20

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

$$q_t + uq_x + vq_y = 0, \tag{20.1}$$

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 $q(x, y, t) = \mathring{q}(x - ut, y - vt)$, but for our present purposes the Taylor series expansion is more illuminating:

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

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

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

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 $Q^{\Downarrow}$ 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

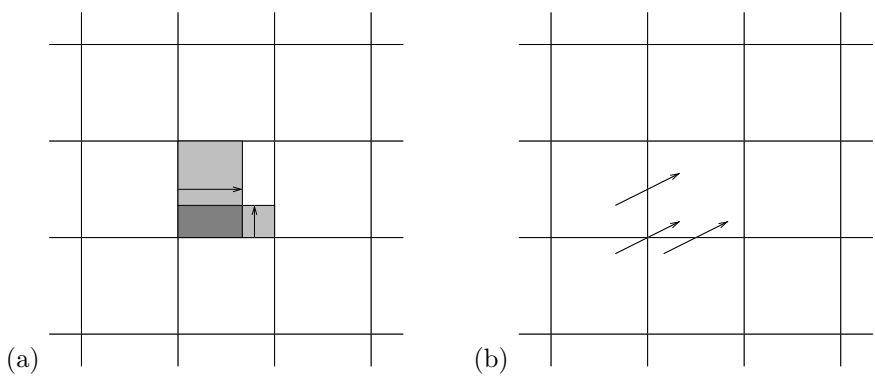

Fig. 20.1. (a) Waves moving normal to the cell interfaces give the updates for the DCU method. (b) The true velocity $(u,\ v)$ is at an angle to the grid, and information from cell $(i-1,\ j-1)$ should also affect the new value in cell $(i,\ j)$. This corner coupling is missing in the DCU method.

method is stable only for Δt small enough that

$$\left|\frac{u\,\Delta t}{\Delta x}\right| + \left|\frac{v\,\Delta t}{\Delta y}\right| \le 1. \tag{20.6}$$

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

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

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 $\mathcal{C}_{ij}$,
- 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

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

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:

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

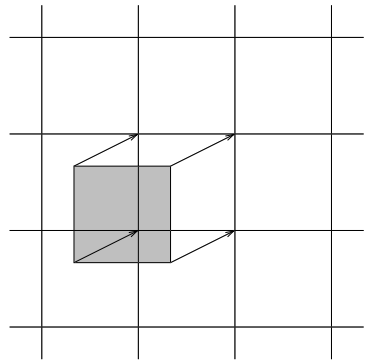

Fig. 20.2. The corner-transport upwind method is obtained by shifting the piecewise constant data by distance $(u\,\Delta t, v\,\Delta t)$ and averaging back on the grid. Alternatively, the new value Q_{ij}^{n+1} is determined by averaging the piecewise constant function over the shaded region shown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

This can be rearranged to yield

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

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 $\mathcal{C}_{i-1,j-1}$ to $\mathcal{C}_{ij}$. 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.

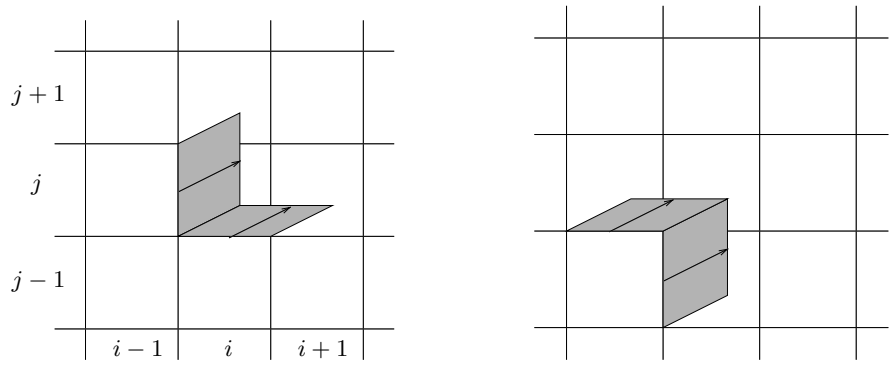

Fig. 20.3. (a) Transverse propagation affecting the fluxes $\tilde{F}_{i+1/2,j}$ and $\tilde{G}_{i,j+1/2}$. (b) Transverse propagation affecting the fluxes $\tilde{F}_{i-1/2,j}$ and $\tilde{G}_{i,j-1/2}$.

