JOURNAL OF COMPUTATIONAL PHYSICS 56, 65–92 (1984)

A Rotationally Biased Upwind Difference Scheme for the Euler Equations

STEPHEN F. DAVIS*

Institute for Computer Applications in Science and Engineering, NASA Langley Research Center, Hampton, Virginia 23665

Received September 27, 1983

The upwind difference schemes of Godunov, Osher, Roe and van Leer are able to resolve one-dimensional steady shocks for the Euler equations within one or two mesh intervals. Unfortunately, this resolution is lost in two dimensions when the shock crosses the computing grid at an oblique angle. To correct this problem, a numerical scheme is developed which automatically locates the angle at which a shock might be expected to cross the computing grid then constructs separate finite difference formulas for the flux components normal and tangential to this direction. Numerical results are presented which illustrate the ability of this new method to resolve steady oblique shocks.

1. INTRODUCTION

Beginning with the work of Godunov [2], considerable effort has been expended seeking numerical methods which can solve the equations of gas dynamics accurately and sharply resolve discontinuities.

In the case of one-dimensional flows containing steady discontinuities, this effort has paid off. Work by van Leer [14–16], Roe [12], Osher [1, 11], Harten [5] and others indicates that the mechanism of "shock capture" is well understood and that it is now possible to design second-order-accurate schemes which resolve steady discontinuities within one or two mesh intervals without wiggles. A recent paper by Harten and Hyman [3] indicates that these results can be extended to one-dimensional flows with moving discontinuities.

In the case of two-dimensional flows, some progress has been made but there is room for improvement. It is well known that the one-dimensional results cited above can be reproduced in two-dimensional calculations if the computing grid is chosen so that only one set of grid lines crosses a discontinuity in the flow. This would appear to be the optimum way to account for the two-dimensional aspects of the flow but in practice it is very difficult to choose an appropriate grid either a priori or adaptively during a calculation. We shall not examine this approach here.

* Research supported by the National Aeronautics and Space Administration under NASA Contract NASI-17070 while the author was in residence at ICASE, NASA Langley Research Center, Hampton, Virginia 23665.

0021-9991/84 $3.00

An alternative approach is to use a locally one-dimensional or fractional step method (cf. Yanenko [18]) with a high-resolution, one-dimensional scheme in each one-dimensional step. This method ignores the two-dimensional orientation of discontinuities but appears to work surprisingly well (cf. Yee *et. al.* [19], Woodward and Colella [17]). Since this method is also simple to program, it is quite attractive for practical application.

Despite these attractions, we attempt to do better. In particular, in this paper, we derive a two-dimensional scheme which can sharply resolve weak shocks which are skew to the fixed computing grid. Locally one-dimensional methods have difficulty dealing with this situation. An outline of the paper follows.

In Section 2, we briefly discuss the ideas behind the Godunov [2] method and show heuristically why it resolves steady shocks so well. We then describe various simplifications of Godunov's method that retain its shock-resolving properties, including the flux vector splitting method of van Leer [16] which we later use.

In Section 3, we derive the numerical scheme. Since our scheme involves rotated differences we first review some early work by Jameson [6] in which he applies rotated differences to the transonic full potential equations. We then derive a rotated difference scheme for the Euler equations. Our scheme differs from Jameson's in that for our scheme both the finite difference formula and the computational stencil vary with angle whereas for Jameson's scheme only the finite difference formula varies.

In Section 4, we consider the choice of angle for the rotated differences. We present a choice which seems to be appropriate for shock resolution with the Euler equations and then we briefly discuss more general discontinuities and more general conservation laws.

Section 5 contains numerical computations using this scheme and comparisons with other schemes.

Section 6 summarizes the present work, discusses the results of the present work and presents an outline for future work.

2. Godunov-Type Methods and One-Dimensional Problems

In this section we study the Godunov method for one-dimensional systems of conservation laws

$$w_t + f(w)_x = w_t + A(w)w_x = 0 \tag{2.1a}$$

where

$$A(w) = \frac{df}{dw}(w). \tag{2.1b}$$

In particular we wish to determine which features of this scheme are responsible for its ability to resolve steady discontinuities. This discussion follows closely that of Harten *et al.* [4], so the reader is encouraged to consult their paper for additional details.

The construction of Godunov's scheme is as follows. At discrete time levels t_n, $n = 0, 1, \ldots$, the numerical approximation $v(x, t_n)$ to the solution $u(x, t_n)$ of (2.1) is taken to be a piecewise constant function of x, i.e.,

$$v(x, t_n) = v_j^n \qquad \text{for} \quad x \in I_j = ((j - \tfrac{1}{2})\,\Delta x, (j + \tfrac{1}{2})\,\Delta x). \tag{2.2}$$

To calculate the numerical approximation at the next time level $t_{n+1} = t_n + \Delta t$ we first solve exactly the problem of the breakup of the discontinuities in $v(x, t_n)$ at all interfaces. This means that at each discontinuity of $v(x, t_n)$ we solve a local Riemann problem. The solution to the Riemann problem due, for example, to the jump at the interface between I_j and I_{j+1} is a similarity solution which depends only on the states v_j^n and v_{j+1}^n and the ratio $(x - (j + \frac{1}{2})\,\Delta x)/(t - t_n)$. We denote this solution by $u\{(x - (j + \frac{1}{2})\,\Delta x)/(t - t_n); v_j^n, v_{j+1}^n\}$. Since signals propagate with finite velocity, there will be no interaction between neighboring Riemann problems if $(|a_{\max}|\,\Delta t)/\Delta x < \frac{1}{2}$, where $|a_{\max}|$ is the largest signal speed. In this case the exact solution of (2.1) with initial conditions (2.2), denoted by $u_n(x, t_n)$, is given by

$$u_n(x, t) = u\{(x - (j + \tfrac{1}{2})\,\Delta x)/(t - t_n); v_j^n, v_{j+1}^n\}$$

for

$$j\,\Delta x < x < (j + 1)\,\Delta x, \qquad t_n < t < t_{n+1}. \tag{2.3}$$

Godunov obtains a piecewise constant approximation $v(x, t_{n+1})$ for the next time step, by averaging $u_n(x, t_{n+1})$, i.e., by defining

$$v_j^{n+1} = 1/\Delta x \int_{I_j} u_n(x, t_{n+1})\,dx. \tag{2.4}$$

A three-point explicit finite difference scheme for (2.1) of the form

$$v_j^{n+1} = v_j^n - \frac{\Delta t}{\Delta x}\,[F(v_j^n, v_{j+1}^h) - F(v_{j-1}^h, v_j^n)] \tag{2.5}$$

is said to be in conservation form. The function F appearing in (2.5) is called the numerical flux function.

If we evaluate the integral in (2.4) by integrating (2.1) over $I_j x(t_n, t_{n+1})$ we can show that Godunov's method is in conservation form with a numerical flux function given by

$$F(v, \omega) = f(u(0; v, \omega)). \tag{2.6}$$

Details can be found in the paper by Harten, Lax, and van Leer [4].

Next we examine how this scheme resolves stationary shocks. For the sake of clarity we consider only a single conservation law. Qualitatively similar results can be derived for systems of conservation laws (see Lax [8]) but the details are far more complicated.

A shock is an exact discontinuous solution to the Riemann problem of the form

$$u_n(x,t) = \begin{cases} u_L, & x < st \\ u_R, & x > st \end{cases} \tag{2.7}$$

where s satisfies the Rankine–Hugoniot jump condition

$$s(u_R - u_L) = f(u_R) - f(u_L) \tag{2.8}$$

and the entropy condition

$$a_L > s > a_R \tag{2.9}$$

where

$$a_L = \frac{df}{du}(u_L), \qquad a_R = \frac{df}{du}(u_R). \tag{2.10}$$

A steady shock is a shock with $s = 0$.

If the initial conditions represent a steady shock located on the interface between cells I_{-1}, I_0, i.e., the initial data are

$$v_j^0 = u_0(x, t_0) = \begin{cases} u_L & \text{for } x \in I_j \text{ and } j < 0 \\ u_R & \text{for } x \in I_j \text{ and } j \geqslant 0 \end{cases} \tag{2.11}$$

then Eq. (2.9) will be satisfied exactly for all time levels and (2.11) is an exact solution to Godunov's method.

We now show that, if the initial conditions represent a steady shock located within a cell, say, I_0, then Eq. (2.4) will yield

$$v_j^0 = \begin{cases} u_L & \text{for } j < 0 \\ u_m & \text{for } j = 0 \\ u_R & \text{for } j > 0 \end{cases} \tag{2.12}$$

where u_m is some intermediate value between u_L and u_R.

