

Using the assumptions (22.49) and (22.50) in the system (22.4) gives the following system of equations (after dropping the equations for ϵ^{13} , ϵ^{23} , ϵ^{33} , and w):

$$\begin{aligned}\epsilon_t^{11} - u_x &= 0, \\ \epsilon_t^{22} - v_y &= 0, \\ \epsilon_t^{12} - \frac{1}{2}(v_x + u_y) &= 0, \\ \rho u_t - \sigma_x^{11} - \sigma_y^{12} &= 0, \\ \rho v_t - \sigma_x^{12} - \sigma_y^{22} &= 0.\end{aligned}\tag{22.54}$$

To close this system we need the constitutive relations relating σ to ϵ , which are given by (22.52) along with $\sigma^{12} = 2\mu\epsilon^{12}$ coming from (22.8). We can write these either as

$$\begin{aligned}\epsilon^{11} &= \frac{1}{E}(\sigma^{11} - \nu\sigma^{22}), \\ \epsilon^{22} &= \frac{1}{E}(-\nu\sigma^{11} + \sigma^{22}), \\ \epsilon^{12} &= \frac{1}{2\mu}\sigma^{12}\end{aligned}\tag{22.55}$$

or, by inverting the system, as

$$\begin{aligned}\sigma^{11} &= \left(\frac{2\mu}{1-\nu}\right)(\epsilon^{11} + \nu\epsilon^{22}), \\ \sigma^{22} &= \left(\frac{2\mu}{1-\nu}\right)(\nu\epsilon^{11} + \epsilon^{22}), \\ \sigma^{12} &= 2\mu\epsilon^{12}.\end{aligned}\tag{22.56}$$

As usual, we can eliminate either ϵ or σ from (22.54). If we eliminate ϵ , then we obtain the

system

$$\begin{aligned}\sigma_t^{11} - \left(\frac{2\mu}{1-\nu}\right)u_x - \left(\frac{2\mu\nu}{1-\nu}\right)v_y &= 0, \\ \sigma_t^{22} - \left(\frac{2\mu\nu}{1-\nu}\right)u_x - \left(\frac{2\mu}{1-\nu}\right)v_y &= 0, \\ \sigma_t^{12} - \mu(v_x + u_y) &= 0, \\ \rho u_t - \sigma_x^{11} - \sigma_y^{12} &= 0, \\ \rho v_t - \sigma_x^{12} - \sigma_y^{22} &= 0.\end{aligned}\tag{22.57}$$

Copyright © 2002, Cambridge University Press.
All rights reserved. May not be reproduced in any form without permission from the publisher, except fair uses permitted under U.S. or applicable copyright law.

Note that this system has different coefficients than the plane-strain system (22.31). However, if we define

$$\hat{\lambda} = \frac{2\mu\nu}{1-\nu} = \frac{2\mu\lambda}{\lambda + 2\mu}\tag{22.58}$$

as an effective Lamé parameter for the plate, then

$$\frac{2\mu}{1-\nu} = \hat{\lambda} + 2\mu,$$

and the system (22.57) has exactly the same form as (22.31) but with $\hat{\lambda}$ in place of λ . Hence any method developed for the plane-strain case can also be applied to the plane-stress equations simply by changing the value of λ . In particular, the eigenstructure is the same as that developed in Section 22.2 (with $\hat{\lambda}$ in place of λ), and so we see from the eigenvalues (22.40) that the characteristic wave speeds for waves in a plate are

$$\hat{c}_p = \sqrt{\frac{\hat{\lambda} + 2\mu}{\rho}} = \sqrt{\frac{2\mu}{\rho(1-\nu)}} = \sqrt{\frac{E}{\rho(1-\nu^2)}}\tag{22.59}$$

and

$$c_s = \sqrt{\mu/\rho},\tag{22.60}$$

respectively. The speed c_s is the usual S-wave speed, but the wave speed \hat{c}_p is smaller than c_p from (22.18) when $0 < \nu < 1/2$, since

$$\hat{\lambda} = \lambda \left(1 - \frac{\nu}{1-\nu}\right) < \lambda.\tag{22.61}$$

We will refer to waves propagating at the velocity \hat{c}_p as \hat{P} -waves.