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

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

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

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

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

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

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

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

where $\nu^x = u\,\Delta t/\Delta x$ and $\nu^y = v\,\Delta t/\Delta y$. (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

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

The method is stable in the 2-norm provided that $|g| \le 1$ for all choices of ξ and η. Values of g lie a distance at most $\nu^x + \nu^y$ from the point $1 - \nu^x - \nu^y$ in the complex plane, and hence the method is stable for $0 \le \nu^x + \nu^y \le 1$. 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

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

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

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

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 $\max(\nu^x,\ \nu^y) \le 1$. 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,

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

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),

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

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

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

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

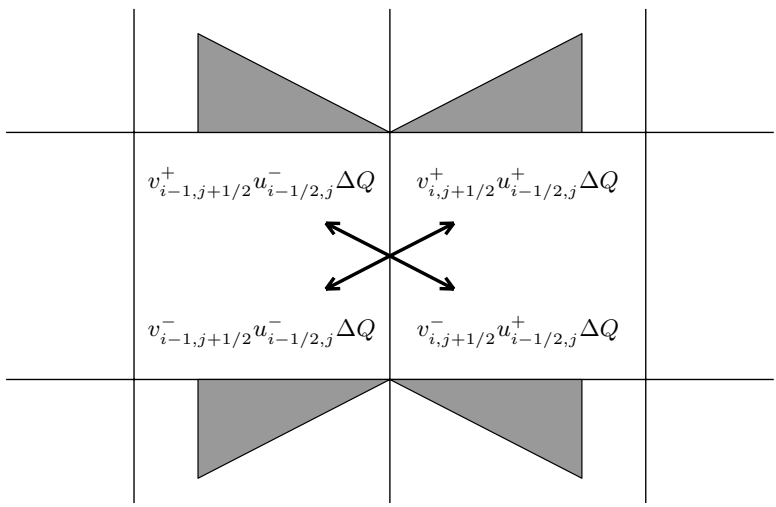

Fig. 20.4. The jump $\Delta Q_{i-1/2,j}$ may propagate in any of four directions, depending on the velocities, and so four neighboring $\tilde{G}$-fluxes may need to be updated.

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

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

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 $\mathcal{A}^{+}\Delta Q_{i-1/2,j} = u^{+}_{i-1/2,j}(Q_{ij} - Q_{i-1,j})$, for example, and produces a down-going transverse fluctuation

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

and an up-going transverse fluctuation

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

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 $\mathcal{A}^{-}\Delta Q_{i-1/2,j} = u^{-}_{i-1/2,j}(Q_{ij} - Q_{i-1,j})$ results in the transverse fluctuations

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

and

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

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:

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

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

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

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

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

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

$$\frac{\Delta t}{\Delta x}\sqrt{u^2 + v^2} \le 1, \tag{20.33}$$

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

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

and the color equation

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

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

$$u_x(x, y) + v_y(x, y) = 0, \tag{20.36}$$

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

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

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

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

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 $\Delta y\, u_{i\pm1/2,j}$ and $\Delta x\, v_{i,j\pm1/2}$ 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,

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

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

$$\begin{aligned} Q_{ij}^{n+1} = Q_{ij} &- \frac{\Delta t}{\Delta x}\left[u^+_{i-1/2,j}(Q_{ij} - Q_{i-1,j}) + u^-_{i+1/2,j}(Q_{i+1,j} - Q_{ij})\right] \\ &- \frac{\Delta t}{\Delta y}\left[v^+_{i,j-1/2}(Q_{ij} - Q_{i,j-1}) + v^-_{i,j+1/2}(Q_{i,j+1} - Q_{ij})\right]. \end{aligned} \tag{20.40}$$

