JOURNAL OF COMPUTATIONAL PHYSICS 53, 359–381 (1984)

The Turn Function and Vorticity Method for Numerical Fluid Dynamics*

PETER J. O'ROURKE

Los Alamos National Laboratory, Los Alamos, New Mexico 87545

Received June 2, 1982; revised August 5, 1983

A numerical method is presented that solves in a consistent fashion, conservation equations for both vorticity and linear momentum in multidimensional fluid-dynamics calculations. The equations are given in both two- and three-dimensional Cartesian geometry, and it is shown how the method can be easily implemented in a two-dimensional Eulerian fluid-dynamics code. The results of example calculations, which were performed with and without the new method, show the large errors that can arise when the vorticity equation is not solved in compressible flow calculations.

INTRODUCTION

This paper presents a finite-difference method for multidimensional numerical fluid-dynamics calculations, which solves conservation equations for both linear momentum and vorticity. The method, called the "Turn function and vorticity" (TFV) method, improves the accuracy of previous techniques for calculating both low and high Mach number flows in which the divergence of the velocity field is nonzero. It is an easy modification to many existing hydrodynamics codes that conserve linear momentum, and the resulting increase in computational time is about twenty-five percent in the two-dimensional calculations we have performed.

The method is implemented in the following manner. In addition to solving the linear momentum equations, one solves the equation(s) for the component(s) of the vorticity. Without any other modification, the result of solving the linear-momentum equations using finite differences, is a velocity field that is not, in general, consistent with the computed vorticity field. To achieve this consistency, we add an anti-symmetric part τ' to the stress tensor that is used in the linear-momentum equations. In two space dimensions, this artificial stress has a single scalar component, which we call the "Turn function." This artificial stress contrasts with the true stress tensor of most fluids, which is symmetric [1]. A derivation similar to that of the usual Helmholtz vorticity equation shows that an anti-symmetric stress can produce

* The U.S. Government's right to retain a nonexclusive royalty-free license in and to the copyright covering this paper, for governmental purposes, is acknowledged. Work performed under the auspices of the U.S. Department of Energy.

0021-9991/84 $3.00

vorticity. In the present circumstances, this production of vorticity is chosen so as to make up for vorticity destruction that otherwise results from the finite-difference approximations to the linear-momentum equations. At the same time, linear momentum is conserved because the anti-symmetric stress is differenced conservatively and vanishes on boundaries. As we will show later, our anti-symmetric stress goes to zero in the limit of vanishingly small computational cell size and time step. It is present to cancel numerical errors that result in fictitious vorticity production or destruction.

When shocks are present in the flow field, the above procedure must be changed to ensure that the correct jump conditions are obtained across the shocks. In the narrow region of the shock, we revert to a conventional formulation in which the conservative linear momentum equations are solved by conservative finite-difference approximations. As is well known [2], such a formulation gives the proper jump conditions across shocks, and, therefore, the proper amount of vorticity production in shocks. Later in this paper, we elaborate the necessity for this special treatment of shocks and tell how we switch between the numerical procedures used inside and outside shocks.

Solving for $\boldsymbol{\tau}'$ is often an elliptic problem. Because we want to prescribe physical wall-boundary conditions, the value of $\boldsymbol{\tau}'$ must not interfere with these physical constraints; for this reason the boundary conditions on the components of $\boldsymbol{\tau}'$ are generally that they vanish on computational boundaries. Elliptic equations must usually be solved by time-consuming iterative methods, but this is mitigated in the present case by the fact that Dirichlet boundary conditions are used. It has been our experience and that of others [2], that convergence is much more rapid when one has Dirichlet, and not Neumann, boundary conditions.

To our knowledge, ours is the first method that solves conservation equations for both linear momentum and vorticity. Traditional methods for solving for flows with solenoidal velocity fields ($\nabla \cdot \mathbf{u} = 0$) have used either the vorticity equation(s) [3] or the linear-momentum equations [4], but not both. When $\nabla \cdot \mathbf{u} = 0$, the TFV method gives the same result as the stream function and vorticity method [3], assuming the same finite-difference equations are used for the vorticity. To see this, we note that there is a unique discrete velocity field that satisfies the finite-difference approximations to $\nabla \cdot \mathbf{u} = 0$ and $\nabla \times \mathbf{u} = \boldsymbol{\omega}$ (vorticity) together with associated boundary conditions. (We do not exactly obtain this unique velocity field in either method because a finite convergence tolerance must be used in solving the finite-difference equations). Since the stream function and vorticity method does not use the linear-momentum equations, it will be faster than TFV, although no loss of accuracy is incurred by using the TFV method when $\nabla \cdot \mathbf{u} = 0$. An interesting corollary to the above argument is that the stream function and vorticity method conserves linear momentum.

When the velocity field is not solenoidal, the TFV method can significantly improve numerical accuracy. In this case, there is no stream function, and past studies have solved just the linear-momentum equations. It has been postulated [5] that in some applications, vortex compression could give rise to intense slowly damped vortices and that current numerical methods could not calculate these due to

large numerical damping. The validity of this criticism will be demonstrated later in this paper.

We give the equations that are approximated in the TFV method—first for shock-free flows and then for flows with shocks. Next, we present four computational examples. The finite-difference approximations are described, and it is shown that solving the vorticity equation, significantly improves the accuracy of the calculations.

The Equations for Shock-free Flows

We present the equations in Cartesian-tensor notation. In contrast to traditional formulations, equations are kept for both linear momentum and vorticity, and the linear momentum equations include terms arising from the anti-symmetric stress τ'_{ij}. The stress τ'_{ij} is taken to be proportional to the difference between the vorticity and the curl of the velocity field.

The continuity equation is

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_i}{\partial x_i} = 0. \tag{1}$$

Before writing the momentum equation, we express the anti-symmetric stress τ'_{ij} as the dual of a vector t'_k,

$$\tau'_{ij} = \varepsilon_{ijk} t'_k,$$

where ε_{ijk} is the alternating tensor [1]. t'_k will be called the Turn vector. The momentum equation becomes

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial \rho u_i u_j}{\partial x_j} + \frac{\partial p}{\partial x_i} = \frac{\partial \sigma_{ij}}{\partial x_j} + \varepsilon_{ijk} \frac{\partial t'_k}{\partial x_j}, \tag{2}$$

where σ_{ij} is the usual symmetric Newtonian stress tensor,

$$\sigma_{ij} = \mu \left[\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right] + \lambda \frac{\partial u_k}{\partial x_k} \delta_{ij}.$$

The vorticity equation is

$$\frac{\partial w_i}{\partial t} + \frac{\partial w_i u_j}{\partial x_j} + \varepsilon_{ijk} \frac{\partial}{\partial x_j} \left(\frac{1}{\rho} \frac{\partial p}{\partial x_k} \right) = \frac{\partial u_i w_j}{\partial x_j} + \varepsilon_{ijk} \frac{\partial}{\partial x_j} \left(\frac{1}{\rho} \frac{\partial \sigma_{kl}}{\partial x_l} \right). \tag{3}$$

Here we have written the vortex stretching term $\partial w_j u_i / \partial x_j$ in its conservative form. We propose the simple constitutive relation for the Turn vector,

$$t'_i = \mu_t \left(w_i - \varepsilon_{ijk} \frac{\partial u_k}{\partial x_j} \right). \tag{4}$$

The value of the proportionality constant μ_t, which has the dimensions of a viscosity, will be discussed shortly.