Example 22.1. We can investigate wave propagation in a thin plate numerically by solving the three-dimensional elasticity equations in a plate with finite thickness $-h < z < h$ and imposing traction-free boundary conditions $\sigma^{13} = \sigma^{23} = \sigma^{33} = 0$ at $z = \pm h$. To illustrate an isolated \hat{P} -wave we can assume the wave is propagating in the x -direction

EBSCO : eBook Collection (EBSCONHOST) printed on 27/28/2018 4:06 PM via FLORIDA INST OF TECHNOLOGY

AN: 112638 ; LeVeque, Randall J.; Finite Volume Methods for Hyperbolic Problems

Account: flit.main.ehost

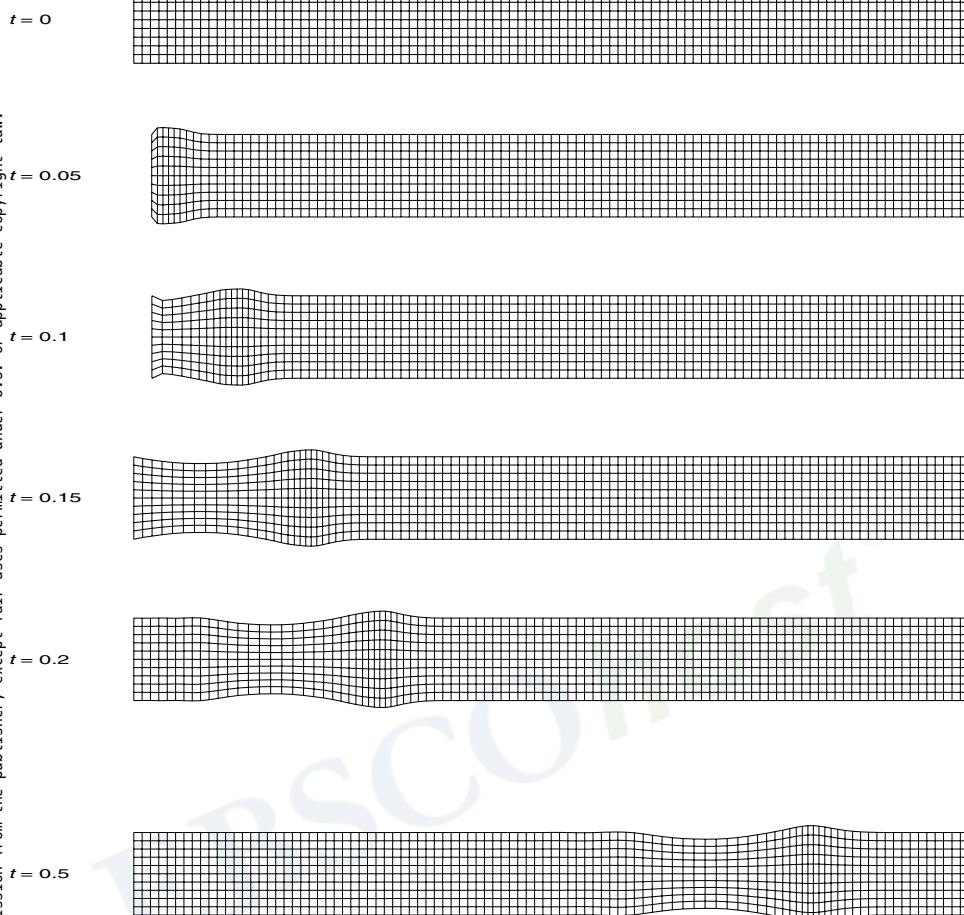


Fig. 22.1. Side view of a thin plate in the x - z plane. The plate is pushed inwards at the left, giving rise to a P-wave as described in the text. [claw/book/chap22/plate]

and there is no variation in y , and hence no strain in the y -direction. In this case we can solve a two-dimensional problem in the x - z plane. The appropriate equations are now the plane-strain equations of Section 22.2, rewritten in x and z instead of x and y . Figures 22.1 and 22.2 show results of a numerical calculation in which the traction-free boundary conditions $\sigma^{13} = \sigma^{33} = 0$ are imposed at $z = \pm 0.01$ (see Section 22.4.2) and the plate is initially undisturbed. Boundary conditions at $x = 0$ are given by specifying the velocity (see Section 22.4.1) as

$$u(0, z, t) = \begin{cases} a \sin(2\pi t/T) & \text{if } t \leq T, \\ 0 & \text{if } t > T, \end{cases} \quad (22.62)$$

$$w(0, z, t) = 0$$

with $T = 0.15$. This corresponds to pushing the plate inward and then bringing the edge back to its original location, slowly enough that the wavelength is long relative to the thickness.