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

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

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 *stream function* $\psi(x, y)$. Any continuous and piecewise differential scalar function $\psi(x, y)$ can be used to define a velocity field via

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

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

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

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

$$\psi(x, y) = x^2 + y^2. \tag{20.44}$$

The resulting velocity field

$$u(x, y) = 2y, \qquad v(x, y) = -2x \tag{20.45}$$

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:

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

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

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

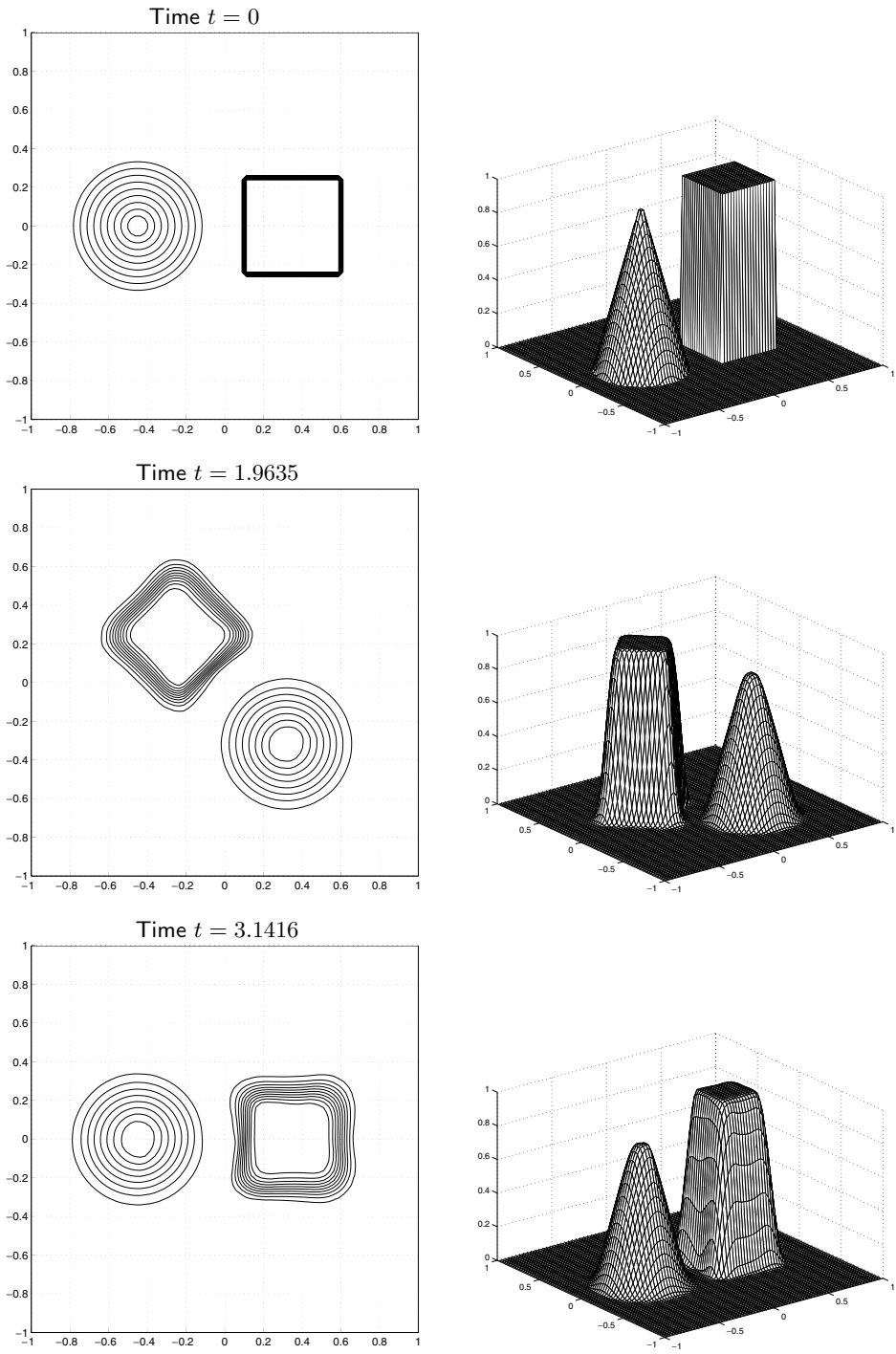

Fig. 20.5. Solid-body rotation from Example 20.1. The solution computed on an 80×80 grid is shown at three different times. Top: $t = 0$; middle: $t = 5\pi/8$; bottom: $t = \pi$. At each time the solution is shown as a contour plot (left) and a mesh plot (right). Contour lines are at the values $q =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