Since Δt is chosen so that waves do not interact, at time t_1 the solution to the Riemann problem at the interface between I_{-1} and I_0 will consist of a shock moving at the speed

$$S_L = \frac{f(u_m) - f(u_L)}{u_m - u_L}. \tag{2.13}$$

If the shock is weak

$$f(u_m) \approx f(u_R) + \frac{df}{du}(u_R)(u_m - u_R) \approx f(u_R) + a_R(u_m - u_R). \tag{2.14}$$

Substitute this into (2.13) and note that $f(u_L) = f(u_R)$, so

$$S_L \approx a_R \frac{(u_m - u_R)}{(u_m - u_L)}. \tag{2.15}$$

Since u_m is between u_L and u_R, $(u_m - u_R)$ and $(u_m - u_L)$ will have opposite signs and their quotient will be negative. Equation (2.9) with $s = 0$ shows that a_R is negative. Thus S_L is positive, the shock moves to the right and

$$u(0; u_L, u_m) = u_L. \tag{2.16}$$

By a similar argument we can show that on the interface between I_0 and I_1

$$u(0; u_m, u_R) = u_R. \tag{2.17}$$

Substitution of (2.16) and (2.17) into (2.6) and (2.5) yields

$$v_0^1 = v_0^0 - \frac{\Delta t}{\Delta x}[f(u_R) - f(u_L)] = v_0^0. \tag{2.18}$$

Thus (2.12) is an exact solution to Godunov's method.

We note in passing that the choice of numerical fluxes

$$F(u_L, u_m) = f(u_L)$$

and

$$F(u_m, u_R) = f(u_R)$$

which does not depend on u_m, confines the effect of the averaging which created u_m to a single cell. This would not be true of a conventional finite difference method whose numerical fluxes depended on all of its arguments. Subsequent time steps would propagate the effect of u_m through the entire domain and thus spread the shock over many cells.

The construction of solutions of Riemann problems for nonlinear systems is a complicated iterative procedure. In addition, Eq. (2.6) shows that although the entire Riemann solution is computed, only values at cell interfaces are actually used. For this reason much recent research has been devoted to the construction of numerical flux functions which retain the shock-capturing ability of Godunov's scheme but which are simpler to construct.

This is accomplished either by approximating the solution to the exact Riemann problem or by approximating certain features of the Godunov numerical flux. In the following we briefly describe some successful methods of both types. The reader is referred to the original papers for details.

Harten *et al.* [4] refer to methods which approximate Riemann solutions as Godunov-type methods. In their paper they construct two such methods by using the conservation laws (1.1) to lump together many of the features of the exact Riemann solution. These methods should be extremely simple to implement but, at present, no numerical results are available.

Another Godunov-type scheme is Roe's [12] method. Like Godunov, Roe begins with the piecewise constant approximation (2.2) but he replaces the solution to a Riemann problem for the exact equation due, for example, to the jump at the interface between I_j and I_{j+1} by the solution to a Riemann problem for the linearized equations

$$u_t + A(v_j^n, v_{j+1}^n)u_x = 0 \tag{2.19}$$

where the matrix A is chosen to satify

$$A(v_j^n, v_{j+1}^n) \cdot (v_{j+1}^n - v_j^n) = f(v_{j+1}^n) - f(v_j^n). \tag{2.20}$$

Conditions (2.20) assure that the Godunov method using the Roe approximate Riemann solution can be written in conservation form (2.5).

Osher and Solomon [11] construct a numerical flux which approximates the directional bias of the Godunov flux. To accomplish this, they write the flux difference as a path integral in state space as follows:

$$\Delta f := f(q) - f(p) = \int_p^q A(u)\, du. \tag{2.21}$$

The path of integration Γ is chosen to be piecewise parallel to the right eigenvectors R_k of A. Without going into detail, we point out that this choice of paths permits the right-hand side of (2.21) to be split as follows:

$$\int_p^q A(u)\, du = \int_p^q A^+(u)\, du + \int_p^q A^-(u)\, du. \tag{2.22}$$

Here, the integrals on the right of (2.22) are path integrals along those parts of the path Γ which are parallel to right eigenvectors R_k corresponding to the positive and negative eigenvalues of A, respectively. With this notation, the Osher–Solomon numerical flux can be written in the alternative forms:

$$F_{\text{os}}(p, q) = f(p) + \int_p^q A^-(u)\, du = f(q) - \int_p^q A^+(u)\, du. \tag{3.23}$$

The numerical flux corresponding to each method that we have dicussed satisfies

$$F(p, q) = f(p) \tag{2.24a}$$

if all eigenvalues of A are >0 and

$$F(p, q) = f(q) \tag{2.24b}$$

if all eigenvalues of A are <0. In [16] van Leer constructs a numerical flux functions for the Euler equations which satisfies (2.24a), (2.24b) and which permits steady shocks to be resolved within two cells.

To accomplish this van Leer splits the flux $f(w)$ of (2.1) for the Euler equations into a forward flux $f^+(w)$ and a backward flux $f^-(w)$ which satisfies

$$f(w) = f^+(w) + f^-(w) \tag{2.25}$$

where

$$f^+(w) = f(w), \qquad f^-(w) = 0 \tag{2.26a}$$

when the Mach number $M > 1$ and

$$f^+(w) = 0, \qquad f^-(w) = f(w) \tag{2.26b}$$

when the Mach number $M < -1$. The split fluxes are required to be continuously differentiable so that computed numerical solutions will be smooth. van Leer requires that the Jacobian matrices of the split fluxes $df^{\pm}/dw$ each have an eigenvalue vanish for $|M| < 1$ and he shows that this condition permits steady shocks to be resolved within two cells. Other conditions are imposed to determine the particular form of van Leer's split fluxes. These conditions and the reason for their choice are described in van Leer's paper.

For the one-dimensional Euler equations with ideal gas law considered as a function of density ρ, sound speed c and Mach number M, the resulting splittings for $|M| < 1$ are

(1) mass

$$\begin{aligned} \rho u = f_{\text{mass}} &= f^+_{\text{mass}} + f^-_{\text{mass}} \\ &= \rho c M = \rho c\{\tfrac{1}{2}(M+1)\}^2 - \rho c\{\tfrac{1}{2}(-M+1)\}^2 \end{aligned} \tag{2.27}$$

(2) momentum

$$\begin{aligned} \rho u^2 + p = f_{\text{momentum}} &= f^+_{\text{momentum}} + f^-_{\text{momentum}} \\ &= \rho c^2(M^2 + 1/\gamma) = \rho c^2 \left(\frac{1}{2}(M+1)\right)^2 \left(\frac{\gamma-1}{\gamma} M + 2/\gamma\right) \\ &\quad + \rho c^2 \left(\frac{1}{2}(-M+1)\right)^2 \left(-\frac{\gamma-1}{\gamma} M + 2/\gamma\right) \end{aligned} \tag{2.28}$$

(3) total energy

$$\begin{aligned} (e+p)u = f_{\text{energy}} &= f^+_{\text{energy}} + f^-_{\text{energy}} = \rho c^3 M(\tfrac{1}{2}M^2 + 1/(\gamma-1)) \\ &= \frac{\gamma^2}{2(\gamma^2-1)} \frac{(f^+_{\text{momentum}})^2}{(f^+_{\text{mass}})} + \frac{\gamma^2}{2(\gamma^2-1)} \frac{(f^-_{\text{momentum}})^2}{(f^-_{\text{mass}})}. \end{aligned} \tag{2.29}$$

The reader is referred to van Leer's paper for a detailed derivation of these expressions. The numerical flux based on this splitting is

$$F(u, v) = f^+(u) + f^-(v). \tag{2.30}$$

In the next section we incorporate this numerical flux into a two-dimensional rotated difference scheme.

3. Derivation of a Rotated Difference Scheme

In a supersonic region, the method of Murman and Cole [10] solves the transonic potential equation by replacing derivatives in the streamwise direction with upwind difference approximations and by replacing derivatives in the direction normal to the streamlines with central difference approximations. This is easy to do when the computing grid is approximately aligned with the streamwise and normal directions but is difficult otherwise.

Jameson [6] overcomes this problem in the following way. He writes the transonic potential equation in a coordinate system aligned with and normal to the local streamwise direction. He then expresses the derivatives in the streamwise-normal coordinate system in terms of derivatives in the coordinate system of his computing grid, making note of which terms come from streamwise derivatives and which terms come from normal derivatives. Finally, those terms which come from streamwise derivatives are approximated by upwind difference formulas and those terms which come from normal derivatives are approximated by central difference formulas. This creates a very effective method.

In the following we derive a method for the Euler equations which is based on this rotated difference idea. In particular, we attempt to choose a local coordinate system which permits us to apply the one-dimensional theory of Section 2. This means that our local coordinate directions must be normal and tangential to potential shock directions. In Section 4 we show how to determine these directions. Here we assume that the directions are known and derive difference formulas that are based on these directions. We note in passing that shocks are approximtely normal to the streamlines in transonic flows. Thus our choice of local coordinate system is equivalent to Jameson's in this case.