Equations (1)–(4), together with an energy equation, are to be approximated by finite differences. The precise form of the energy equation does not concern us in this paper, but we do note that the Turn vector t'_k should not appear in it. The reason for this is that the Turn vector is used in (2) to cancel certain truncation errors that arise when we approximate (2) by finite differences, and that otherwise cause a fictitious numerical stress. Thus, no net stress results in the linear-momentum equation or should appear in the energy equation.

Due to truncation errors, when they are approximated by finite differences, Eqs. (1)–(3) are not solved exactly. The equations that are solved (at a finite number of grid points) can be obtained by expanding the terms in the finite difference equations in Taylor series in δx_i and δt, the computational cell size and time step [6]. The result of this expansion will be modified forms of Eqs. (1)–(3),

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_i}{\partial x_i} = R, \tag{1'}$$

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial \rho u_i u_j}{\partial x_j} + \frac{\partial p}{\partial x_i} = \frac{\partial \sigma_{ij}}{\partial x_j} + \varepsilon_{ijk} \frac{\partial t'_k}{\partial x_j} + S_i, \tag{2'}$$

and

$$\frac{\partial w_i}{\partial t} + \frac{\partial w_i u_j}{\partial x_j} + \varepsilon_{ijk} \frac{\partial}{\partial x_j}\left(\frac{1}{\rho}\frac{\partial p}{\partial x_k}\right) = \frac{\partial w_j u_i}{\partial x_j} + \varepsilon_{ijk} \frac{\partial}{\partial x_j}\left(\frac{1}{\rho}\frac{\partial \sigma_{kl}}{\partial x_l}\right) + T_i. \tag{3'}$$

R, S_i, T_i are sums of truncation error terms that are proportional to positive powers of δx_i and δt, and that therefore become vanishingly small when δx_i and δt approach zero.

From Eqs. (1')–(3') we can derive a transport equation for the discrepancy D_i between the vorticity field and the curl of the velocity field; that is,

$$D_i = w_i - \varepsilon_{ijk} \frac{\partial u_k}{\partial x_j}. \tag{5}$$

Taking the time derivative of (5), substituting from (1') to (3'), and using $t'_i = \mu_t D_i$ gives

$$\frac{\partial D_i}{\partial t} = -\frac{\partial D_i u_j}{\partial x_j} + \frac{\partial u_i D_j}{\partial x_j} + \frac{\partial}{\partial x_j}\left[\frac{1}{\rho}\left(\frac{\partial(\mu_t D_i)}{\partial x_j} - \frac{\partial(\mu_t D_j)}{\partial x_i}\right)\right] + T_i - \varepsilon_{ijk}\frac{\partial}{\partial x_j}\left(\frac{S_k - R u_k}{\rho}\right). \tag{6}$$

Thus the discrepancy is convected and "stretched," like the vorticity. It is produced due to truncation errors and is diffused with diffusion coefficient μ_t/ρ.

An important conclusion can be drawn from Eq. (6), and this is that the Turn vector terms vanish in the limit of vanishingly small δx_i and δt. Thus we are solving the correct equations for a compressible fluid. To see this let us assume, for example, that we are using a second order accurate numerical method for all equations. Then reducing δx_i and δt by a factor of n will result in the truncation errors T_i, S_i, and R being reduced approximately by a factor of n^2. Since D_i vanishes on computational boundaries, it is clear from a scaling argument applied to (6), that D_i, and hence t_i', will also be reduced by a factor of n^2 for the refined mesh. This same argument shows that including the Turn vector terms does not degrade the order of accuracy of the numerical solution if the order of accuracy of the finite-difference approximation to (3) is not lower than those to Eqs. (1) and (2).

We now discuss possible choices for the viscosity μ_t. One choice is to take μ_t very large. When μ_t approaches infinity, Eq. (4) is replaced by

$$w_i = \varepsilon_{ijk} \frac{\partial u_k}{\partial x_j}. \tag{7}$$

Equation (7) is then a constraint that determines the Turn vector. From (6), we see this is equivalent to solving the elliptic Poisson equation for t_i',

$$\frac{\partial}{\partial x_j}\left[\frac{1}{\rho}\left(\frac{\partial t_i'}{\partial x_j} - \frac{\partial t_j'}{\partial x_i}\right)\right] = \varepsilon_{ijk}\frac{\partial}{\partial x_j}\left(\frac{S_k - Ru_k}{\rho}\right) - T_i. \tag{8}$$

The advantage of taking μ_t large is that strict equality is maintained between the vorticity field and the curl of the velocity field. One disadvantage is that we must always solve an elliptic equation for t_i', which is computationally time-consuming.

There is another apparent disadvantage of the large μ_t formulation, and this can be seen from Eq. (6). Any discrepancy D_i that is produced, is instantaneously diffused to the boundaries of the computational region, and thereby can alter the flow upstream of the region of discrepancy production, even in a supersonic flow. One manifestation of this is that we cannot specify all the boundary conditions we would like at an inflow boundary where there is inviscid supersonic flow. Since (8) is elliptic, we can specify t_i' or the normal derivative of t_i' but not both, at an inflow boundary. Suppose we specify t_i'. Then the derivatives of t_i' normal to the inflow boundary are obtained as part of the calculation. Since these normal derivatives appear on the right-hand sides of the equations for the tangential velocity components [cf. Eq. (2)], not all the quantities on the left-hand sides of these equations can be specified at an inflow boundary. Thus at an inflow boundary we cannot specify all three velocity and vorticity components and all thermodynamic properties. A similar situation is encountered if the normal derivatives of t_i' are specified.

The magnitudes of the upstream disturbances can be significantly reduced if we choose a value for μ_t based on the criterion that diffusion wave velocities be less than or equal to the local adiabatic sound speed c. Since a diffusion wave of length L propagates with approximate speed $\mu_t/\rho L$, this criterion gives

$$\mu_t \lesssim \rho c L,$$

where L is a characteristic length associated with the problem. Since we also want as large a value of μ_t as possible to reduce the size of the discrepancy D_i, we take

$$\mu_t = \rho c L. \tag{9}$$

The advantage of this finite value of μ_t is that waves of length greater than or equal to L have subsonic diffusion velocities. One disadvantage is that again the Turn vector terms must usually be differenced implicitly. Another disadvantage is that we are allowing a nonzero discrepancy between the vorticity and the curl of the velocity field, although this discrepancy does vanish in the low Mach number limit. To see this, we note that the Reynolds number based on μ_t and L, is simply the characteristic Mach number of the flow. Hence, for low Mach numbers, the convective terms in (6) are negligible, and the infinite μ_t formulation is recovered. Both the infinite and finite values of μ_t have advantages and disadvantages, and will be further evaluated in the computational section of this paper.

