Fresh Look at Floating Shock Fitting

Peter M. Hartwich*
ViGYAN, Inc., Hampton, Virginia 23666

A fast implicit upwind procedure for the two-dimensional Euler equations is described that allows accurate computations of shocked flows on nonadapted meshes. The upwinding is based on the nonconservative split-coefficient-matrix (SCM) method that is extended here in two ways: 1) floating shock fitting has been implemented for proper treatment of shocks, and 2) a diagonalized approximate factorization (AF) algorithm has been devised to promote convergence to steady-state solutions. Results are presented for Riemann's problem, for a regular shock reflection at an inviscid wall, for supersonic flow past a cylinder, and for a transonic airfoil. These results demonstrate 1) that all shocks are ideally sharp and in excellent agreement with other numerical results or "exact" solutions, 2) that this has been accomplished on unusually crude meshes without any attempt to align grid lines with shock fronts or to cluster grid lines around shocks, and 3) that it is possible to construct general shock-fitting schemes.

I. Introduction

ALTHOUGH Euler solvers using shock fitting have always been known for being faster than shock-capturing schemes[1-3] and of equal if not better accuracy,[1-4] they have also been notorious for requiring skilled personal intervention every time a new problem is to be studied. It was particularly this shortcoming that led to the demise of shock-fitting methods and to the ascent of shock-capturing schemes. For a long time, shock capturing has been treated as the sole candidate for constructing largely user-independent, general-purpose Euler solvers. This conviction roots in the fact that shock-capturing schemes solve for the conservative Euler equations that admit shocks as weak solutions.[5] Thus, shock-capturing methods can be highly automated in the sense of using the same differencing concept in regions of smooth flow as well as across shocks.

For years, only Moretti and a few dedicated colleagues (see, for instance, Refs. 6-8 and the references cited there) have prevented shock fitting from extinction by producing several accurate floating-shock-fitting schemes based on Moretti's λ-formulation.[9] But lately, the interest of the CFD community in the concept and notions of shock fitting appears to be on the rise again. Elements of shock fitting have been introduced in some shock-capturing methods to enhance their accuracy.[10-12] Other researchers feel that it might be time to reconsider pure shock-fitting schemes.[4,13]

This report falls into the latter category. The objective has been to formulate a general, nonconservative Euler solver using floating shock fitting that can handle all kinds of shocked flows with user-specified input limited to grid geometry, initial and boundary conditions. To expedite the development of such a procedure, proven numerical technology has been utilized whenever possible. The formulation of the governing equations follows closely that of the split-coefficient-matrix (SCM) method.[14] Away from shocks, second-order-accurate, fully one-sided spatial differences are used. Differences across shocks are efficiently suppressed via operator arrays. The solution is advanced in time with a diagonalized approximate factorization (AF) algorithm that was originally introduced by Pulliam and Chaussee.[15] Shocks are explicitly computed using several of Moretti's ideas and notions[6,16] with one major change: The shock orientation is computed locally by observing that the shock normal runs parallel to the change in velocity across shocks. This approach has been originally employed by Davis[10] for enhancing the accuracy of shock-capturing schemes across shocks by using rotated differences.

The formulation of this method is given in the next six sections. The resulting scheme has been applied to Riemann's problem to demonstrate its ability to accurately track transient shock waves. The accuracy and efficiency of the code are illustrated for an inviscid shock reflection problem, in which the shocks run obliquely across a Cartesian mesh. The computations of supersonic flow past a circular cylinder show that the code sustains its performance on curvilinear, body-fitted grids that are not tailored to a strong bow shock. The code also does well in computing transonic airfoils with embedded shocks on unadapted meshes.

II. Governing Equations

Upwind methods are aimed at constructing finite-difference schemes based on the theory of characteristics. Information about characteristic behavior becomes useful along the perimeter of the integration domain and along discontinuities. Nonconservative shock-fitting schemes can incorporate upwinding very cheaply, since the dependent variables can be chosen for convenience. Three, in a way, "natural" choices are a (speed of sound) and u, v (Cartesian velocities), for they are required in any upwind scheme. The fourth choice is s (entropy) to reduce the number and complexity of the terms in the governing equations.

If we let

$$\delta = (\gamma - 1)/2 \qquad (\gamma = \text{ratio of specific heats}) \tag{1}$$

the two-dimensional, compressible, nonconservative Euler equations for a polytropic gas at constant γ are written in general coordinates as

$$\boldsymbol{Q}_t + \boldsymbol{A}\boldsymbol{Q}_\xi + \boldsymbol{B}\boldsymbol{Q}_\eta = 0 \tag{2}$$

with

$$\boldsymbol{Q} = (a, u, v, s)^T$$

All quantities are normalized with reference values for pressure, density, and length (p_{ref}, ρ_{ref}, and x_{ref}, respectively). The reference velocity, temperature, and time are thus

$$v_{ref} = \sqrt{p_{ref}/\rho_{ref}} \qquad T_{ref} = p_{ref}/(R\rho_{ref})$$

$$t_{ref} = x_{ref}/v_{ref} \tag{3}$$

Received Nov. 8, 1989; presented as Paper 90-0108 at the AIAA 28th Aerospace Sciences Meeting, Reno, NV, Jan. 8-11, 1990; revision received May 17, 1990; accepted for publication May 29, 1990.