Consider a local cartesian coordinate system chosen as above. Such a coordinate system has coordinates (x', y'). The coordinate system of the computing grid, assumed for simplicity to be cartesian, has coordinates (x, y). Dependent variables measured in the local and global coordinate systems are primed and unprimed, respectively. This geometry is shown in Fig. 1.

Since the Euler equations are invariant under rotation, they can be written immediately in the local coordinate system as

$$\frac{\partial w'}{\partial t} + \frac{\partial f'(w')}{\partial x'} + \frac{\partial g'(w')}{\partial y'} = 0 \tag{3.1}$$

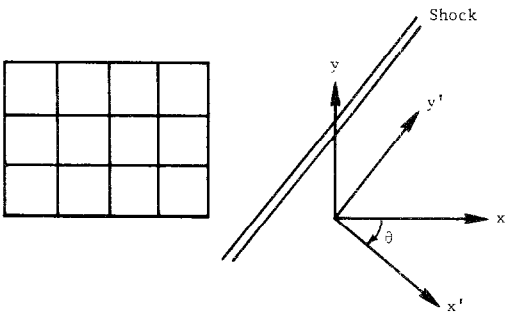

FIG. 1. Geometry of local and global coordinate systems.

where

$$w' = [\rho, \rho u', \rho v', e]^T$$

$$f' = [\rho u', \rho u'^2 + p, \rho u' v', (e + p)u']^T$$

$$g' = [\rho v', \rho u' v', \rho v'^2 + p, (e + p)v']^T \tag{3.2}$$

$$e = p/(\gamma - 1) + \tfrac{1}{2}\rho(u'^2 + v'^2) \tag{3.3}$$

and

$$u' = u \cos\theta + v \sin\theta$$

$$v' = -u \sin\theta + v \cos\theta. \tag{3.4}$$

If the coordinate system is chosen as described in the vicinity of a plane shock, the derivatives in the second term on the right of (3.1) exist and this term is equal to zero. This reduces the two-dimensional problem, locally, to the one-dimensional problem described in the previous section. We attempt to construct a numerical scheme which mimicks this behavior.

If we express the first term on the right of (3.1) in global coordinates we get

$$\frac{\partial f'}{\partial x'} = \cos\theta \frac{\partial f'}{\partial x} + \sin\theta \frac{\partial f'}{\partial y}. \tag{3.5}$$

This can be approximated by a finite difference expression

$$\frac{\cos\theta}{\Delta x}[F'(W'_{i+1,j}, W'_{i,j}) - F'(W'_{i,j}, W'_{i-1,j})]$$

$$+ \frac{\sin\theta}{\Delta y}[F'(W'_{i,j+1}, W'_{i,j}) - F'(W'_{i,j}, W'_{i,j-1})] \tag{3.6}$$

where F' is an appropriately chosen numerical flux and W' is a numerical approx-

imation to w. The conditions that this expression be zero across a steady shock independent of Δx and Δy are

$$[F'(W'_{i+1,j}, W'_{i,j}) - F'(W'_{i,j}, W'_{i-1,j})] = 0$$

and

$$[F'(W'_{i,j+1}, W'_{i,j}) - F'(W'_{i,j}, W'_{i,j-1})] = 0. \tag{3.7}$$

The numerical flux functions corresponding to methods discussed in Section 2 admit solutions to (3.7) which spread steady shocks over only one or two mesh intervals. For this reason we choose F' to be one of these numerical flux functions. For this work we chose the van Leer flux vector splitting but any other method discussed would have sufficed.

The second term on the right of (3.1) says that g' is constant along lines which are parallel to a shock. To approximate this condition on a discrete grid we must make an assumption about the behavior of g' between grid points. For lack of any better information we assume that g' varies linearly between grid points.

Consider a local coordinate system defined by a rotation through an angle θ as shown in Fig. 2. Define

$$l = \Delta x/(-\tan\theta). \tag{3.8}$$

Then

$$l/\Delta y = \Delta x/(-\Delta y \tan\theta) = -\frac{\Delta x \cos\theta}{\Delta y \sin\theta}. \tag{3.9}$$

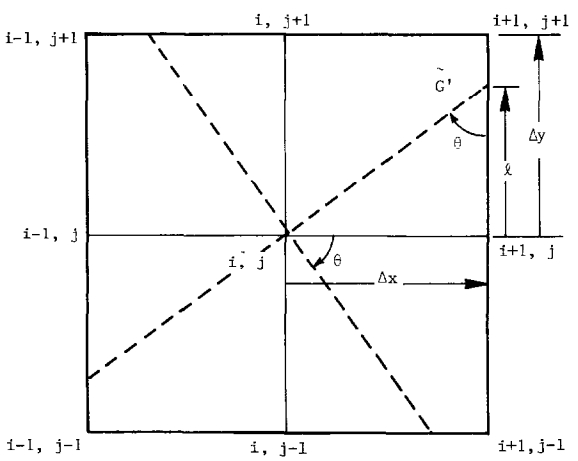

FIG. 2. Construction of tangential flux.

In the following we let $g'_{i,j}$ denote $g'(w'(x_i, y_j))$. The condition that g' varies linearly between grid points (x_{i+1}, y_j) and (x_{i+1}, y_{j+1}) can be written as

$$(g'_{i+1,j+1} - g'_{i+1,j})/\Delta y = (\tilde{g}' - g'_{i+1,j})/l$$

or

$$-(g'_{i+1,j} - \tilde{g}')\frac{\sin\theta}{\Delta x} + (g'_{i+1,j+1} - g'_{i+1,j})\frac{\cos\theta}{\Delta y} = 0. \tag{3.10}$$

Here $\tilde{g}'$ denotes $g'(w(x_{i+1}, y_j + l))$. The condition that g' be constant along a line parallel to a shock can be written

$$\tilde{g}' = g'_{i,j}. \tag{3.11}$$

If we express the second term on the right of (3.1) in global coordinates, we get

$$\frac{\partial g'}{\partial y'} = -\sin\theta\frac{\partial g'}{\partial x} + \cos\theta\frac{\partial g'}{\partial y}. \tag{3.12}$$

We allow the numerical flux on the cell boundary between the points (x_i, y_j) and (x_{i+1}, y_j) to depend on W' values at the points (x_i, y_j), $(x_i, y_{j\pm1})$, (x_{i+1}, y_j) and $(x_{i+1}, y_{j\pm1})$. We allow the numerical flux on the cell boundary between the points (x_i, y_j) and (x_i, y_{j+1}) to depend on W' values at the points (x_i, y_j), $(x_{i\pm1}, y_j)$, (x_i, y_{j+1}) and $(x_{i\pm1}, y_{j+1})$. Then the numerical approximation to (3.12) takes the form

$$-\frac{\sin\theta}{\Delta x}[G'(W'_{i+1,j}, W'_{i+1,j\pm1}, W'_{i,j}, W'_{i,j\pm1}) - G'(W'_{i,j}, W'_{i,j\pm1}, W_{i-1,j}, W'_{i-1,j\pm1})]$$

$$+\frac{\cos\theta}{\Delta y}[G'(W'_{i,j}, W'_{i\pm1,j}, W'_{i,j+1}, W'_{i\pm1,j+1}) - G'(W'_{i,j-1}, W'_{i\pm1,j-1}, W'_{i,j}, W'_{i\pm1,j})]. \tag{3.13}$$

Since (3.10) and (3.11) represent a discrete approximation to (3.12) in the vicinity of a shock, we choose the numerical flux values in (3.13) to be equal to the corresponding physical flux values in (3.10) and (3.11). We note that by choosing lines parallel to a shock which pass through different grid points, we can construct other expressions which are similar to (3.10) and (3.11). Linear combinations of these formulas can be used to determine numerical flux functions for (3.13). Thus the formulas presented below are not unique but they have performed well in practice.

To make our G' numerical flux look similar to our F' numerical flux, we split it up as follows:

$$G'(W'_{i+1,j}, W'_{i+1,j\pm1}, W'_{i,j}, W'_{i,j\pm1}) = \tfrac{1}{2}[(1+Q)\,G'^{+}(W'_{i,j}, W'_{i,j\pm1}) + (1-Q)\,G'^{-}(W'_{i+1,j}, W'_{i+1,j\pm1})]. \quad (3.14)$$

Q is a parameter chosen to assure that the overall method remains stable. Numerical experiments indicate that if the method is stable, the computed results are not sensitive to the particular choice of Q but more study of these matters is needed. Further discussion on the choice of Q appears in Section 5. Table I specifies how G'^{+} and G'^{-} depend on the angle θ.