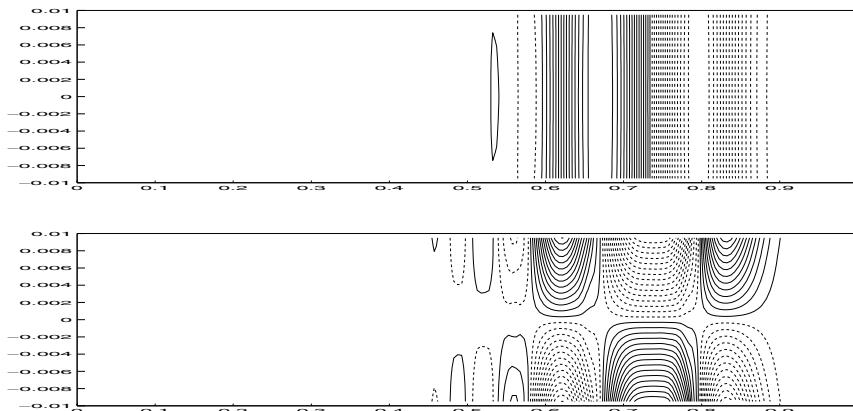


Fig. 22.2. The stress σ^{11} (top) and the vertical velocity w (bottom) at time $t = 0.5$ for the wave propagation problem shown in Figure 22.1. Contour plots are shown with solid lines for positive values and dashed lines for negative values. [claw/book/chap22/plate]

Figure 22.1 shows the deformation of the plate at several times. This deformation has been determined by integrating the velocity (u, w) in each cell over the course of the computation in order to compute the displacement. Points on an initially uniform grid are then displaced by this amount to illustrate the deformed solid. For clarity these deformations are magnified considerably over what is appropriate for linear elasticity (i.e., the constant a in the boundary data (22.62) is taken to be orders of magnitude larger than physically reasonable, but since the equations are linear, the solution simply scales linearly as well). Recall that the computations are always done on a fixed grid in the reference configuration, since the elasticity equations we are considering are written in the Lagrangian frame.

Figure 22.2 shows contours of the computed σ^{11} and w at the final time $t = 0.5$. We see that σ^{11} essentially varies only with x in spite of the fact that the velocity w varies linearly in the z -direction. Moreover we observe that the leading edge of the wave has not yet reached $x = 0.9$ at this time. The material parameters were chosen to be $\rho = 1$, $\lambda = 2$, and $\mu = 1$ so that $c_p = 2$. Rather than traveling at this P-wave speed, the disturbance is propagating at the velocity $\hat{c}_p = \sqrt{3}$, and at time $t = 0.5$ has reached approximately $0.5\hat{c}_p = 0.866$ rather than $0.5c_p = 1$.

If this same computation were repeated but with $u = w = 0$ specified along $z = \pm h$ instead of $\sigma^{13} = \sigma^{33} = 0$ (plane strain rather than a thin plate), the result would look like the figure on the left of Figure 2.2 rather than Figure 22.1, and the disturbance would have reached $x = 1$ at time $t = 0.5$. In this case $w = 0$ would be exactly maintained and there would be no deformation in the z -direction.

It may seem strange that the speed of a P-wave in a thin plate is smaller than c_p . After all, the plate is composed of a material characterized by the Lamé parameters λ and μ , and should have a characteristic propagation speed given by c_p . However, a P-wave is not simply a P-wave propagating along the plate with deformation confined to the x - y plane. Rather, it is a wave that can be viewed as a superposition of many waves propagating transversely through the plate, bouncing back and forth between the top and bottom surfaces and hence moving at a slower effective speed along the plate. This internal reflection at the

free surface is not apparent in Figure 22.2, since the zig-zagging waves effectively combine into standing waves in the z -direction. In a thicker plate the transverse wave motion would be more apparent. (A similar effect was seen in Section 9.14 for waves propagating in a layered heterogeneous material.)

22.6 A One-Dimensional Rod

Consider a thin elastic rod that is long in the x -direction and has small cross-sectional area, say $-h \leq y \leq h$ and $-h \leq z \leq h$. If we consider compressional waves propagating down the rod whose wavelength is long compared to h , then these can be modeled with a one-dimensional system of equations. We can start with the plane-stress equations (22.54), which model a thin plate $-h \leq z \leq h$, and now restrict also to $-h \leq y \leq h$ by imposing traction-free boundary conditions on these surfaces: $\sigma^{12} = \sigma^{22} = 0$. As in the derivation of the plane-stress equations, we now assume that in fact $\sigma^{12} = \sigma^{22} = 0$ throughout the rod, and also that $v = 0$ (in addition to the previous assumption that $w = 0$ and $\sigma^{k3} = 0$ for $k = 1, 2, 3$). Then (22.54) reduces to

$$\begin{aligned}\epsilon_t^{11} - u_x &= 0, \\ \rho u_t - \sigma_x^{11} &= 0.\end{aligned}\tag{22.63}$$

From (22.52) we now have the constitutive relation $\epsilon^{11} = \sigma^{11}/E$, and so the one-dimensional system can be rewritten as

$$\begin{aligned}\sigma_t^{11} - Eu_x &= 0, \\ \rho u_t - \sigma_x^{11} &= 0.\end{aligned}\tag{22.64}$$

This has the same structure as the equations (22.43) derived for a one-dimensional slice of a three-dimensional solid. But the fact that the rod has traction-free boundaries and is free to contract or expand in y and z leads to a different wave speed,

$$c_p^{\text{rod}} = \sqrt{E/\rho},\tag{22.65}$$

as seen by computing the eigenvalues of the coefficient matrix from (22.64).