The Equations for Flows with Shocks

When shocks are present in the flow field in high Reynold's number problems, numerical solutions of the equations of the preceding section can have the incorrect jump conditions across the shocks. In this section, we tell how this error arises and give a method that ensures that the correct shock jump conditions are obtained. This method is to return in the neighborhood of shocks to a traditional conservative formulation to the equations without Turn vector terms.

To see why solutions to the equations of the preceding section can have the incorrect jump conditions, let us examine the momentum conservation equation (2). From it we see that the momentum flux per unit area in the direction of the unit normal n_j to a surface S is

$$P_i = \rho u_i (u_j - v_j) n_j + p n_i + \varepsilon_{ijk} t'_j n_k.$$

Here v_j is the surface velocity. By considering the total flux of momentum into a thin, pancake-shaped volume whose flat faces lie on either side of the shock, we can in the usual manner derive the shock conservation condition for momentum

$$\rho^+ u_i^+ (u_j^+ - v_j) n_j^+ + p^+ n_i^+ + \varepsilon_{ijk} (t'_j)^+ n_k^+ - [\rho^- u_i^- (u_j^- - v_j) n_j^+ + p^- n_i^+ + \varepsilon_{ijk} (t'_j)^- n_k^+] + \iint_{\mathscr{L}} P_i \, da = 0. \tag{10}$$

Here n_j^+ is the unit normal pointing to the "+" side of the shock, and the integral is over the lateral, thin surface of the pancake. This integral could be dropped in the limit of a very thin pancake if the integrand were bounded, but in our case this cannot be done. Due to truncation errors, the Turn vector can be very large in

shocks. We usually like to keep a fixed number of computational cells in the shock, and hence, as the mesh is refined, the truncation errors in the shock can grow without bound. Even if this integral is ignored, if we take the scalar product of (10) with any unit vector $\mathbf{l}_i$ in the plane of the shock, we see that

$$u_i^+ l_i \rho^+ (u_j^+ - v_j) n_j^+ + \varepsilon_{ijk} (t_j')^+ n_k^+ l_i = u_i^- l_i \rho^- (u_j^- - v_j) n_j^+ + \varepsilon_{ijk} (t_j')^- n_k^+ l_i.$$

Since

$$m = \rho^+ (u_j^+ - v_j) n_j^+ = \rho^- (u_j^- - v_j) n_j^+$$

is the mass flow through the shock, we have

$$(u_i^+ - u_i^-) l_i = \frac{\varepsilon_{ijk}}{m} n_k^+ l_i ((t_j')^- - (t_j')^+). \tag{11}$$

From (11) we see that in order to have no jump in tangential velocity components across a shock, we need to have no jump in the tangential components of the Turn vector. This will probably not be true due to the large truncation errors in the neighborhood of computational shocks.

To remedy this situation, in shocks we revert to a traditional finite difference formulation of the compressible flow equations. The Turn vector is set to zero, and the vorticity is computed from its defining equation in terms of velocity. Our treatment of shocks is depicted schematically in Fig. 1. The shock separates regions II and III, where the Turn vector is zero. In regions I and IV the vorticity equation is solved. Switch surfaces separate neighboring regions where different algorithms are used. By equating the momentum fluxes across the three interfaces in Fig. 1, one now obtains for the momentum conservation condition across the shock,

$$m u_i^{\mathrm{I}} + p^{\mathrm{I}} n_i^{\mathrm{I}} + \varepsilon_{ijk} (t_j')^{\mathrm{I}} n_k^{\mathrm{I}} = m u_i^{\mathrm{IV}} + p^{\mathrm{IV}} n_i^{\mathrm{I}} + \varepsilon_{ijk} (t_j')^{\mathrm{IV}} n_k^{\mathrm{I}}. \tag{12}$$

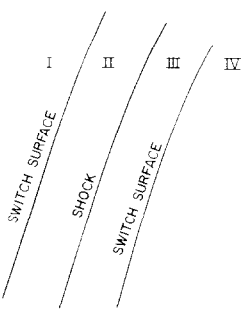

FIG. 1. Schematic depiction of our numerical treatment of shocks. The vorticity equation is solved in regions I and IV and not solved in regions II and III.

Now since the values of the Turn vector in regions I and IV vanish in the limit of vanishingly small computational cell size, the correct jump conditions are obtained.

It remains to tell how we determine whether or not we are in the neighborhood of a shock. To do this we compute a switching parameter β from the formula

$$\beta = \frac{L}{u_0} \frac{\partial u_k}{\partial x_k}, \tag{13}$$

where L and u_0 are characteristic length and velocity scale of the problem. If $\beta < -1$, we are in a shock, and the Turn vector is set to zero. If $\beta \geqslant -1$, the vorticity equation is used.

Numerical Considerations

In this section we describe the numerical techniques that we use to solve the equations of the TFV method. Emphasis is placed on the finite-difference equations used to obtain w and t'. We also discuss the numerical treatment of the computational boundaries that are used in the calculations of the next section.

In the next section we describe solutions to the equations of the TFV method in planar coordinates and in cylindrical coordinates assuming axial symmetry and no-swirl velocity. For future reference we give these equations here for high Reynolds number flows. In planar Cartesian coordinates with x and y as independent variables and u and v the associated velocity components, the equations are

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} = 0, \tag{14}$$

$$\frac{\partial \rho u}{\partial t} + \frac{\partial \rho u^2}{\partial x} + \frac{\partial \rho uv}{\partial y} + \frac{\partial p}{\partial x} = \frac{\partial t'}{\partial y}, \tag{15}$$

$$\frac{\partial \rho v}{\partial t} + \frac{\partial \rho uv}{\partial x} + \frac{\partial \rho v^2}{\partial y} + \frac{\partial p}{\partial y} = -\frac{\partial t'}{\partial x}, \tag{16}$$

$$\frac{\partial w}{\partial t} + \frac{\partial wu}{\partial x} + \frac{\partial wv}{\partial y} + \frac{\partial}{\partial y}\frac{1}{\rho}\frac{\partial p}{\partial x} - \frac{\partial}{\partial x}\frac{1}{\rho}\frac{\partial p}{\partial y} = 0, \tag{17}$$

and

$$t' = \mu_t \left[w - \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \right) \right]. \tag{18}$$

In cylindrical coordinates, with z and r as independent variables, and u and v the associated velocity components, the equations are

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u}{\partial z} + \frac{1}{r}\frac{\partial \rho v r}{\partial r} = 0, \tag{19}$$

$$\frac{\partial \rho u}{\partial t} + \frac{\partial \rho u^2}{\partial z} + \frac{1}{r}\frac{\partial \rho u v r}{\partial r} + \frac{\partial p}{\partial z} = \frac{1}{r}\frac{\partial t' r}{\partial r}, \tag{20}$$

$$\frac{\partial \rho v}{\partial t} + \frac{\partial \rho u v}{\partial z} + \frac{1}{r}\frac{\partial \rho v^2 r}{\partial r} + \frac{\partial p}{\partial r} = -\frac{\partial t'}{\partial z}, \tag{21}$$

$$\frac{\partial w}{\partial t} + \frac{\partial w u}{\partial z} + \frac{\partial w v}{\partial r} + \frac{\partial}{\partial z}\left(\frac{1}{\rho}\frac{\partial p}{\partial r}\right) - \frac{\partial}{\partial r}\left(\frac{1}{\rho}\frac{\partial p}{\partial z}\right) = 0, \tag{22}$$

$$t' = \mu_t \left[w - \left(\frac{\partial v}{\partial z} - \frac{\partial u}{\partial r} \right) \right]. \tag{23}$$

It can be seen that (22) conserves w/r. These equations are, of course, to be used outside of shocks. In shocks, the vorticity equation is not solved, and we set $t' \equiv 0$. A form of upwind differencing, which we will describe shortly, was used to stabilize the calculations and to provide dissipation in shocks.