Until now we have discussed the derivation of numerical fluxes of local variables in local coordinate directions. To compute a global solution we need numerical fluxes of global variables in global coordinates in terms of the flux of local variables in local coordinates as shown below.

Case 1. Flux of a scalar variable:

$$f_1(w) = \rho u = \rho u' \cos\theta - \rho v' \sin\theta = f'_1(w') \cos\theta - g'_1(w') \sin\theta. \quad (3.15)$$

Case 2. Flux of a vector component:

$$\begin{aligned} f_2(w) &= \rho u^2 + p = \rho(u' \cos\theta - v' \sin\theta)(u' \cos\theta - v' \sin\theta) + p \\ &= \rho u'^2 \cos^2\theta - 2\rho u'v' \cos\theta \sin\theta + \rho v'^2 \sin^2\theta + p(\cos^2\theta + \sin^2\theta) \\ &= f'_2(w') \cos^2\theta - g'_2(w') \cos\theta \sin\theta - g'_3(w') \cos\theta \sin\theta + f'_3(w') \sin^2\theta. \end{aligned} \quad (3.16)$$

The numerical flux is computed by replacing the local flux by its corresponding local numerical flux. The two cases shown demonstrate that the flux of a scalar variable, such as ρ or e, transforms like a vector or first-order cartesian tensor and the flux of a vector component, such as a momentum component, transforms like a

TABLE I

Angle range	$G'^{+}(U'_{i,j}, U'_{i,j\pm1})$	$G'^{-}(U'_{i+1,j}, U'_{i+1,j\pm1})$	$G'^{+}(U'_{i,j}, U'_{i\pm1,j})$	$G'^{-}(U'_{i,j+1}, U'_{i\pm1,j+1})$
$-\infty < \mathrm{Tan}\,\theta < -\dfrac{\Delta x}{\Delta y}$	$G'(U'_{i,j})$	$G'(U'_{i+1,j})$	$G'(U'_{i-1,j})$	$G'(U'_{i+1,j+1})$
$-\dfrac{\Delta x}{\Delta y} < \mathrm{Tan}\,\theta < 0$	$G'(U'_{i,j-1})$	$G'(U'_{i+1,j+1})$	$G'(U'_{i,j})$	$G'(U'_{i,j+1})$
$0 < \mathrm{Tan}\,\theta < \dfrac{\Delta x}{\Delta y}$	$G'(U'_{i,j+1})$	$G'(U'_{i+1,j-1})$	$G'(U'_{i,j})$	$G'(U'_{i,j+1})$
$\dfrac{\Delta x}{\Delta y} < \mathrm{Tan}\,\theta < \infty$	$G'(U'_{i,j})$	$G'(U'_{i+1,j})$	$G'(U'_{i+1,j})$	$G'(U'_{i-1,j+1})$

second-order cartesian tensor. These transformation laws are well known (cf. Segal [13]), and are easily programmed.

To summarize, if we are given an angle θ, we compute a numerical flux as follows:

(1) Compute velocity components in the local coordinate system.

(2) Compute normal and tangential numerical flux components in the local coordinate system as described above.

(3) Use the cartesian tensor transformation laws to compute numerical flux components of the global variables in the global coordinate system.

We conclude this section with a brief discussion on what it means for a scheme to be conservative and consistent and thus show that the scheme described here possesses these characteristics.

A numerical scheme is called conservative if its numerical flux satisfies a discrete version of the divergence theorem. For a rectangular domain, this means that we must satisfy an identity of the form

$$\sum_{i=1}^{N}\sum_{j=1}^{N}\left[\frac{F_{i+1/2,j}-F_{i-1/2,j}}{\Delta x}+\frac{G_{i,j+1/2}-G_{i,j-1/2}}{\Delta y}\right]\Delta x\,\Delta y$$
$$=\sum_{i=1}^{N}\left[F_{N+1/2,j}-F_{1/2,j}\right]\Delta y+\sum_{i=1}^{N}\left[G_{i,N+1/2}-G_{i,1/2}\right]\Delta x. \tag{3.17}$$

This relation is trivially satisfied for the so-called finite-volume method. Therefore we formulate our scheme as a finite-volume method. That is, we integrate the Euler equations over the rectangular cell

$$x_{i-1/2}<x<x_{i+1/2},\qquad y_{j-1/2}<y<y_{j+1/2} \tag{3.18}$$

where

$$x_{i\pm1/2}=x_i\pm\frac{\Delta x}{2},\qquad y_{j\pm1/2}=y_j\pm\frac{\Delta y}{2}$$

to get

$$\frac{\partial}{\partial t}\int_{y_{j-1/2}}^{y_{j+1/2}}\int_{x_{i-1/2}}^{x_{i+1/2}} w(t,x,y)\,dx\,dy+\int_{y_{j-1/2}}^{y_{j+1/2}} f(w(t,x_{i+1/2},y))\,dy$$
$$-\int_{y_{j-1/2}}^{y_{j+1/2}} f(w(t,x_{i-1/2},y))\,dy+\int_{x_{i-1/2}}^{x_{i+1/2}} g(w(t,x,y_{j+1/2}))\,dx$$
$$-\int_{x_{i-1/2}}^{x_{i+1/2}} g(w(t,x,y_{j-1/2}))\,dx=0. \tag{3.19}$$

In the spirit of Godunov's method, our numerical method is an approximation to

(3.19). That is, the time derivative is approximated by a difference quotient, the numerical solution $W^n_{i,j}$ is interpreted as an approximation to

$$\frac{1}{\Delta x\,\Delta y}\int_{y_{j-1/2}}^{y_{j+1/2}}\int_{x_{i-1/2}}^{x_{i+1/2}} w(t_n, x, y)\,dx\,dy$$

the numerical flux $F^n_{i+1/2,j}$ is an approximation to

$$\frac{1}{\Delta y}\int_{y_{j-1/2}}^{y_{j+1/2}} f(w(t_n, x_{i+1/2}, y))\,dy$$

and the numerical flux $G^n_{i,j+1/2}$ is an approximation to

$$\frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_{i+1/2}} g(w(t_n, x, y_{j+1/2}))\,dx.$$

The numerical fluxes are computed as described above; no time splitting is used.

Since each numerical flux value depends on an angle θ, we associate a value of θ with each cell boundary point. Thus there are four values of θ associated with each cell. The derivation of the tangential local flux G' assumed that θ was constant through a cell. This would indicate that we might prefer to associate the angle θ with cell centers rather than cell boundaries. Unfortunately we could not find a way to do this and maintain conservation.

Lax and Wendroff [7] have shown that a numerical scheme for hyperbolic conservation laws is consistent with the differential equations if the numerical flux reduces to the differential equation flux when all arguments of the numerical flux are set equal. That is,

$$F(u, u,..., u) = f(u). \tag{3.20}$$

The scheme described here is derived so that (3.20) is satisfied by the numerical flux of local variables in local coordinates. Thus (3.20) is satisfied by the global numerical flux because both the global differential equation flux and the global numerical flux are computed from their respective local flux values by the same formulas.

4. Choice of Direction

In Section 3 we described a method designed to resolve shocks by using different numerical flux functions in directions normal and tangential to shocks. In this section we show how to find these directions.

It is important to note that the algorithm we describe does not determine whether or not a shock exists. Thus it is not a shock-fitting algorithm. Instead the algorithm always assumes that on each cell boundary a steady shock exists and computes its

normal direction. Our numerical tests indicate that this approach locates the proper direction when it is needed and causes no problems otherwise.

To determine the direction of a steady oblique shock, we note, as do Gasdynamics texts (cf. Liepmann and Roshko [9]), that a steady oblique shock can be studied as a normal shock with a superimposed uniform tangential velocity. Thus, if we are given two velocity vectors, we can locate a possible shock direction by finding a rotated coordinate frame in which both vectors have a common component.

A simple way to accomplish this was suggested by John Strikwerda. He pointed out that since the velocity in the tangential direction does not change across the shock, the shock must be normal to the velocity jump. That is, to compute the angle at which a shock would cross the cell interface that passes through the point $(x_{i-1/2}, y_j)$, we use the two velocity vectors

$$\mathbf{v}(i-1, j) = [u(i-1, j), v(i-1, j)]$$

and

$$\mathbf{v}(i, j) = [u(i, j), v(i, j)]$$

to compute the normal to the potential shock direction as the direction of the vector.

$$\delta_x \mathbf{v}(i, j) = [u(i, j) - u(i-1, j), v(i, j) - v(i-1, j)] = [\delta_x u, \delta_x v]. \tag{4.1}$$

Thus the angle θ_x used to compute rotated numerical fluxes on this interface is

$$\theta_x = \arctan(\delta_x v / \delta_x u). \tag{4.2}$$

In practice, the velocity components are computed using a first-order finite difference formula. In the smooth parts of the flow the error in the velocity differences can be of the same order as the velocity differences themselves. This can cause the angle θ_x to vary wildly in a part of the flow field where little is happening. Numerical experiments indicate that this degrades the performance of the method.