22.7 Two-Dimensional Elasticity in Heterogeneous Media

The multidimensional elastic wave equations can be solved numerically using essentially the same procedure as for the linear acoustics equations, as was described in Chapter 21. This is also easily implemented for a heterogeneous medium by allowing each grid cell to have distinct values for the density ρ and Lamé parameters λ and μ . For the two-dimensional equations discussed in Section 22.2, the Riemann solvers in [claw/book/chap22/rp] give an implementation of the necessary eigendecompositions. These are based directly on the eigenstructure determined in Section 22.2. Examples will be presented for the plane-strain case described there, but the same solver work also for the plane-stress equations modeling a thin plate if λ is replaced by $\hat{\lambda}$ as described in Section 22.5.

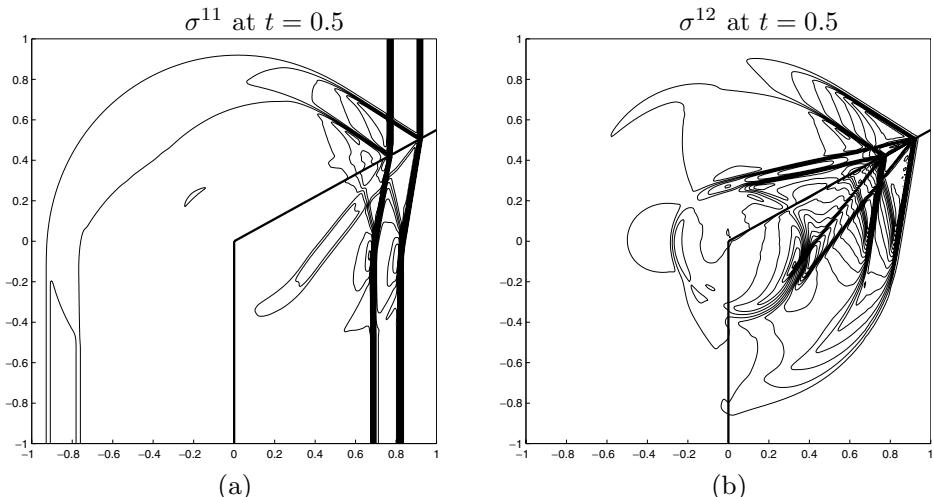


Fig. 22.3. Elastic wave propagation with initial data consisting of a P-wave as shown in Figure 21.1(b):
 (a) the stress σ^{11} ; (b) the shear stress σ^{12} . [claw/book/chap22/corner]

Example 22.2. Figure 22.3 shows an example in the same domain indicated in Figure 21.1(a), but with the two regions now containing different elastic materials with parameters

$$\begin{aligned} \rho_l &= 1, & \rho_r &= 1, \\ \lambda_l &= 4, & \lambda_r &= 2, \\ \mu_l &= 0.5, & \mu_r &= 1, \\ c_{pl} &= \sqrt{5} \approx 2.2, & c_{pr} &= 2, \\ c_{sl} &= \sqrt{0.5} \approx 0.7, & c_{sr} &= 1. \end{aligned} \quad (22.66)$$

The initial data is zero everywhere, except for a perturbation as in Figure 21.1(b), in which

$$\sigma^{11} = \lambda_l + 2\mu_l, \quad \sigma^{22} = \lambda_l, \quad \sigma^{12} = 0, \quad u = c_{pl}, \quad v = 0 \quad (22.67)$$

for $-0.35 < x < -0.2$. This is an eigenvector r^2 from (22.41) (with $\vec{n} = (1, 0)$) and hence is a right-going P-wave. After hitting the interface, the transmitted P-wave moves more slowly and there is a partial reflection, as seen in Figure 22.3(a), where a contour plot of σ^{11} is shown. In the elastic case there is also both a transmitted and reflected S-wave along the ramp portion of the interface. These are faintly visible in Figure 22.3(a), since S-waves at an oblique angle have a nonzero σ^{11} component (see Section 22.2). The S-waves are much more clearly visible in Figure 22.3(b), which shows σ^{12} . Note that the transmitted and reflected P-waves also contain significant components of σ^{12} , since they are moving at an angle to the grid.