All the calculations were performed with a modified version of the LDEF computer program [7]. LDEF solves conservative mass and momentum equations using the ICE technique [8], and a thermal enthalpy equation. This energy equation does not conserve total energy, and thus, in the problems of the next section where energy conservation is important, we calculated an isothermal flow so that errors due to lack of energy conservation did not obscure our validation of the TFV method.

To modify LDEF to incorporate the TFV method, we simply added an additional subroutine that was called at the end of each computational cycle. The subroutine updated the values of w and t' by one computational time step δt and calculated the changes in velocity due to t'. The locations of the computational cell (i, j) variables are shown in Fig. 2. The staggered-mesh system of the ICE method was used and the

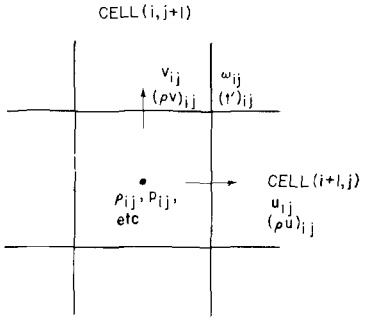

FIG. 2. LDEF computational cell (i, j) and the locations of the variables with subscripts (i, j).

vorticity w_{ij} and Turn function t'_{ij} (the subscripts being cell indices) were naturally located at the upper-right corner of the cell, where the difference approximation,

$$(t'_{ij})^{n+1} = \mu_{t_{ij}}^{n+1} \left[w_{ij}^{n+1} - \frac{v_{i+1,j}^{n+1} - v_{i,j}^{n+1}}{\delta x} + \frac{u_{ij+1}^{n+1} - u_{ij}^{n+1}}{\delta y} \right], \tag{24}$$

had second-order accuracy in space. The superscript n denotes that the quantity is evaluated at time $t = n\delta t$.

The convective terms in the vorticity equation were approximated by taking the average of the results obtained using Leith's method and pure upwind or donor-cell differencing [2]. In problems where the flow was isothermal, the pressure gradient terms were neglected. Otherwise, we differenced $(\partial/\partial y)(1/\rho)(\partial p/\partial x)$ by

$$\frac{2}{\delta x\, \delta y} \left[\frac{P_{i+1\,j+1}^{n+1} - P_{ij+1}^{n+1}}{\rho_{i+1\,j+1}^{n+1} + \rho_{ij+1}^{n+1}} - \frac{P_{i+1j}^{n+1} - P_{ij}^{n+1}}{\rho_{i+1j}^{n+1} + \rho_{ij}^{n+1}} \right],$$

and $(\partial/\partial x)(1/\rho)(\partial p/\partial y)$ similarly. Advaced-time pressures and densities were known and used at that point in the computational cycle when the vorticity equation was solved.

After solving for w_{ij}^{n+1}, the values of t_{ij}^{n+1} were found by simultaneously solving the following equations, and Eq. (24) by Gauss–Seidel iteration,

$$(\rho u)_{i,j}^{n+1} = (\widetilde{\rho u})_{i,j} + \delta t \frac{(t')_{i,j}^{n+1} - (t')_{i,j-1}^{n+1}}{\delta y}, \tag{25}$$

$$(\rho v)_{i,j}^{n+1} = (\widetilde{\rho v})_{i,j} - \delta t \frac{(t')_{i,j}^{n+1} - (t')_{i-1,j}^{n+1}}{\delta x}, \tag{26}$$

$$v_{i,j}^{n+1} = \frac{2(\rho v)_{i,j}^{n+1}}{\rho_{i,j}^{n+1} + \rho_{i,j+1}^{n+1}}, \tag{27}$$

and

$$u_{i,j}^{n+1} = \frac{2(\rho u)_{i,j}^{n+1}}{\rho_{i,j}^{n+1} + \rho_{i+1j}^{n+1}}. \tag{28}$$

The quantities $(\widetilde{\rho u})_{i,j}$ and $(\widetilde{\rho v})_{i,j}$ were partially updated values of the mass fluxes that included changes due to all terms except the Turn function terms. In shocks, the vorticity equation and Eq. (24) were replaced with

$$(t')_{ij}^{n+1} = 0 \tag{29}$$

and

$$w_{ij}^{n+1} = \frac{v_{i+1j}^{n+1} - v_{ij}^{n+1}}{\delta x} - \frac{u_{ij+1}^{n+1} - u_{ij}^{n+1}}{\delta y}, \tag{30}$$

and (25)–(28) were retained.

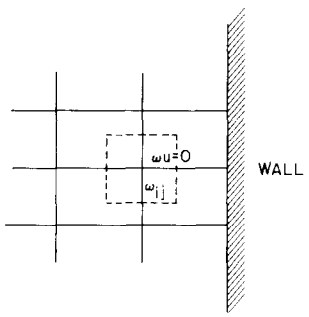

FIG. 3. The vorticity flux is set to zero across boundaries adjacent to rigid walls.

The mass equation was uncoupled from this iteration. This is because the finite-difference approximation to $(\partial \rho u/\partial x) + (\partial \rho v/\partial y)$, which was

$$\frac{(\rho u)_{i,j}^{n+1} - (\rho u)_{i-1j}^{n+1}}{\delta x} + \frac{(\rho v)_{i,j}^{n+1} - (\rho v)_{i,j-1}^{n+1}}{\delta y},$$

does not change when $(\rho u)^{n+1}$ and $(\rho v)^{n+1}$ are replaced by $\widetilde{\rho u}$ and $\widetilde{\rho v}$. Thus the finite-difference equations reflect the fact that the Turn vector does not appear in the transport equation for $\nabla \cdot (\rho \mathbf{u})$.

In the example calculations, three types of computational boundaries were encountered: rigid-walls, inflow boundaries, and outflow boundaries. We next discuss how these boundaries were treated when computing the Turn function and vorticity.