To prevent this, we replace the velocity differences in Eq. (4.2) by weighted averages of the velocity differences at a number of points. At present we use the averaging

$$\delta_x u(i, j) = [\Delta_x u(i-1, j) + 4\Delta_x u(i, j) + \Delta_x u(i+1, j)]/6 \tag{4.3}$$

where

$$\begin{aligned} \Delta_x u(i, j) = \{&[u(i, j-1) - u(i-1, j-1)] \\ &+ 4[u(i, j) - u(i-1, j)] + [u(i, j+1) - u(i-1, j+1)]\}/6. \end{aligned} \tag{4.4}$$

Similarly, the angle θ_y used to commute rotated numerical fluxes on the cell interface which passes through the point $(x_i, y_{j-1/2})$ is

$$\theta_y = \arctan(\delta_y v / \delta_y u) \tag{4.5}$$

where

$$\delta_y u(i,j) = [\Delta_y u(i,j-1) + 4\Delta_y u(i,j) + \Delta_y u(i,j+1)]/6 \tag{4.6}$$

and

$$\begin{aligned}\Delta_y u(i,j) = \{&[u(i-1,j) - u(i-1,j-1)] \\ &+ 4[u(i,j) - u(i,j-1)] + [u(i+1,j) - u(i+1,j-1)]\}/6.\end{aligned} \tag{4.7}$$

Numerical tests indicate that this procedure locates the proper shock angles and provides smooth angle variations.

The disadvantage of this approach is that it locates steady shocks for the Euler equations but does not locate steady contact discontinuities for the Euler equations or steady discontinuities for other systems of conservation laws. This limitation is also true of the van Leer formulas that we used to compute numerical flux components in the normal direction. In the future we plan to construct a method which can locate these more general discontinuities and to use the method of Roe [12] to resolve them.

One way that we might do this is to note that for the scalar equation

$$u_t + f(u)_x + g(u)_y = 0 \tag{4.8}$$

the normal flux is continuous across a steady discontinuity but the tangential flux is not. Therefore the discontinuity, if it exists, is located in the direction of the vector

$$\delta_x \bar{f} = [f(i,j) - f(i-1,j),\ g(i,j) - g(i-1,j)] = [\delta_x f,\ \delta_x g]. \tag{4.9}$$

For systems we might take linear combinations of vectors like (4.9). One possibility for a system of m equations would be to seek potential discontinuities in the direction of the vector

$$\delta\vec{F} = \left[\sum_{j=1}^{m} \frac{(\delta f_j)(\delta u_j)}{(\delta u_j)^2},\ \sum_{j=1}^{m} \frac{(\delta g_j)(\delta u_j)}{(\delta u_j)^2}\right]. \tag{4.10}$$

Thus far no numerical experiments have been performed using Eq. (4.10).

Baines[1] has proposed another promising approach to the problem of locating general discontinuities. He chooses the angle which satisfies a discrete approximation to the equation

$$\frac{\partial G'}{\partial y'} = -\sin\theta \frac{\partial G'}{\partial x} + \cos\theta \frac{\partial G'}{\partial y} = 0. \tag{4.11}$$

At this time, some details of this procedure need to be worked out and no numerical results are available.

[1] M. J. Baines, 1982, University of Reading, Reading, England, personal communiation.

5. Numerical Results

In this section we present some numerical computations which demonstrate the ability of the present first-order method to resolve steady, oblique shocks. The results are compared with the results of computations using state-of-the-art first- and second-order upwind methods.

Consider the problem of supersonic flow over a wedge which is illustrated in Fig. 3. The solution to this problem is a single oblique shock wave. We consider only cases where the flow is supersonic everywhere.

To solve this problem we construct a uniform computing grid aligned with the wedge as shown in Fig. 4. We specify all variables at the left and top boundaries and we extrapolate all variables at the right boundary. These boundary conditions are correct at the left and right boundaries but the top boundary is overspecified since the normal velocity at this boundary is subsonic. Fortunately, for the cases considered here, no signals reach this boundary from inside the computational domain and this overspecification causes no difficulties.

At the lower boundary we specify that the velocity normal to the wall be zero and use some numerical procedure to specify the remaining variables at the wall. We have experimented with a number of numerical boundary condition procedures and have found that the following gives the best results, at least for the methods and problems considered here.

To compute the pressure and density at the wall, we follow Chakravarthy and Osher [1] and assume that locally, only simple plane waves leave the boundary. In this case the relations

$$c - \frac{(\gamma - 1)}{2} v = \text{const}$$

$$p/\rho^{\gamma} = \text{const} \tag{5.1}$$

$$c = \sqrt{\frac{\gamma p}{\rho}}$$

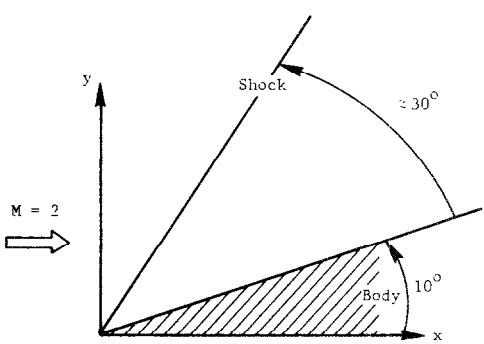

Fig. 3. Oblique shock problem in physical domain.

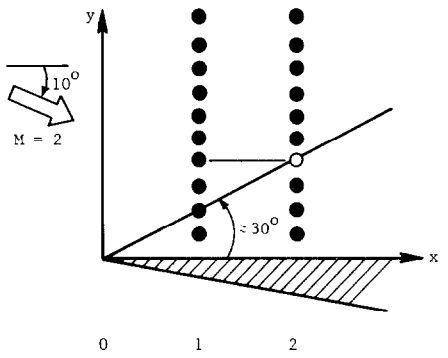

FIG. 4. Oblique shock problem in computational domain.

which are satisfied along the characteristics that reach the wall from the interior, and the wall boundary condition, $v = 0$, determine the pressure and density at the wall.

Various schemes have been used to determine the tangential velocity at the wall. Here we use the condition that, in the steady state, the total enthalpy along the wall should be constant. This gives a relation

$$\frac{e+p}{\rho} = \frac{\gamma}{\gamma-1}\, p/\rho + u^2/2 = \text{const} \tag{5.2}$$

along the wall, which permits us to determine u once p and ρ are known.

Our initial conditions consist of the shock jump of the exact solution oriented at an angle of 45° to the computing grid.

Figure 5a is a three-dimensional plot of the density over a 10° wedge at Mach 2 computed using the first-order method of Osher and Solomon [11]. Figure 5b is a density profile along the line indicated by an arrow in Fig. 5a. The shock angle is approximately 30°. The 30° shock was very difficult for any method to resolve as can be seen from the fact that the shock is spread over many grid points. This is not an acceptable solution.

To be fair we note that the results shown for the Osher–Solomon method are typical of results for first-order upwind schemes. All first-order upwind schemes have difficulty with weak, oblique shocks and the Osher–Solomon method has no more difficulty than any other.

Figures 6a and b show the results of computations of the flow over a 10° wedge at Mach 2 using the second-order upwind method of van Leer described in [14] and [15]. The numerical flux for this method is based on the flux vector splitting formulas, described in Section 2, applied to a second-order approximation to the dependent variables. A slope-limiting function is used to prevent the oscillations which usually occur when shocks are computed using second-order methods. These results show that the scheme is monotonic and that it spreads the shock over approximately five grid points in this case.

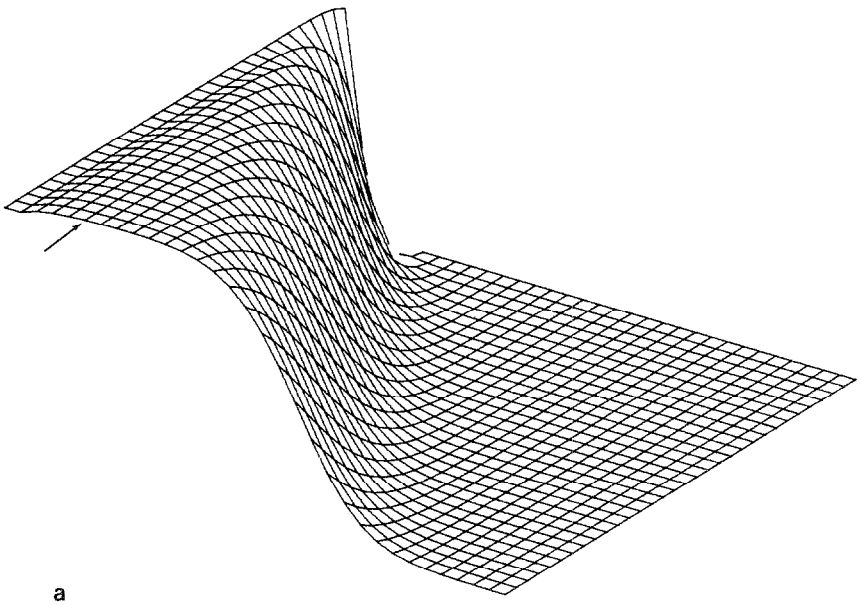

FIG. 5a. Three-dimensional plot of density for oblique shock problem computed using the method of Osher and Solomon.