Example 22.3. As another example of elastic wave propagation in a heterogeneous medium, we consider a wave propagating into a solid that has embedded within it an *inclusion* made out of a stiffer material, as shown in Figure 22.4. The darker region represents material with

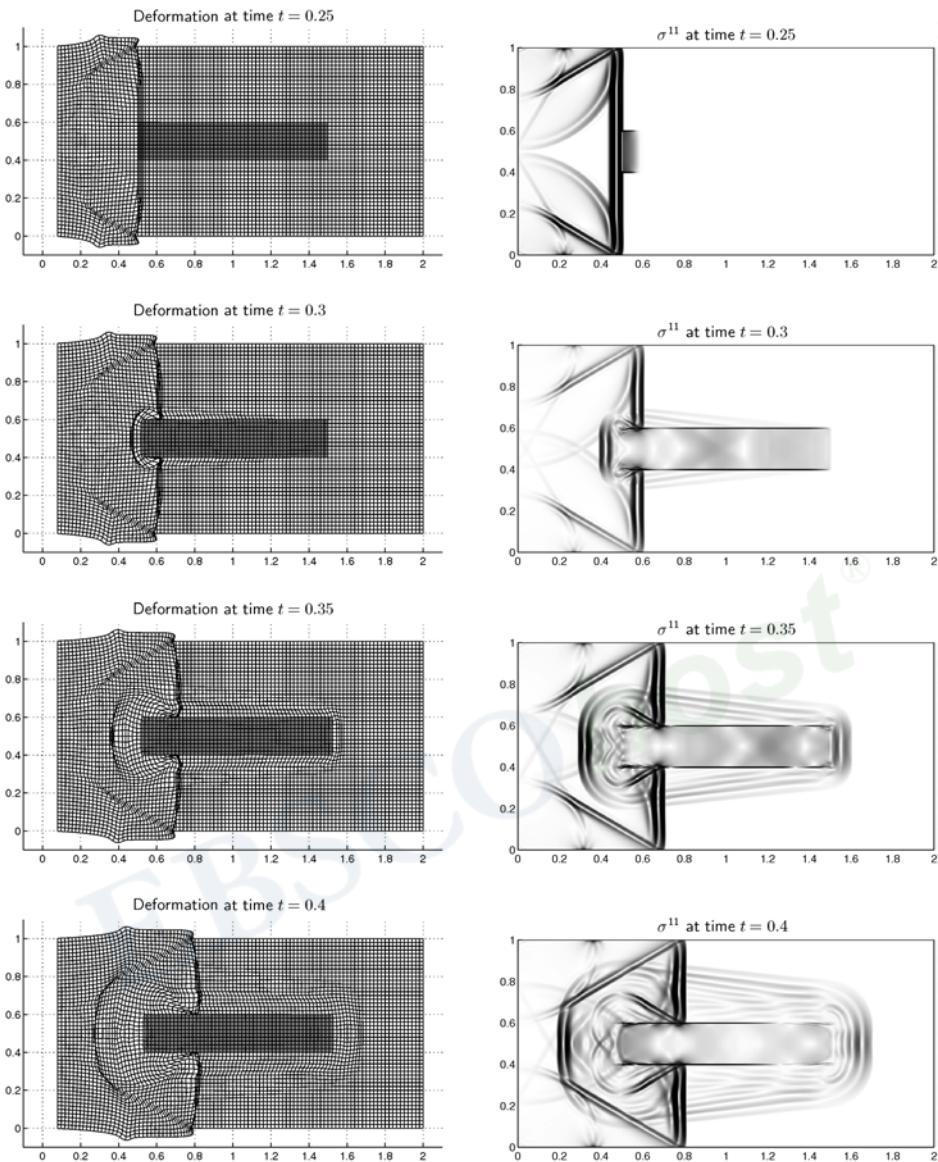


Fig. 22.4. Left column: Deformation of an elastic solid with a stiff inclusion due to compression at the left boundary. The linear deformation is greatly exaggerated. Right column: Schlieren image of σ^{11} . [claw/book/chap22/inclusion]

$\lambda = 200$, $\mu = 100$, while the lighter-colored material has $\lambda = 2$ and $\mu = 1$. The density is the same everywhere, $\rho = 1$. The plane-strain equations are solved with traction-free boundary conditions at $y = 0, 1$ and at $x = 1$. At $x = 0$ the velocity is specified as

$$u(0, y, t) = \begin{cases} \epsilon \sin(\pi t / 0.025) & \text{if } t < 0.025, \\ 0 & \text{if } t \geq 0.025, \end{cases} \quad v(0, y, t) = 0. \quad (22.68)$$

The solid is simply pushed over by a small amount at the left boundary. This creates

form without permission from the publisher, except fair uses permitted under U.S. or applicable copyright law.