We treated the Turn function and vorticity at cell vertices on walls the same as for cell vertices in shocks; that is, the Turn function was set to zero. This ensured that there was no loss of the tangential linear momentum component at the wall. When a vorticity cell was a half-cell size away from a wall, as is depicted in Fig. 3, we set to zero the convective flux of vorticity across the boundary that lay closest to the wall. Although it was usually not needed the vorticity associated with the half-cell next to the wall was calculated by differencing $w = \pm(\partial v_t/\partial \mathbf{n})$ (whichever sign applied), using the computed value of v_t closest to the wall and $v_t = 0$ at the wall. Here v_t is the tangential velocity component to the wall and $\mathbf{n}$ is the normal direction to the wall. This much vorticity is confined to a very thin boundary layer next to the wall. This wall vorticity value was only used if flow separation caused the wall vorticity layer to enter the interior flow.

At inflow boundaries, we specified the normal velocity v_n to be uniform and the Turn function to be zero. Then, as we have discussed previously, with the TFV mthod we cannot specify both the tangential velocity component and the vorticity at an inflow boundary, even if it is a supersonic inflow boundary. In the calculations of this paper we have specified $w = 0$ at the inflow boundary. Then we have taken $(\partial v_t/\partial \mathbf{n}) = 0$, as is required by the condition that the Turn function be zero.

We now describe our outflow boundary treatment. Figure 4 shows a portion of such an outflow boundary located at the right hand side of the computational mesh

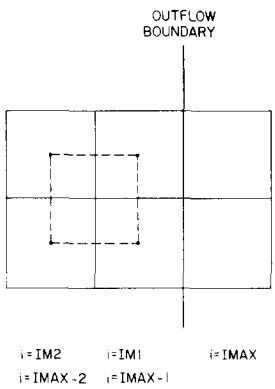

FIG. 4. Vorticity cell located next to an outflow boundary.

Donor-cell differencing is used for the convective terms at the right-hand edge of those cells located next to the boundary. In particular, donor-cell differencing was used on the right hand edge of the vorticity cell outlined by dashes in Fig. 4. The normal velocity is computed at the outflow boundary, and for this purpose the pressures in the column of cells $i = \text{IMAX}$ were set according to $P^{n+1}_{\text{IMAX},j} = P^{n}_{\text{IM1},j}$. In an analogous fashion, the Turn function at the outflow boundary was set according to $(t')^{n+1}_{\text{IM1},j} = (t')^{n}_{\text{IM2},j}$. Other outflow boundary treatments are undoubtedly possible, but having found this "continuative" outflow boundary to be adequate, we did not pursue the matter further.

COMPUTATIONAL RESULTS

In this section we describe the results of three series of calculations performed both with the old LDEF program and the version modified to include the TFV method. The purpose of the calculations was to test the ease of implementation of the TFV method and to see how it affected computational accuracy. The results demonstrate both the versatility of the method and the large errors that can result when the vorticity equation is not solved in compressible flow calculations.

Problem 1. *Incompressible Vortex*

The problem was to calculate a Rankine vortex located at the center of a square box. Figure 5 shows the computer-generated velocity vector and vorticity contour plots at time $t = 0.0$. All quantities are specified in nondimensional units. In this and all subsequent contour plots, H and L are the highest and lowest values associated with contour lines. The initial vortex had a core with angular velocity one ($w = 2$) and a radius of one. The sound speed was ten, and hence, the flow was nearly incompressible. The dimensions of the box were twelve units on a side, and all boundaries were rigid, free-slip walls. The box was resolved with a uniform mesh of square cells

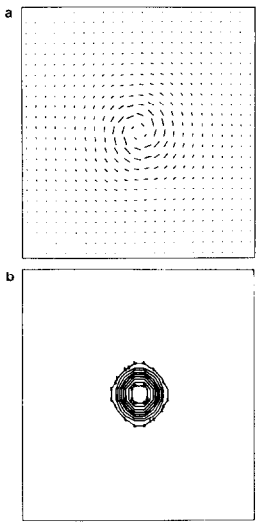

FIG. 5. Velocity vectors (a) and vorticity contours (b) at time $t = 0.0$ in the incompressible vortex calculations, ($H = 1.80$, $L = 0.20$).

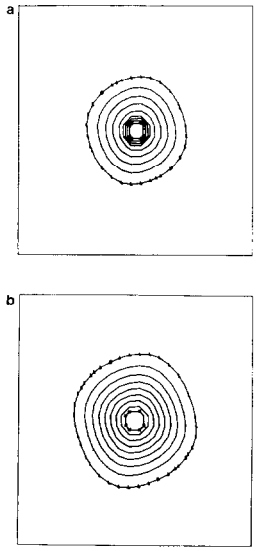

FIG. 6. Vorticity contour plots at time $t = 25.0$ in the calculations with (a) ($H = 0.79$, $L = 0.09$) and without (b) ($H = 0.49$, $L = 0.05$) the TFV method.

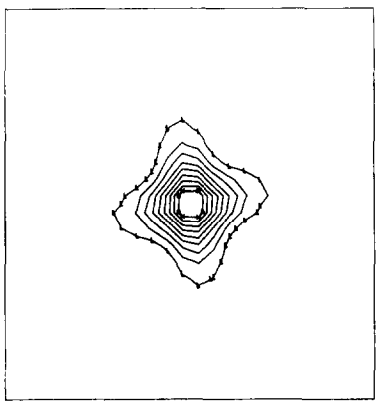

FIG. 7. Turn function contour plot, ($H = 0.019$, $L = 0.001$).

with $\delta x = 0.5$, and the computational time step was $\delta t = 0.125$. Because the flow was inviscid and incompressible, the exact solution was circulation-preserving. Thus for the exact solution both the area of the vortex core and the total vorticity contained in it should be constant, although the shape of the core could change due to boundary effects.

Figure 6 gives vorticity contour plots for the calculations with an without the TFV method at a time $t = 25.0$, which is approximately four revolution-times of the original vortex core. It is seen that the high vorticity contour value in the TFV calculation is 60% higher than in the LDEF calculation, and that there has been less spreading of the core vorticity. The diminished vorticity values in both calculations were not due to a lack of conservation of total vorticity. The TFV calculation conserved total vorticity while the LDEF calculation lost less than one percent of the original vorticity in the mesh. The errors in both calculations were due to numerical diffusion of vorticity, and this diffusion was much less in the TFV method calculation.

The plot of the Turn function in Fig. 7 is informative. The gradients in the Turn function are largest where the largest production of discrepancy $D = w - ((\partial v/\partial x) - (\partial u/\partial z))$ occurs due to truncation errors. It can be seen that the largest truncation errors occur where the flow is in the direction of cell diagonals.

Problem 2. *Isothermal Bow Shock*

The purpose of this calculation was to test our treatment of shocks in the TFV method. The geometry for this planar shock problem is shown in Fig. 8. A uniform Mach two flow entered the computational mesh from the left. At a distance of 3.0 from the inflow boundary, the flow encountered a step with a height of 1.0. Again, all units are nondimensionalized. The top and bottom boundaries, which were separated by a distance of 8.0, were rigid, free-slip walls. The outflow boundary was located 5 units downstream from the inflow boundary.