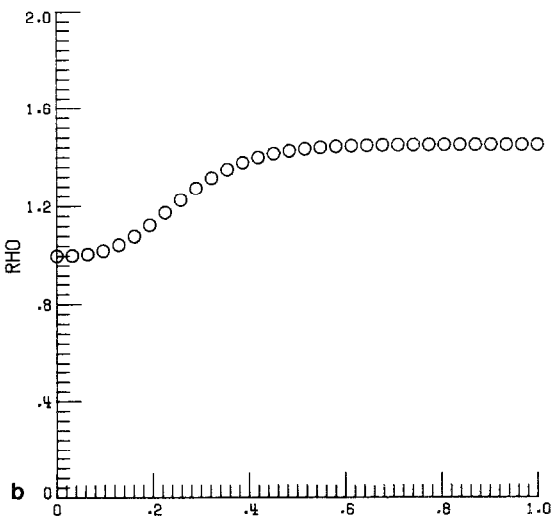

FIG. 5b. Density profile six grid points from the wall for the oblique shock problem computed using the method of Osher and Solomon.

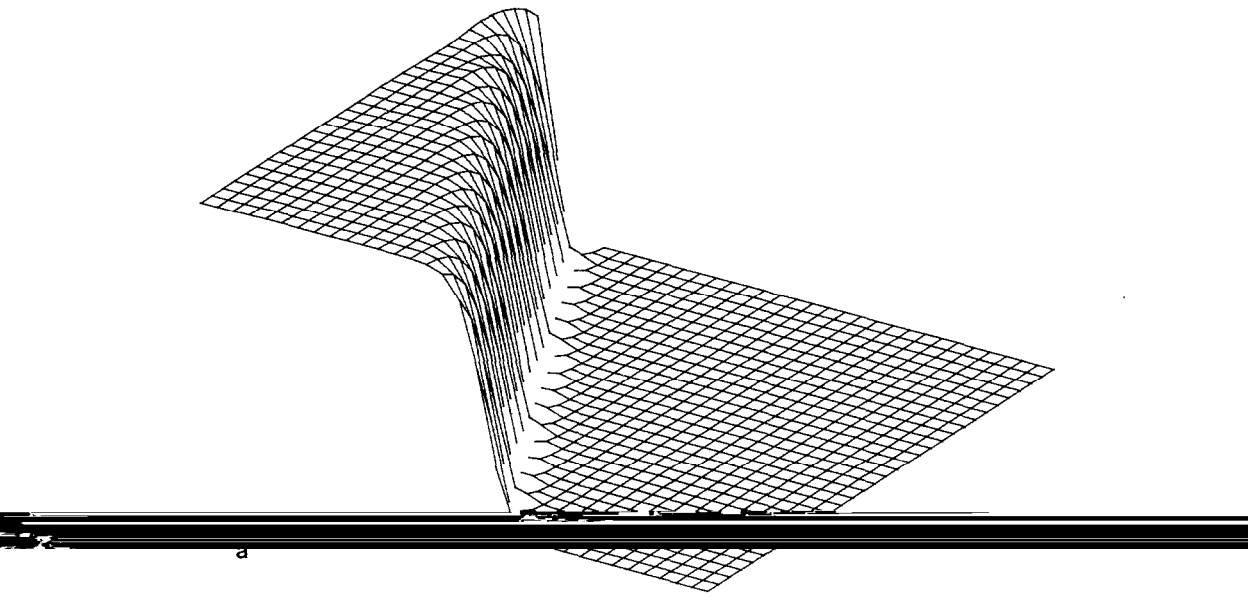

FIG. 6a. Three-dimensional plot of density for oblique shock problem computed using the second-order method of van Leer.

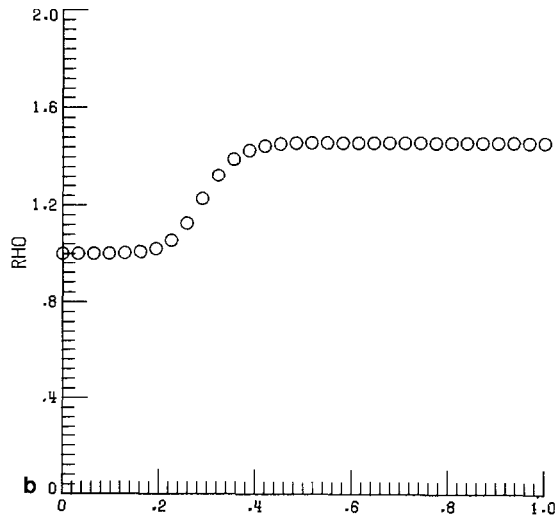

FIG. 6b. Density profile six grid points from the wall for the oblique shock problem computed using the second-order method of van Leer.

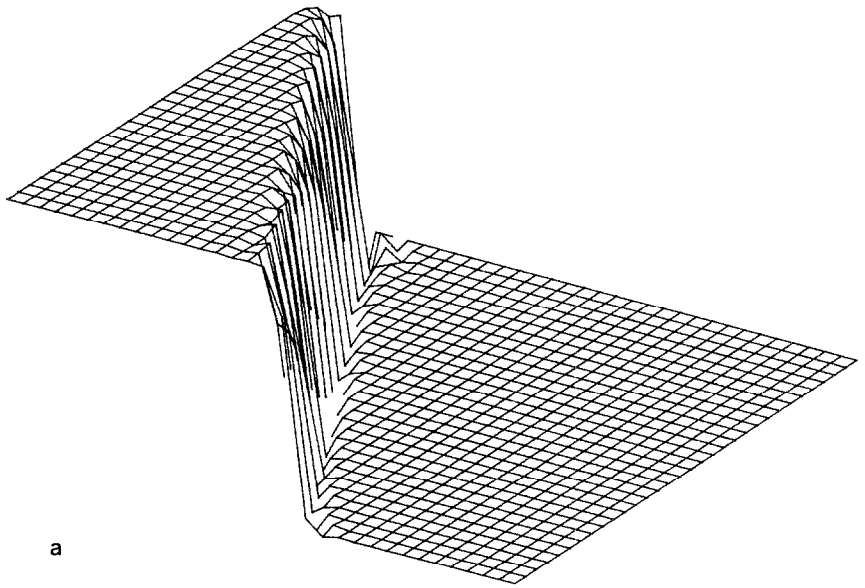

FIG. 7a. Three-dimensional plot of density for oblique shock problem computed using rotationally biased differences with exact shock angle.

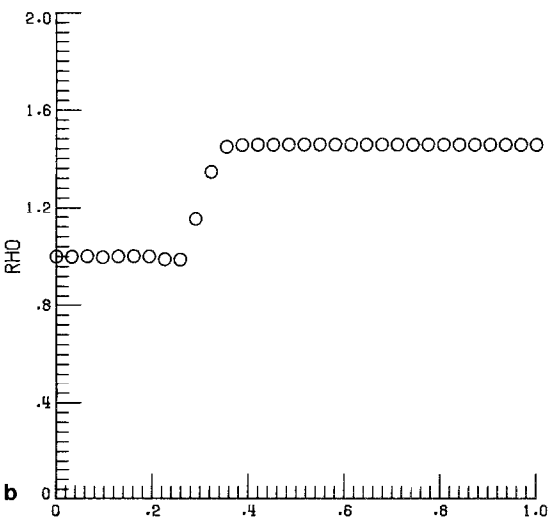

FIG. 7b. Density profile six grid points from the wall computed using rotationally biased differences with exact shock angle.