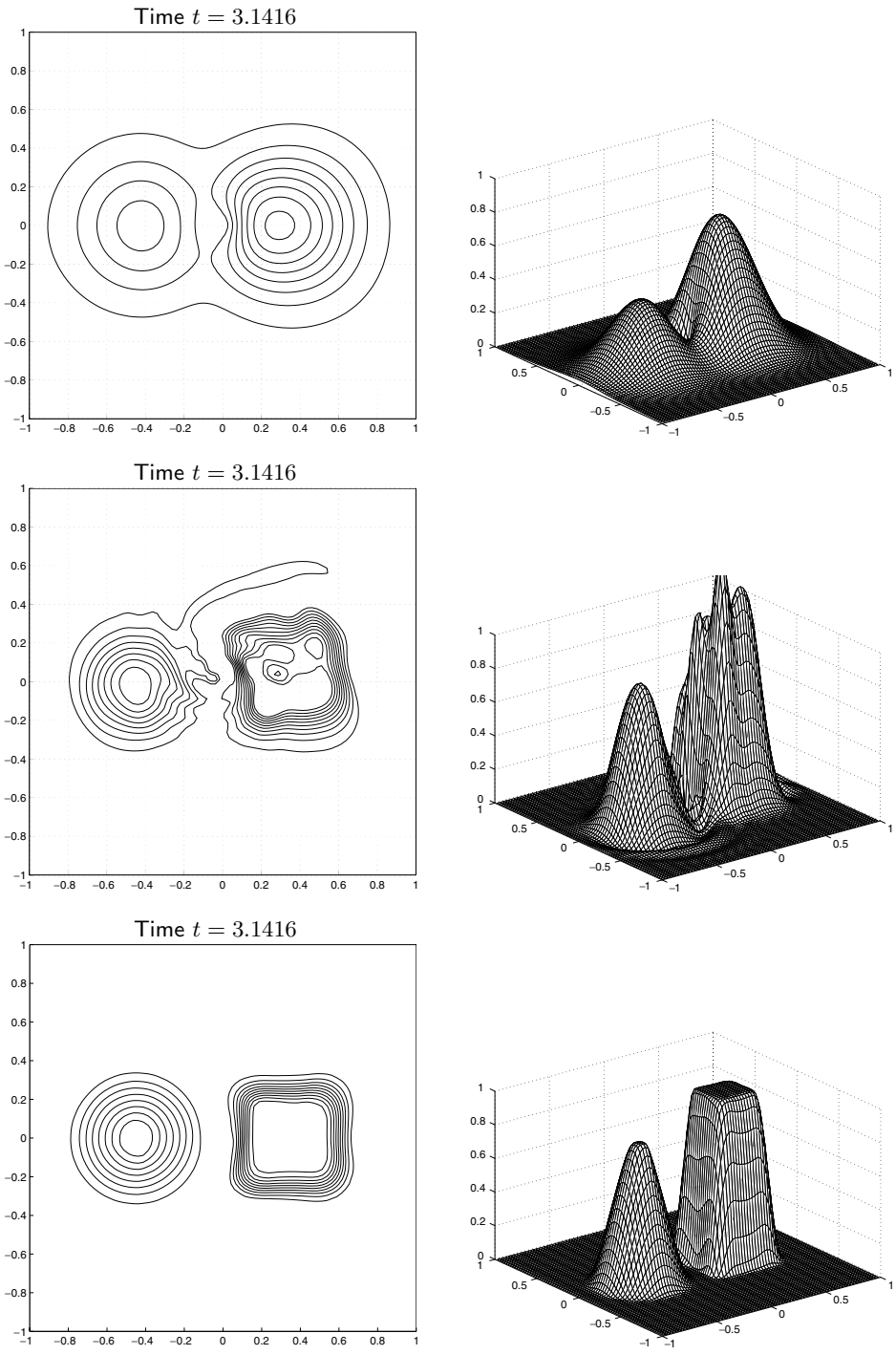

Fig. 20.6. Solid-body rotation from Example 20.1. Results obtained with the first-order CTU method (top), the second-order "Lax–Wendroff" method with no limiter (middle), and dimensional splitting with the high-resolution method in each direction (bottom). Each is shown only at the final time $t = \pi$.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

and in the y-direction

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

The fluctuations can be defined simply as

$$\begin{aligned}
\mathcal{A}^{\pm}\Delta Q_{i-1/2,j} &= s^{\pm}_{i-1/2,j}\mathcal{W}_{i-1/2,j},\\
\mathcal{B}^{\pm}\Delta Q_{i,j-1/2} &= s^{\pm}_{i,j-1/2}\mathcal{W}_{i,j-1/2},
\end{aligned} \tag{20.49}$$

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),

$$\begin{aligned}
\tilde{F}_{i-1/2,j} &:= \tilde{F}_{i-1/2,j} + \frac{1}{2}|s_{i-1/2,j}|\left(1 - \frac{\Delta t}{\Delta x}|s_{i-1/2,j}|\right)\tilde{\mathcal{W}}_{i-1/2,j},\\
\tilde{G}_{i,j-1/2} &:= \tilde{G}_{i,j-1/2} + \frac{1}{2}|s_{i,j-1/2}|\left(1 - \frac{\Delta t}{\Delta y}|s_{i,j-1/2}|\right)\tilde{\mathcal{W}}_{i,j-1/2}.
\end{aligned} \tag{20.50}$$

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