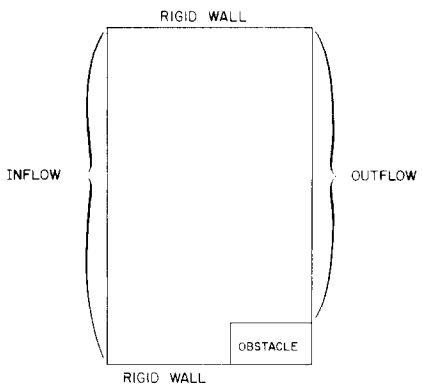

FIG. 8. Geometry for all the bow shock calculations.

Numerical treatment of the corner of the step in Fig. 8 requires careful consideration. Experimentally it is observed that the flow along the leading edge of the obstacle separates at the corner. When the flow separates, the vorticity associated with the very thin boundary layer along the leading edge, is injected into the interior of the flow region. To include this effect in our TFV method calculation, we allow a vorticity flux through the bottom boundary of the vorticity cell immediately above the corner of the obstacle. This vorticity flux was computed for the surface shown in Fig. 9 extending from point (x_1, y_1) at the center of the top boundary of cell (i, j) to the point (x_2, y_1) at the corner of the obstacle. Although this surface does not coincide with the bottom boundary of the vorticity cell above the corner (the dashed line in Fig. 9), we compute the vorticity flux through it because it is the surface across which vorticity enters the flow. The vorticity flux is simply computed from

$$\int_{x_1}^{x_2} w(x, y_1)\, v(x, y_1)\, dx \approx \int_{x_1}^{x_2} v \frac{\partial v}{\partial x}\, dx = -\frac{1}{2}(v_{ij})^2.$$

We have used here the facts that the u velocity is small near the wall and that v is zero on the wall. These considerations are not encountered when one does not solve the vorticity equation, and, as the computational results will show, the TFV method can give a much more accurate calculation of the flow around corners.

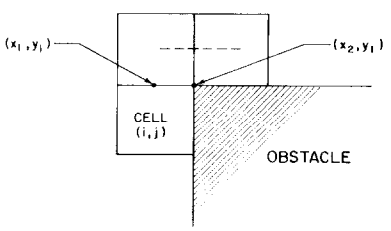

FIG. 9. Enlarged view of the corner of the obstacle. The vorticity flux was computed for the surface extending from (x_1, y_1) to (x_2, y_1).

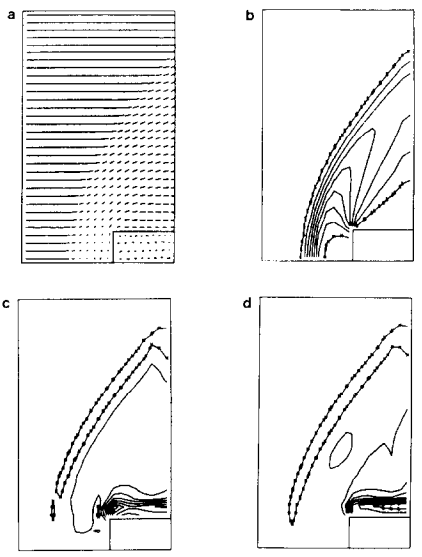

FIG. 10. Plots from the steady solution obtained with LDEF. Shown are velocity vectors (a) and contours of ρ (b) ($H = 4.09$, $L = 1.21$), w (c) ($H = 0.64$, $L = -10.40$), and w/ρ (d) ($H = 0.49$, $L = -8.64$).

Figure 10 shows plots from the steady solution obtained with the LDEF code. The velocity vectors at first turn slightly upward in the shock (positive vorticity) and then behind the shock all the angular motion is in a clockwise direction (negative vorticity). The density behind the shock is highest where the shock is normal to the flow, and the value associated with the highest density contour (4.09) is close to the value the density should have along the center line immediately behind the shock (4.00). When the flow behind the shock goes around the corner of the obstacle, it becomes supersonic, and goes through a rarefaction fan. Because of this the fluid densities along the top of the obstacle are actually lower than the inflow value. The plots of w and w/ρ show the regions of small positive vorticity in the shock and negative vorticity behind the shock, but the most dominant feature of these plots is the small region of intense vorticity near the corner of the obstacle. This is associated with the flow separation phenomenon. Flow separation was not calculated (as is evidenced in the velocity vector plot) due to insufficient numerical resolution at the corner. This solution computed by LDEF is in qualitative agreement with the results of other numerical studies of bow shocks [9].

When we computed this same problem using the TFV method with a very large value of μ_t, a steady solution was not obtained. This is seen in Fig. 11, which gives plots of velocity vectors and contours of density at time $t = 1.6$. The shock has become less oblique to the inflow and has intersected the upper boundary of the computational region. After $t = 1.6$, the shock continues to move upstream. We postulated that this behavior was due to the large upstream disturbances in the

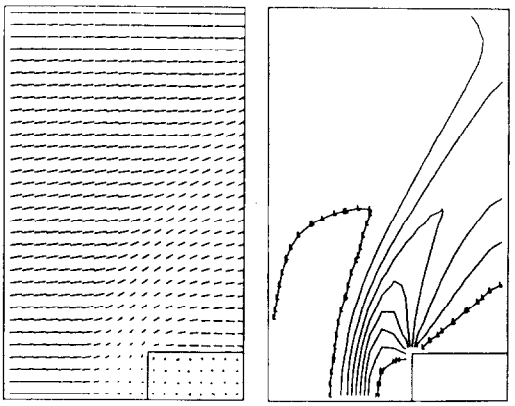

FIG. 11. Velocity vectors and density contours at time $t = 1.6$ in the TFV method calculation with $\mu_t = \infty$, $(H = 3.68, L = 0.98)$.

computed velocity field of Fig. 11. The possibility that such disturbances could occur has already been discussed. In this calculation, positive v velocities were computed before the switching parameter β signaled that a shock was present. Thus the vorticity equation was being used and the constraint $w = 0$ enforced, in the region of positive v velocities. Apparently this vorticity constraint forced positive v velocities even at the inflow boundary. The altered flow direction ahead of the shock caused the shock to change its angle and to intersect the upper boundary. Clearly the numerical errors in

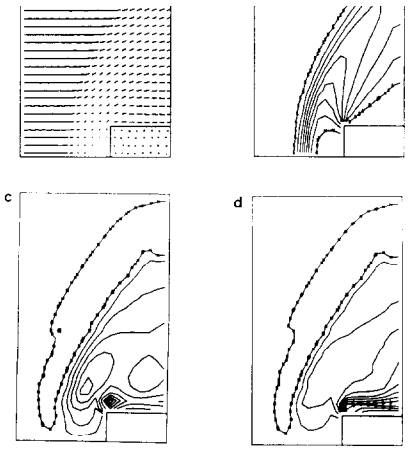

FIG. 12. Plots from the steady solution of the finite μ_t TFV equations. Velocity vectors and contours of ρ (b) $(H = 3.89, L = 1.13)$, w (c) $(H = 0.07, L = -4.38)$, and w/ρ (d) $(H = 0.08, L = -2.71)$.