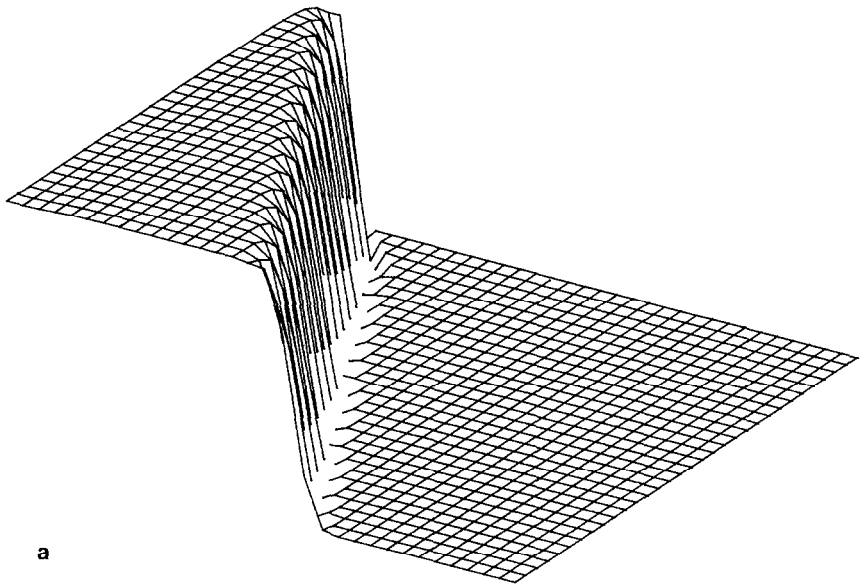

FIG. 8a. Three-dimensional plot of density for oblique shock problem computed using rotationally biased differences with automatic angle algorithm.

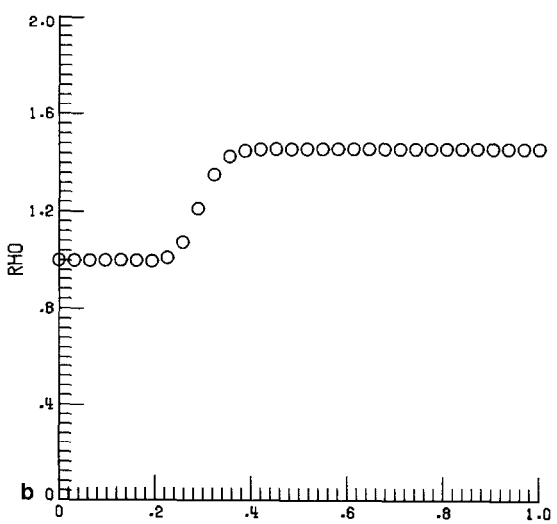

FIG. 8b. Density profile six grid points from the wall computed using rotationally biased differences with automatic angle algorithm.

Figures 7a and b illustrate an ideal situation. These results were computed using the rotated finite difference scheme of Section 3 with the exact shock angle specified and the parameter Q set to zero. In this case the shock is confined to between two and three grid points. In [16] van Leer shows that flux vector splitting requires two grid points to resolve steady one-dimensional shocks. This leads us to believe that the results shown are the best that can be expected for a two-dimensional method based on these numerical flux formulas. These results also indicate that the assumptions used to derive this method are reasonable.

Unfortunately, when we select the shock angle by the algorithm of Section 4, the method is unstable when $Q = 0$. We suspect that this is due to the spatial variations of the computed shock angle since these were not taken into account when the method was derived. We are presently trying to account for this.

At this time we can draw no specific conclusions as to the proper choice for the parameter Q. Extensive numerical experimentation indicates that the choice

$$Q_{i+1/2,j} = \frac{v'_{i,j} + v'_{i+1,j}}{2} \frac{\Delta t}{\Delta x} \tag{5.3}$$

is stable. Figures 8a and b demonstrate the performance of this choice combined with the automatic angle method of Section 4. The shock is confined to approximately three grid points.

The second problem that we study is the regular reflection of a shock from a plane wall. The physical situation is shown in Fig. 9.

The computations shown below were made on a 61×21 uniform grid. Exact values of all variables were specified on the left and top boundaries. All variables were extrapolated at the right boundary and the wall boundary conditions described above were imposed on the lower boundary.

The initial conditions imposed consisted of two shocks which had the same jumps as the exact solution but were located in the wrong place.

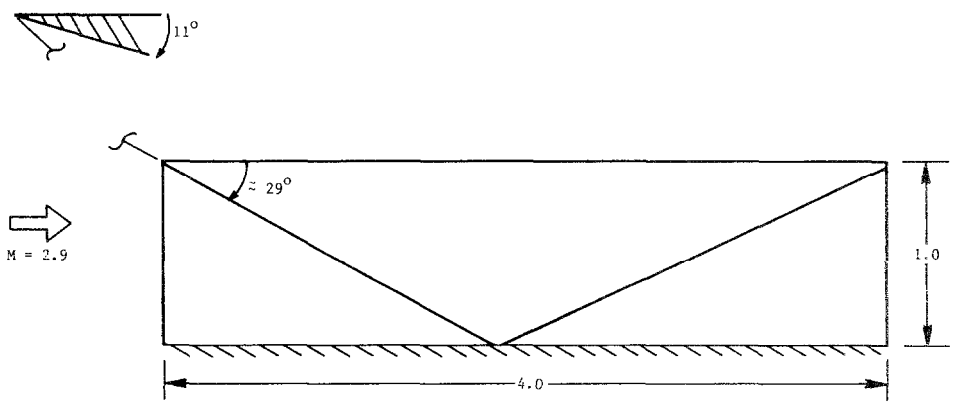

FIG. 9. Shock reflection problem.

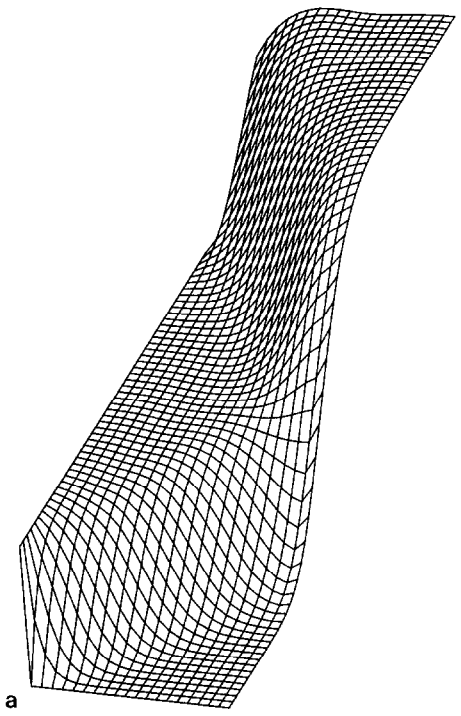

FIG. 10a. Three-dimensional plot of density for shock reflection problem computed using the method of Osher and Solomon.

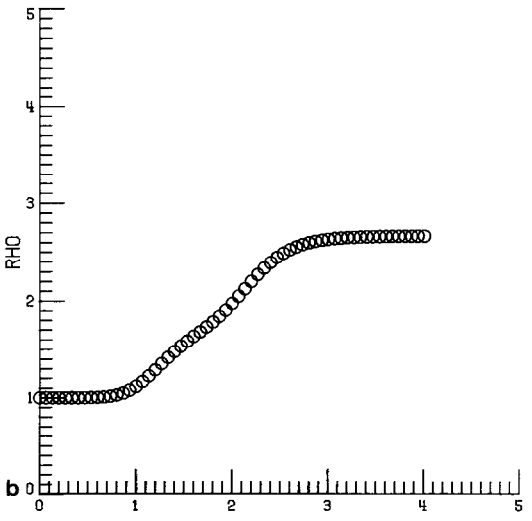

FIG. 10b. Desnity profile at $y = 0.25$ for shock reflection problem computed using the method of Osher and Solomon.

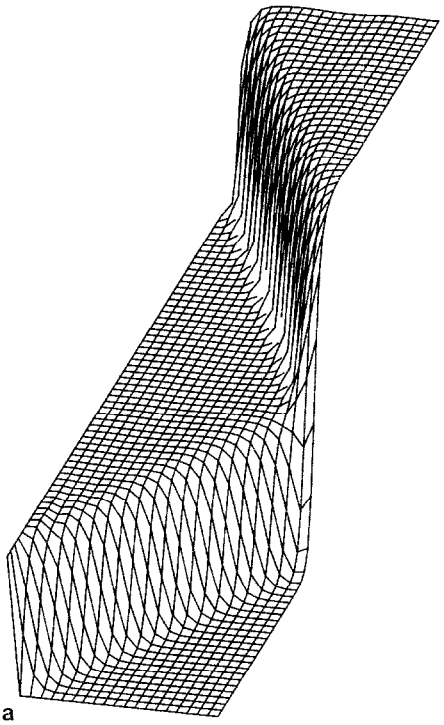

FIG. 11a. Three-dimensional plot of density for shock reflection problem computed using the second-order method of van Leer.