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

We then set

$$\begin{aligned}
\tilde{G}_{i-1,j-1/2} &:= \tilde{G}_{i-1,j-1/2} - \frac{1}{2}\frac{\Delta t}{\Delta x}\hat{v}^{-}_{i-1/2,j}s^{-}_{i-1/2,j}(Q_{ij} - Q_{i-1,j}),\\
\tilde{G}_{i-1,j+1/2} &:= \tilde{G}_{i-1,j+1/2} - \frac{1}{2}\frac{\Delta t}{\Delta x}\hat{v}^{+}_{i-1/2,j}s^{-}_{i-1/2,j}(Q_{ij} - Q_{i-1,j}),\\
\tilde{G}_{i,j-1/2} &:= \tilde{G}_{i,j-1/2} - \frac{1}{2}\frac{\Delta t}{\Delta x}\hat{v}^{-}_{i-1/2,j}s^{+}_{i-1/2,j}(Q_{ij} - Q_{i-1,j}),\\
\tilde{G}_{i,j+1/2} &:= \tilde{G}_{i,j+1/2} - \frac{1}{2}\frac{\Delta t}{\Delta x}\hat{v}^{+}_{i-1/2,j}s^{+}_{i-1/2,j}(Q_{ij} - Q_{i-1,j}).
\end{aligned} \tag{20.52}$$

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 $v_{i,j\pm1/2}$ and $v_{i-1,j\pm1/2}$ 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

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

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

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

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 $\mathcal{S}(t)\mathring{q}$ is the (unique) entropy-satisfying solution at time t, $(\mathcal{S}(t)\mathring{q})(x, t) = q(x, y, t)$. 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.,*