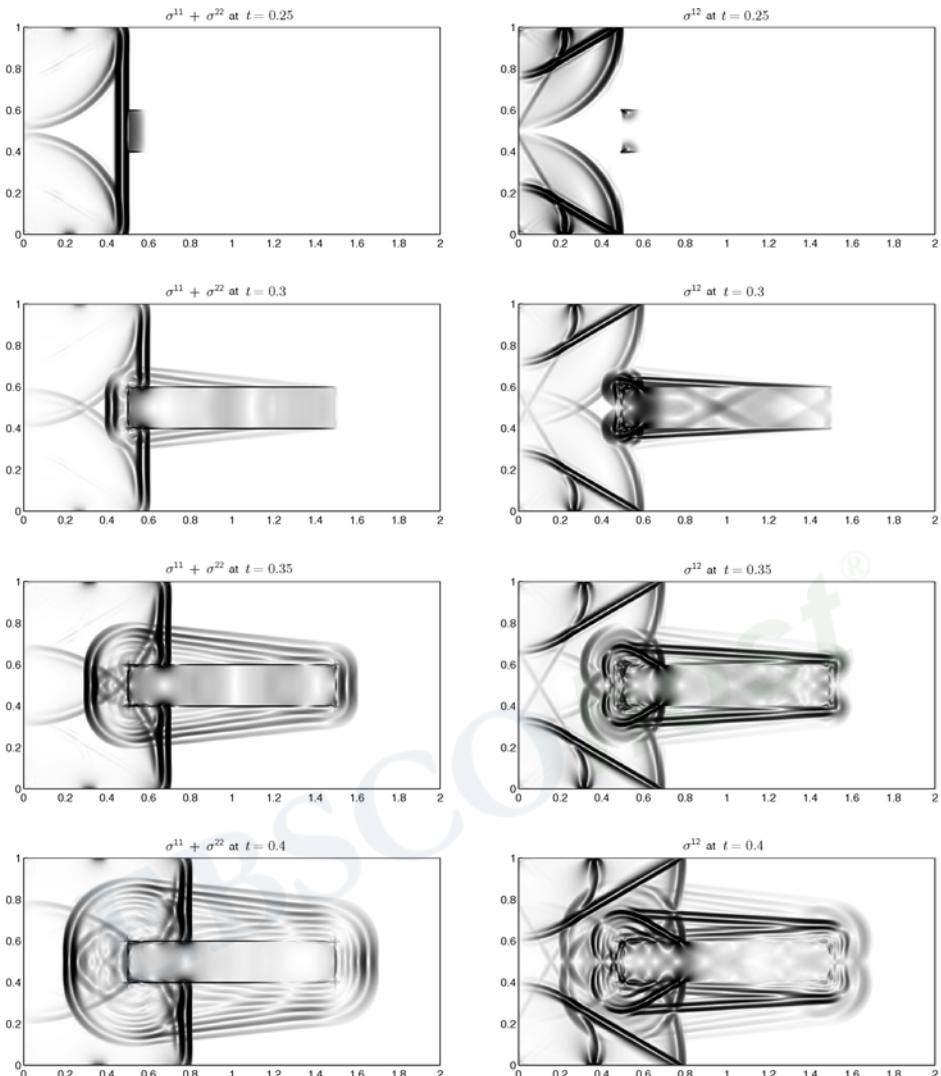


Fig. 22.5. Elastic wave propagation for the example shown in Figure 22.4. Left column: the mean stress $\sigma^{11} + \sigma^{22}$. Right column: the shear stress σ^{12} . [claw/book/chap22/inclusion]

a compressional wave, and the resulting wave motion is illustrated in Figure 22.4. The deformation of the solid is computed as described in Example 22.2. Again the displacements shown are much larger than actual linear displacements should be, and have been greatly exaggerated so that they will be visible.

Note that the compression wave is not purely a P-wave, due to its interaction with the free boundaries at $y = \pm 1$. At about $t = 0.25$ the wave hits the inclusion, which is much stiffer and hence tends to be pushed over as a rigid unit. At later times there is very little distortion of the inclusion, which simply shifts over slightly, launching smaller-amplitude waves into the exterior material at the far end and also along its length due to the resistance

to shear. This is not really rigid motion, however. Elastic waves are rapidly bouncing back and forth in the stiff material to accomplish this motion. Note that $c_p = 20$ and $c_s = 10$ in the stiff region, whereas $c_p = 2$ and $c_s = 1$ outside.

Figure 22.5 shows the wave motion more clearly. Here the gradients of $\sigma^{11} + \sigma^{22}$ and σ^{12} are plotted at various times using a *schlieren* image style in which a highly nonlinear color scale is used, so that even very small-amplitude waves show up distinctly. At time $t = 0.25$ we see that waves are just beginning to travel down the stiff inclusion. By time $t = 0.3$ they have reached the far end. As they move along the bar, they launch waves into the surrounding material. The inclusion then begins to vibrate, giving rise to further waves moving vertically away from it.