*Research Scientist. Member AIAA.

with R being the specific gas constant. The nondimensional entropy is defined as

$$s = \ell n(p/\rho^{\gamma})/(2\gamma\delta) \tag{4}$$

A coefficient matrix $\boldsymbol{C}$ is defined as

$$\boldsymbol{C} = \begin{bmatrix} \lambda_5 & -\hat{\alpha}\delta\lambda_6 & \hat{\beta}\delta\lambda_6 & a\delta(2\lambda_2 - \lambda_5) \\ -\hat{\alpha}\lambda_6/\delta & \hat{\alpha}^2\lambda_5 + 2\hat{\beta}^2\lambda_2 & \hat{\alpha}\hat{\beta}(\lambda_5 - 2\lambda_2) & a\hat{\alpha}\lambda_6 \\ -\hat{\beta}\lambda_6/\delta & \hat{\alpha}\hat{\beta}(\lambda_5 - 2\lambda_2) & \hat{\beta}^2\lambda_5 + 2\hat{\alpha}^2\lambda_2 & a\hat{\beta}\lambda_6 \\ 0 & 0 & 0 & 2\lambda_2 \end{bmatrix} \tag{5}$$

The eigenvalues of $\boldsymbol{C}$ are

$$\lambda_1 = (\alpha u + \beta v - a\sqrt{\alpha^2 + \beta^2})/2$$

$$\lambda_2 = \lambda_3 = (\alpha u + \beta v)/2$$

$$\lambda_4 = (\alpha u + \beta v + a\sqrt{\alpha^2 + \beta^2})/2 \tag{6}$$

They are combined in Eq. (5) for reasons of compactness as

$$\lambda_5 = \lambda_1 + \lambda_4 \qquad \text{and} \qquad \lambda_6 = \lambda_1 - \lambda_4$$

The metric coefficients in Eq. (5) are normalized as

$$\hat{\alpha} = \alpha/\sqrt{\alpha^2 + \beta^2} \qquad \hat{\beta} = \beta/\sqrt{\alpha^2 + \beta^2}$$

For $\alpha = \xi_x$ and $\beta = \xi_y$, $\boldsymbol{C} = \boldsymbol{A}$, and for $\alpha = \eta_x$ and $\beta = \eta_y$, $\boldsymbol{C} = \boldsymbol{B}$. The metric coefficients are

$$\xi_x/J = y_\eta \qquad \eta_x/J = -y_\xi$$

$$\xi_y/J = -x_\eta \qquad \eta_y/J = x_\xi$$

$$J^{-1} = x_\xi y_\eta - x_\eta y_\xi \tag{7}$$

III. Spatial Differencing

The coefficient matrix $\boldsymbol{C}$ can be split according to the sign of its eigenvalues

$$\boldsymbol{C} = \boldsymbol{C}^{+} - \boldsymbol{C}^{-} \tag{8}$$

with $\boldsymbol{C}^{\pm} = \boldsymbol{C}(\lambda_m^{\pm})$, $m = 1, 2, 3, 4$, and

$$\lambda^{\pm} = (|\lambda_m| \pm \lambda_m)/2 \tag{9}$$

Upon inserting Eq. (8), Eqs. (2) assumes

$$\boldsymbol{Q}_t - \boldsymbol{A}^{-}\boldsymbol{Q}_\xi + \boldsymbol{A}^{+}\boldsymbol{Q}_\xi - \boldsymbol{B}^{-}\boldsymbol{Q}_\eta + \boldsymbol{B}^{+}\boldsymbol{Q}_\eta = 0 \tag{10}$$

A semidiscrete finite-difference approximation to Eq. (10) that is numerically stable in the sense of a linear stability analysis as in Steger and Warming[17] reads

$$\boldsymbol{Q}_t - \boldsymbol{A}^{-}\Delta^{+}_{i+1/2}\boldsymbol{Q} + \boldsymbol{A}^{+}\Delta^{-}_{i-1/2}\boldsymbol{Q} - \boldsymbol{B}^{-}\Delta^{+}_{j+1/2}\boldsymbol{Q} + \boldsymbol{B}^{+}\Delta^{-}_{j-1/2}\boldsymbol{Q} = 0 \tag{11}$$

with

$$\Delta^{+}_{i+1/2}\boldsymbol{Q} = (3/2 - \phi_1)\Delta_{i+1/2}\boldsymbol{Q} - (1/2 - \phi_1)\Delta_{i+3/2}\boldsymbol{Q}$$