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

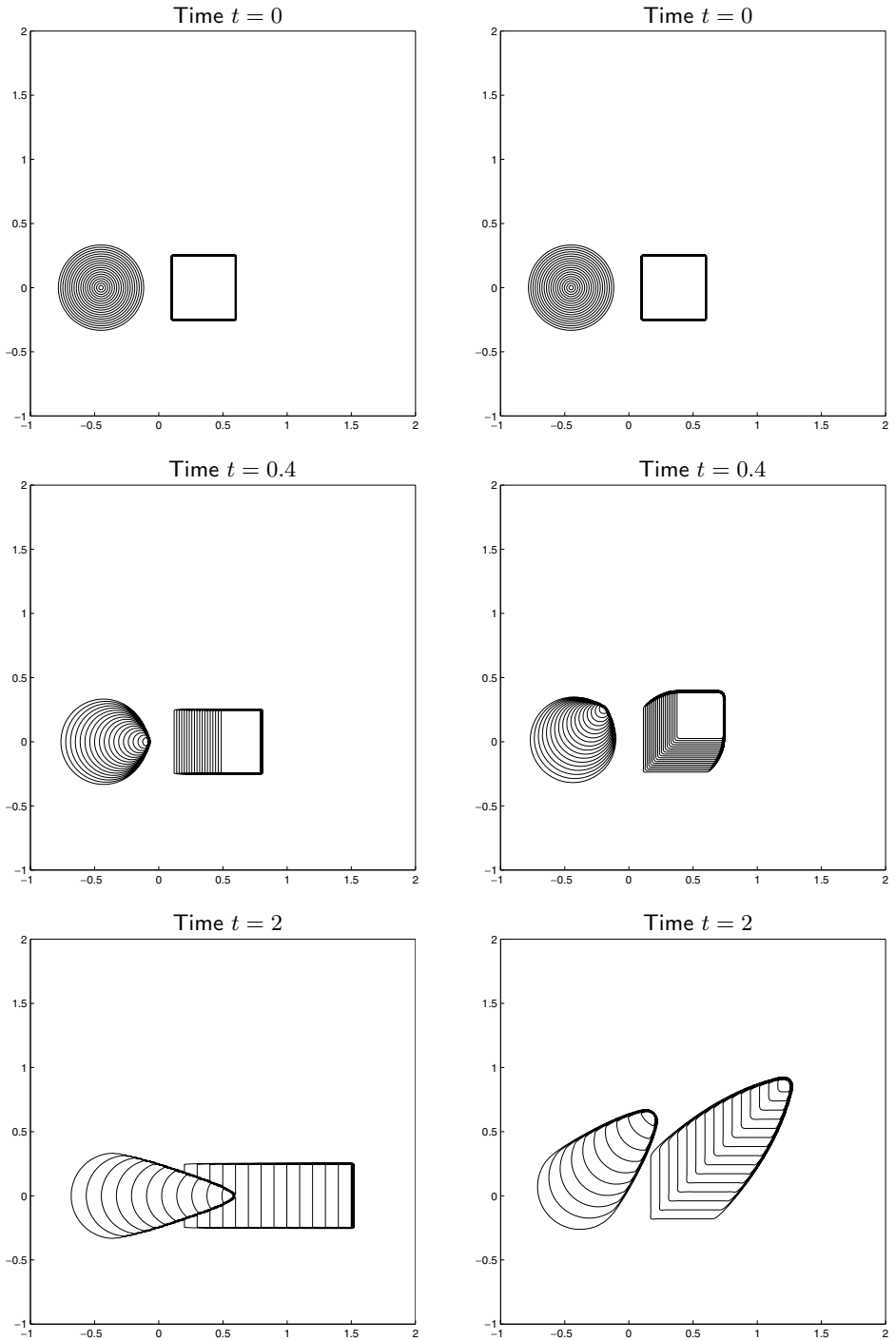

Fig. 20.7. Solution of the two-dimensional Burgers equation (20.53) at angle θ to the x-axis. Left column: $\theta = 0$. Right column: $\theta = \pi/4$. Contour lines are at $u = 0.05 : 0.05 : 0.95$. `[claw/book/chap20/burgers]`

and

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

as $\Delta t \to 0$ *and* $n \to \infty$ *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

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

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

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

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

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

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.

(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 $\mathring{q}(x, y) \equiv 0$ and boundary conditions

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

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 $q_t + u(y)q_x = 0$.