When the TFV calculation was repeated with $\mu_t = \rho cL$ and $L = 1.0$, a steady solution was obtained. The plots of Fig. 12 show that the upstream disturbances in the velocity field have been almost completely eliminated, and the shock location is nearly the same as in the LDEF calculation. In fact the solutions are very close except for the region close to and downstream of the corner of the obstacle. That the two solutions are close away from the obstacle can be seen more clearly by comparing the plots of Fig. 12 with those of Fig. 13, which are contour plots of the LDEF solution using the same values associated with the contour lines as in Fig. 12. Near the corner of the obstacle the maximum magnitude of the vorticity in the TFV calculation was 4.93, and in the LDEF calculation it was 11.78. Thus the TFV calculation gave the same vorticity production due to the shock as the LDEF calculation, but gave a more accurate calculation of the flow near the corner of the obstacle.

Figure 14 gives plots of the Turn function and the switching parameter β. The Turn function plot again shows that the largest differences between the TFV and LDEF calculations occurred near the corner of the obstacle. On the β contour plot, we have indicated the region where the vorticity equation was not solved. The velocity and length scales used to compute β (see Eq. 13) were the inflow boundary velocity and $L = 20$.

The exact values for the vorticity behind a curved shock can be calculated in terms of the shock curvature, angle, and the upstream fluid variables [10]. Agreement can be obtained between our calculation and theory, but due to the large uncertainty in the shock location in the calculation, such a comparison is not very meaningful.

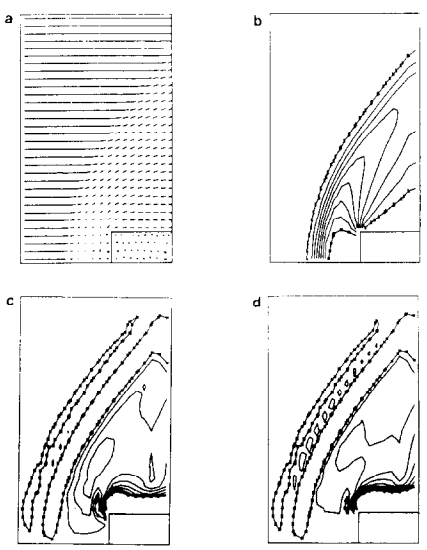

FIG. 13. Plots of the steady LDEF solution using the same contour values as in the contour plots of Fig. 12. Velocity vectors and contours of ρ (b) ($H = 3.89$, $L = 1.13$), w (c) ($H = 0.07$, $L = -4.38$), and w/ρ (d) ($H = 0.08$, $L = -2.71$).

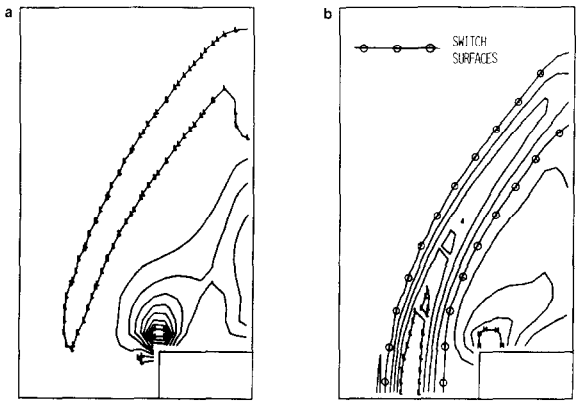

FIG. 14. Plots of the Turn function (a) ($H = 18.9$, $L = -1.6$) and switching parameter β (b) ($H = 3.14$, $L = -4.16$) from the TFV method steady solution.

Problem 3. *Shock–Vortex Interaction*

The problem to be solved was that of an isothermal shock passing over a vortex in a square box. Figure 15 shows the computer-generated vorticity contour plot at time $t = 0.0$ All quantities are specified in nondimensional units. The initial Rankine vortex had a core with angular velocity equal to one ($w = 2$) and a radius of one. The initial fluid density in the box was one, and the sound speed was ten. The dimensions of the box were three units on each side. The top, right, and bottom boundaries were rigid, free-slip walls. The left boundary was an inflow boundary at which the normal component of the velocity was specified to be 15.0 and the density of the inflowing fluid was specified to be 4.0. This resulted in a shock with density (and pressure) ratio 4.0 propagating from left to right with a speed of 20.0. The reflected shock from

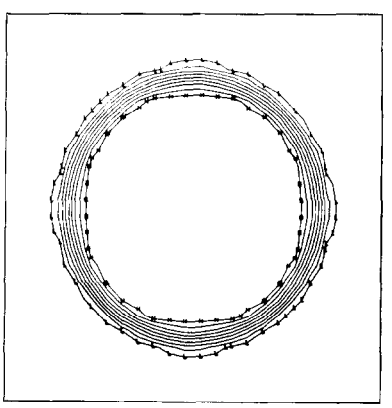

FIG. 15. Vorticity contour plot at time $t = 0.0$ for the shock-vortex interaction calculation, ($H = 1.80$, $L = 0.20$).

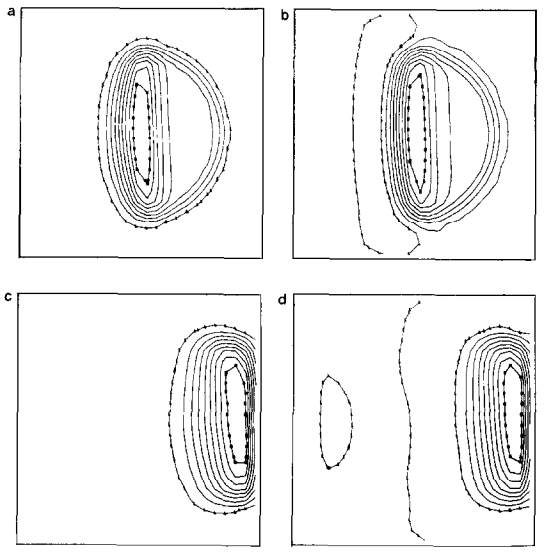

FIG. 16. Contour plots of vorticity at (a) $t = 0.075$, $H = 4.04$, $L = 0.33$ from the TFV calculation, (b) $t = 0.075$, $H = 3.49$, $L = -0.35$ from the LDEF calculation, (c) $t = 0.15$, $H = 4.65$, $L = 0.30$ from the TFV calculation, and (d) $t = 0.15$, $H = 4.15$, $L = -0.04$ from the LDEF calculation.

the right wall also has a density ratio of 4.0 and propagated to the left with a speed of 5.0. The box was resolved with a uniform mesh of square cells with $\delta x = 0.125$. The computational time step δt was 0.002.