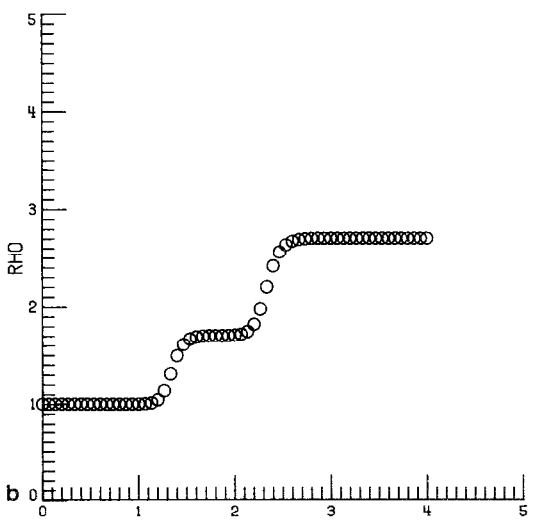

FIG. 11b. Density profile at $y = 0.25$ for shock reflection problem computed using the second-order method of van Leer.

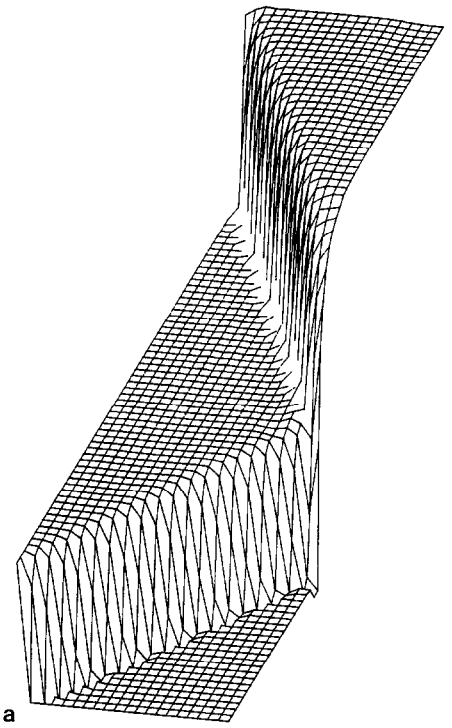

FIG. 12a. Three-dimensional plot of density for shock reflection problem computed using rotationally biased differences with automatic angle algorithm.

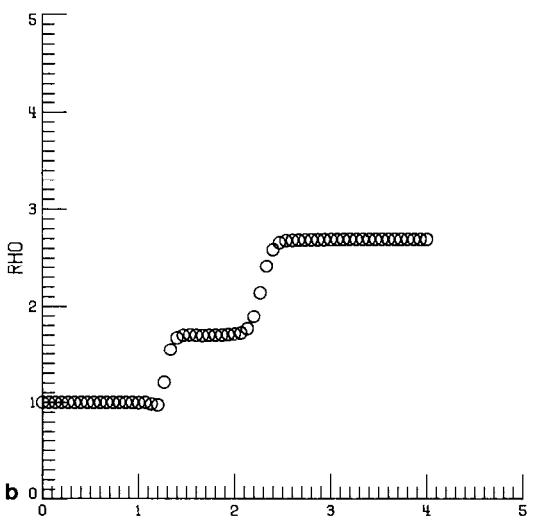

FIG. 12b. Density profile at $y = 0.25$ for shock reflection problem computed using rotationally biased differences with automatic angle algorithm.

Figure 10a is a carpet plot of the density for this flow as computed using the first-order method of Osher and Solomon. Figure 10b is a section of this plot taken at $y = 0.25$. These pictures show that this method smears the shocks so badly that, at $y = 0.25$, they cannot be distinguished.

Figures 11a and b are the corresponding density plots for this flow as computed using the second-order upwind method of van Leer. These results are much better than the previous ones. Here we can distinguish two distinct shocks. The first one is spread over between four and five mesh points and the second one is spread over between six and seven mesh points.

Finally, in Figs. 12a and b, we show density plots for this flow computed using the first-order rotated method. This method spreads the first shock over between two and three mesh points and the second shock over between five and six mesh points. There is a slight undershoot in front of the first shock.

At first sight, these results might appear to be less than dramatic but we should note that we have been comparing a first-order method with one of the more sophisticated of second-order methods. Under these circumstances, we are encouraged by the fact that the first-order method consistently resolved steady oblique shocks within fewer mesh intervals than did the second-order method. This confirms that our oblique shock model works. With additional effort we can make it work better.

6. Summary and Discussion

In this paper we describe a method which determines the orientation of possible shock solutions to the Euler equations. In addition we show how this information can be used to construct a first-order upwind method which computes solutions with greatly improved steady shock resolution. Indeed, we have shown that our first-order method resolves steady shocks within fewer mesh intervals than a sophisticated second-order upwind method.

This work has shown that the shock-resolving ability of numerical methods for hyperbolic equations can be improved if the methods take the orientation of possible shocks into account. Although the results thus far have been encouraging, we wish to improve the present method. Before this can be done some questions need to be answered. In particular, we wish to understand how the variation in angle affects the performance of the method and how we should select the parameter Q in the tangential flux. At present this parameter is selected in an ad hoc manner. It is also important that we construct a careful stability analysis for this method and that we study appropriate choices for boundary conditions. It might also be helpful to study other choices of upwind formulas for the normal flux.

On a more practical level, we realize that first-order methods are usually not accurate enough in regions of smoothly varying flow to be useful in applications. Therefore we are presently developing a second-order-accurate version of our method. This work will be described in a future paper.

ACKNOWLEDGMENTS

I would like to thank Stanley Osher and Bram van Leer for many stimulating discussions on upwind schemes and for their encouragement during this work. I would also like to thank John Strikwerda for providing the simple derivation of Eq. (4.2) which appears in this paper.

REFERENCES

1. S. CHAKRAVARTHY AND S. OSHER, "Numerical Experiments with the Osher Upwind Scheme for the Euler Equations," AIAA Paper 82-0975, AIAA/ASME 3rd Joint Thermophysics, Plasma and Heat Transfer Conference, June, 1982.
2. S. K. GODUNOV, A finite difference method for the numerical computation of discontinuous solutions of the equations of fluid dynamics (in Russian), *Mat. Sb.* **47** (1959); also Cornell Aeronautical Lab (CALSPAN) translation.
3. A. HARTEN AND J. M. HYMAN, "A Self-Adjusting Grid for the Computation of Weak Solutions Of Hyperbolic Conservation Laws," Report LA9105, Center for Nonlinear Studies, Theoretical Division, Los Alamos National Lab, Los Alamos, N. M., 1981.
4. A. HARTEN, P. D. LAX, AND B. VAN LEER, On upstream differencing and Godunov-type schemes for hyperbolic conservation laws, *SIAM Rev.* **25** (1983), 35–62.
5. A. HARTEN, High resolution schemes for hyperbolic conservation laws, *J. Comput. Phys.* **49** (1983), 357–393.
6. A. JAMESON, Iterative solution of transonic flows over airfoils and wings, including flows at Mach 1, *Comm. Pure Appl. Math.* **27** (1974), 283–309.
7. P. LAX AND B. WENDROFF, Systems of conservation laws, *Comm. Pure Appl. Math.* **13** (1960), 217–237.
8. P. D. LAX, "Hyperbolic Systems of Conservation Laws and the Mathematical Theory of Shock Waves," SIAM, Philadelphia, 1972.
9. H. W. LIEPMANN AND A. ROSHKO, "Elements of Gasynamics," Wiley, New York, 1957.
10. E. M. MURMAN, Analysis of embedded shock waves calculated by relaxation methods, *AIAA J.* **12** (1974), 626–633.
11. S. OSHER AND F. SOLOMON, Upwind difference schemes for hyperbolic systems of conservation laws, *Math. Comp.* **38** (1982), 339–374.
12. P. L. ROE, Approximate Riemann solvers, parameter vectors and difference schemes, *J. Comput. Phys.* **43** (1981), 357–372.
13. L. A. SEGAL, "Mathematics Applied to Continuum Mechanics," Macmillan Co., New York, 1977.
14. G. D. VAN ALBADA, B. VAN LEER, AND W. W. ROBERTS, "A Comparative Study of Computational Methods in Cosmic Gas Dynamics," ICASE Report 81-24, 1981.
15. B. VAN LEER, "On the Relation Between the Upwind-Differencing Schemes of Godunov, Enquist–Osher and Roe," ICASE Report 81-11, 1981.
16. B. VAN LEER, "Flux Vector Splitting for the Euler Equations," ICASE Report 82-30, 1982.
17. P. WOODWARD AND P. COLELLA, *in* "Proceedings, 7th International Conference on Numerical Methods in Fluid Dynamics," Stanford/NASA Ames, 1980.
18. N. N. YANENKO, "The Method of Fractional Steps," Springer-Verlag, New York/Berlin, 1971.
19. H. C. YEE, R. F. WARNING, AND A. HARTEN, "On the Application and Extension of Harten's High Resolution Scheme," NASA TM 84256, 1982.