Finite Volume Methods on Quadrilateral Grids

Many multidimensional problems of practical interest involve complex geometry, and in general it is not sufficient to be able to solve hyperbolic equations on a uniform Cartesian grid in a rectangular domain. In Section 6.17 we considered a nonuniform grid in one space dimension and saw how hyperbolic equations can be solved on such a grid by using a uniform grid in *computational space* together with a coordinate mapping and appropriate scaling of the flux differences using *capacity form differencing*. The *capacity* of the computational cell is determined by the size of the corresponding physical cell.

In this chapter we consider nonuniform finite volume grids in two dimensions, such as those shown in Figure 23.1, and will see that similar techniques may be used. There are various ways to view the derivation of finite volume methods on general multidimensional grids. Here we will consider a direct physical interpretation in terms of fluxes normal to the cell edges. For simplicity we restrict attention to two space dimensions. For some other discussions of finite volume methods on general grids, see for example [156], [245], [475], [476].

The grids shown in Figures 23.1(a) and (b) are logically rectangular *quadrilateral grids*, and we will concentrate on this case. Each cell is a quadrilateral bounded by four linear segments. Such a grid is also often called a *curvilinear grid*. If we label the cells in a logical manner, indexing “rows” and “columns” by i and j , then cell (i, j) has the four neighbors $(i \pm 1, j)$ and $(i, j \pm 1)$. The grid can be made to wrap around the cylinder by imposing periodic boundary conditions in one direction.

The triangulation shown in Figure 23.1(c), on the other hand, gives an *unstructured grid* for which there is no simple logical structure underlying the connectivity between cells. One must keep track of the neighbors of each cell explicitly. An unstructured triangulation is often easier to generate for complicated geometries than a structured grid such as the one shown in Figure 23.1(a), but may be somewhat more difficult to work with in regard to data structures and the development of fast and accurate solvers. See [20], [324], [385], [472] for some discussions of unstructured grids.

The grids of Figure 23.1(a)–(c) are all *body-fitted grids* that conform to the geometry of the problem. Figure 23.1(d) shows a different approach in which a uniform Cartesian grid is used over most of the domain, but with some smaller irregular cells allowed where the boundary cuts through the grid. Finite volume methods of this type are often called *Cartesian-grid* or *embedded-boundary* methods. For problems with more complicated geometry this approach allows for very easy grid generation, and so it has recently become quite popular. With these

Copyright © 2002, Cambridge University Press.
All rights reserved. May not be reproduced in any form without permission from the publisher, except fair uses permitted under U.S. or applicable copyright law.