Figure 16 shows plots of the vorticity obtained in calculations with and without the TFV method at times $t = 0.075$ and $t = 0.15$, when the incoming shock had just reached the right boundary of the box. For the TFV calculation the finite μ_t formulation was used with $L = 1.0$. At time $t = 0.075$, the shock has just compressed the left half of the vortex core, and at time $t = 0.15$, the fluid behind the shock, has swept the vortex downstream to the right wall of the box. The shapes of the contours obtained in the two solutions look nearly the same; however, the values associated with the high contours at $t = 0.15$ differ by ten percent. In fact at $t = 0.15$ in the calculation with the old LDEF computer code the total circulation in the mesh was 3.40, and in the TFV calculation it was 5.32. The original total circulation was 6.28. While confirmation of accuracy through comparison with analytic solutions is not possible, it is clear that in the LDEF calculation there has been considerable numerical damping of the vortical motion.

The differences between the two calculations are even more striking at times after the shock reflects off the right wall ($t = 0.15$). Figure 17 gives vorticity contour plots at $t = 0.30$. The high contour value in the LDEF calculation is 72% of that in the TFV calculation. There are two reasons for the appearance of this large discrepancy at times after $t = 0.15$, First, the LDEF calculation continues to lose vorticity due to numerical damping. The second reason has to do with the method we used to

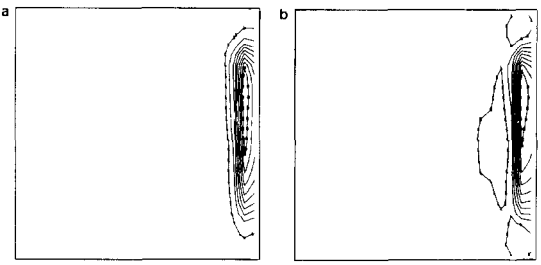

FIG. 17. Contour plots of vorticity at time $t = 0.30$ from the TFV calculation (a) ($H = 7.68$, $L = 0.28$) and LDEF calculation (b) ($H = 5.52$, $L = -0.38$).

difference the convective terms in the vorticity equation. This method was conservative but fairly diffusive. Up to $t = 0.15$, the vortex-core region was moved about ten cells to the right, and there was a large amount of numerical diffusion of the vorticity of the core region. This diffusion kept the values associated with the high contour in the two calculations relatively close before $t = 0.15$. Between $t = 0.15$ and $t = 0.30$, the core region is compressed by the reflected shock, but its position is nearly unchanged. There is less numerical diffusion of vorticity during this time, and the discrepancy between the two calculations widens. These numerical diffusion errors are not, of course, inherent in the TFV method, and they can be reduced by using a more accurate numerical method for convecting vorticity.

Problem 4. *Mach* 0.4 *Potential Flow Over a Disc*

The purpose of this calculation was to demonstrate some of the increased flexibility one obtains by using the TFV method. This problem cannot be calculated with either a stream function and vorticity method, because $\nabla \cdot \vec{u} \neq 0$, or with a traditional compressible flow method, in which the irrotationality condition cannot be specified.

The calculation was of an axisymmetric, Mach 0.4, potential flow over a circular disc. An enthalpy equation was solved for this calculation, and the equation of state was that of an ideal gas with $\gamma = C_p/C_v = 1.4$. The computational mesh had 30 cells axially and 20 cells radially, and the cell dimensions were constant with $\delta r = \delta z = 0.20$. The bottom boundary was the symmetry axis and the top boundary was a rigid wall. At the left boundary, uniform inflow conditions were specified, and the density was set equal to one. The right boundary was a continuative outflow boundary. The disc had a radius of one.

Figure 18 shows the velocity vectors and density contour plots of the steady solution. The minimum density behind the disc is 0.673. When we performed this same calculation but omitted the pressure gradient terms in the vorticity equation, the same numerical solution was obtained. Since the flow in this problem was homentropic, we have

$$\nabla \times \frac{1}{\rho} \nabla p(\rho, s) = -\left(\nabla \rho \times \frac{\partial p}{\partial \rho}\bigg|_s \nabla \rho\right) \Big/ \rho^2 = 0,$$

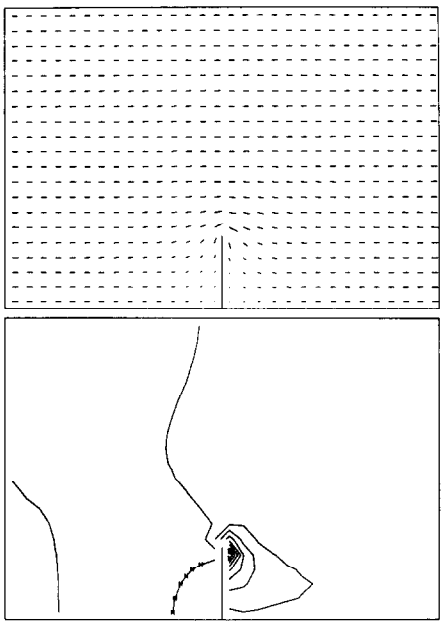

FIG. 18. Velocity vectors and density contours for the computed steady solution of the Mach 0.4 potential flow over a disc, ($H = 1.06$, $L = 0.72$).

for the exact solution. Thus the truncation errors associated with the finite difference approximation to $\nabla \times (1/\rho)\nabla p$ were small.

CONCLUSIONS

A numerical method, called the Turn function and vorticity method, has been developed that will increase the accuracy of compressible fluid-flow calculations in consistent fashion and is easy to implement in existing compressible flow codes. Better computational results are obtained with a version of the method in which the vorticity is maintained close to but not equal to the curl of the velocity field. In problems with shocks, it is necessary to set the Turn function to zero in the neighborhood of the shocks in order to calculate the correct jump conditions. The TFV method gave more accurate results in example calculations of Rankine vortices, shock–vortex interactions, and a bow shock in front of a blunt obstacle.

ACKNOWLEDGMENTS

I would like to thank Dr. Francis H. Harlow for his help in this work and for first suggesting to me that a numerical method might be devised that solves both the linear-momentum and vorticity equations. This work was funded by institutional supporting research and the Center for Nonlinear Studies of the Los Alamos National Laboratory.

REFERENCES

1. R. ARIS, "Vectors, Tensors, and the Basic Equations of Fluid Mechanics," Prentice–Hall, Englewood Cliffs, N.J., 1962.
2. P. J. ROACH, "Computational Fluid Dynamics," Hermosa, Albuquerque, N. Mexico, 1976.
3. J. E. FROMM AND F. H. HARLOW, *Phys. Fluids* **6** (1963), 975.
4. F. H. HARLOW AND J. E. WELCH, *Phys. Fluids* **8** (1965), 2182.
5. J. NACHAMKIN, "The Invariant Decomposition of the Equations Governing Hydrodynamics Flow and Thermal Conductivity," Los Alamos National Laboratory Report LA-5547-MS, 1974.
6. C. W. HIRT, *J. Comput. Phys.* **2** (1968), 339.
7. P. J. O'ROURKE, "Collective Drop Effects on Vaporizing Liquid Sprays," Los Alamos National Laboratory Report LA-9069-T, 1981.
8. F. H. HARLOW AND A. A. AMSDEN, *J. Comput. Phys.* **8** (1971), 197.
9. T. D. BUTLER, *AIAA J.* **4** (1965), 460.
10. W. D. HAYES, *J. Fluid. Mech.* **2** (1957), 595.