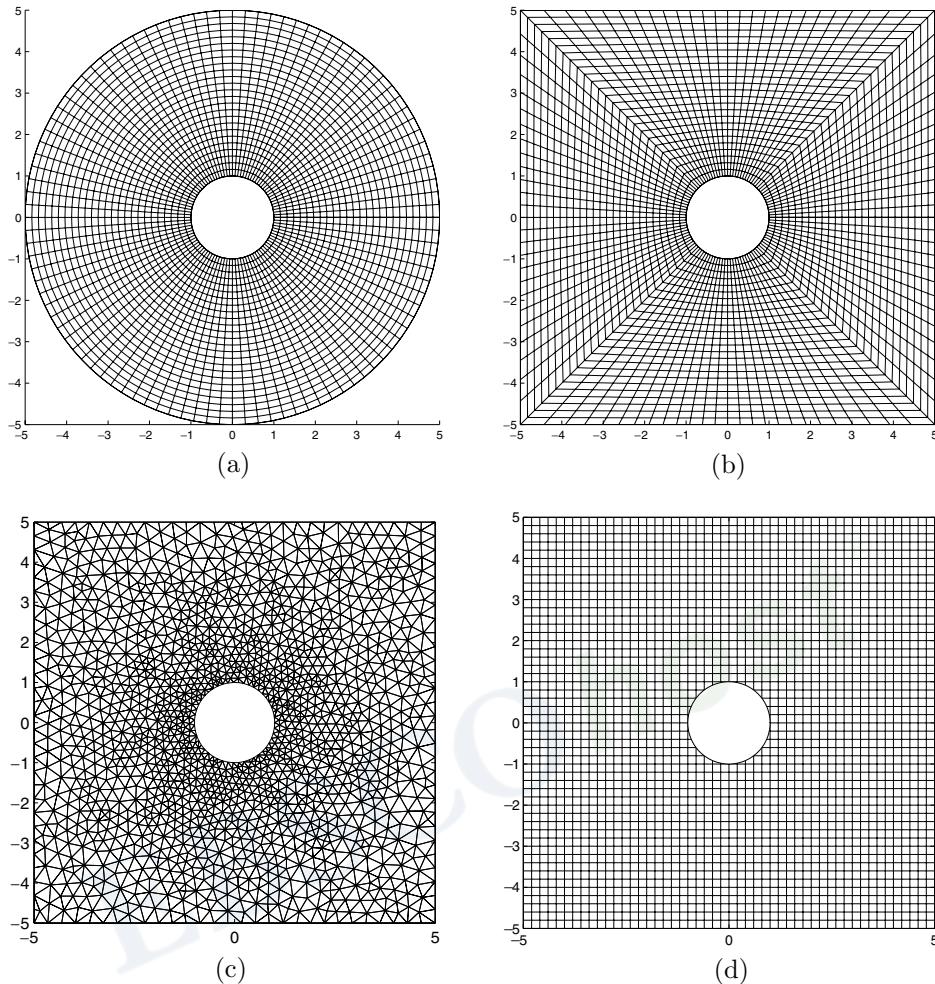


Fig. 23.1. Four possible grids for flow around a cylinder. (a) 25×100 polar grid in $r-\theta$. (b) 25×104 quadrilateral grid interpolating between the cylinder and a square computational domain. (c) An unstructured triangulation. (d) A Cartesian grid with embedded boundary.

methods the main difficulty is in specifying fluxes at the edges of the small cut cells in such a way that good accuracy is obtained and stability is preserved with reasonable-size time steps. A variety of different approaches have been introduced; see for example [7], [31], [57], [59], [77], [104], [140], [277], [278], [358], [363], [490], [492].

23.1 Cell Averages and Interface Fluxes

With a finite volume method we view each discrete value Q as a cell average over a grid cell, which is modified due to fluxes through the edges of the cell, in the case of a conservation law, or more generally by waves moving into the cell from each edge. We start by discussing the case of a conservation law and will see how to extend this to nonconservative linear systems in Section 23.4.

The finite volume approach can be applied on any shape cell \mathcal{C} using the integral form of the conservation law,

$$\frac{d}{dt} \iint_{\mathcal{C}} q(x, y, t) dx dy = - \int_{\partial\mathcal{C}} \vec{n}(s) \cdot \vec{f}(s, t) ds, \quad (23.1)$$

and hence is applicable on any of the grids of Figure 23.1 if a good numerical approximation to the interface flux can be determined. Here $\vec{n}(s)$ is the outward-pointing normal and

$$\vec{f}(s, t) = \begin{bmatrix} f(q(x(s), y(s), t)) \\ g(q(x(s), y(s), t)) \end{bmatrix},$$

with the boundary $(x(s), y(s))$ of \mathcal{C} parameterized by the arclength s . Then

$$\check{F}(s) = \vec{n}(s) \cdot \vec{f}(s, t) = n^x(s) f(q(x(s), y(s), t)) + n^y(s) g(q(x(s), y(s), t))$$

gives the flux per unit length per unit time in the direction $\vec{n}(s)$. Note that for a system of m equations f and g are vectors of length m and \vec{f} is a vector of length $2m$, while the normal $\vec{n}(s)$ contains two scalar components n^x and n^y . (See Section 18.1 for more about the multidimensional notation used here.)

Integrating (23.1) from time t_n to t_{n+1} and dividing by $|\mathcal{C}|$, the area of the cell, gives

$$\begin{aligned} \frac{1}{|\mathcal{C}|} \iint_{\mathcal{C}} q(x, y, t_{n+1}) dx dy &= \frac{1}{|\mathcal{C}|} \iint_{\mathcal{C}} q(x, y, t_n) dx dy \\ &\quad - \frac{1}{|\mathcal{C}|} \int_{t_n}^{t_{n+1}} \int_{\partial\mathcal{C}} \vec{n}(s) \cdot \vec{f}(s, t) ds dt. \end{aligned} \quad (23.2)$$

If Q^n represents the cell average over this cell at time t_n , then this suggests the finite volume method

$$Q^{n+1} = Q^n - \frac{\Delta t}{|\mathcal{C}|} \sum_{j=1}^N h_j \check{F}_j^n, \quad (23.3)$$

where \check{F}_j^n represents a numerical approximation to the average normal flux across the j th side of the cell, N is the number of sides, and h_j is the length of the j th side. The factors Δt and h_j are introduced by taking \check{F}_j^n as an approximation to the interface flux per unit length, per unit time,

$$\check{F}_j^n \approx \frac{1}{\Delta t} \int_{t_n}^{t_{n+1}} \left(\frac{1}{h_j} \int_{\text{side } j} \vec{n} \cdot \vec{f}(s, t) ds \right) dt.$$

This agrees with the normalization used previously on Cartesian grids.