$$\Delta^{-}_{i-1/2}\boldsymbol{Q} = (3/2 - \phi_2)\Delta_{i-1/2}\boldsymbol{Q} - (1/2 - \phi_2)\Delta_{i-3/2}\boldsymbol{Q}$$

$$\Delta^{+}_{j+1/2}\boldsymbol{Q} = (3/2 - \phi_3)\Delta_{j+1/2}\boldsymbol{Q} - (1/2 - \phi_3)\Delta_{j+3/2}\boldsymbol{Q}$$

$$\Delta^{-}_{j-1/2}\boldsymbol{Q} = (3/2 - \phi_4)\Delta_{j-1/2}\boldsymbol{Q} - (1/2 - \phi_4)\Delta_{j-3/2}\boldsymbol{Q} \tag{12}$$

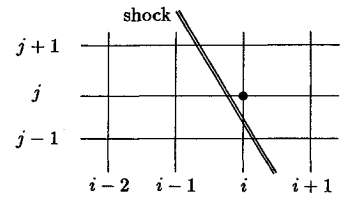

Fig. 1 Computation in the neighborhood of a shock wave.

and

$$\Delta_{l-1/2}\boldsymbol{Q} = \boldsymbol{Q}_l - \boldsymbol{Q}_{l-1}, \qquad l = i \text{ or } j$$

where $\boldsymbol{Q}_l$, $\boldsymbol{A}^{\pm}$, $\boldsymbol{B}^{\pm}$, and ϕ_k with $k = 1, 2, 3, 4$ are taken at the centroids (i, j). For $\phi_1 = \phi_2 = \phi_3 = \phi_4 = 0$, Eq. (11) gives the semidiscrete baseline scheme that uses second-order, one-sided differences. All mesh points, except those in the neighborhood of a shock wave, are computed with the same baseline scheme. Close to shocks, this baseline scheme is modified via the parameters ϕ_k to accommodate a floating-shock-fitting technique.

Consider, for example, the computations of the points $(i-2, j)$, $(i-1, j)$, (i, j), and $(i+1, j)$ in Fig. 1. In order to avoid taking differences across the shock, the forward differences at $(i-2, j)$ and the backward differences at $(i+1, j)$, both in the i direction, are given by first-order, one-sided differences. That is accomplished by setting

$$(\phi_1)_{i-2,j} = 1/2 \qquad \text{and} \qquad (\phi_2)_{i+1,j} = 1/2 \tag{13}$$

Forward differences at $(i-1, j)$ and backward differences at (i, j), both again in the i direction, are altogether suppressed. This is easily coded when the discrete equivalent to Eq. (9) for the eigenvalues associated with the matrix $\boldsymbol{A}$ is written as

$$(\lambda_m^{-})_{i,j} = [\,|\,(\lambda_m)_{i,j}\,| - (\lambda_m)_{i,j}][1/2 - (\phi_1)_{i-1,j}]$$

$$(\lambda_m^{+})_{i,j} = [\,|\,(\lambda_m)_{i,j}\,| + (\lambda_m)_{i,j}][1/2 - (\phi_2)_{i+1,j}] \tag{14}$$

Analogous expressions are easily derived to control the one-sided differences in the j direction at $(i, j-2)$, $(i, j-1)$, (i, j), and $(i, j+1)$ for a shock transversing an $i = \text{const}$ line between (i, j) and $(i, j-1)$ (see Fig. 1).

IV. Time Differencing

A time-implicit operator containing block-tridiagonal matrices for the two-dimensional Euler equations is given by

$$[\boldsymbol{I} - \psi\tau(\boldsymbol{A}^{-}\Delta_{i+1/2} - \boldsymbol{A}^{+}\Delta_{i-1/2} + \boldsymbol{B}^{-}\Delta_{j+1/2} - \boldsymbol{B}^{+}\Delta_{j-1/2})]^n\Delta\boldsymbol{Q}^n = -\tau \cdot \mathrm{RES}(\boldsymbol{Q}^n) \tag{15}$$

where $\boldsymbol{I}$ = identity matrix, τ = time step, $\Delta\boldsymbol{Q}^n = \boldsymbol{Q}^{n+1} - \boldsymbol{Q}^n$, and n indicates the time level. As before, the Jacobians $\boldsymbol{A}$ and $\boldsymbol{B}$ are formed at the centroids (i, j). For $\psi = 1/2$, Eq. (15) resembles a second-order-accurate Crank–Nicholson scheme, which is used in calculating transient flows. For $\psi = 1$, Eq. (15) recovers first-order-accurate Euler backward-time differencing, which is preferred for steady-state applications since it produces lower asymptotic spectral radii than the Crank-Nicholson scheme due to its better damping properties. The algorithm is cast in delta form to compute steady-state solutions that are independent of the time step size. The residual $\mathrm{RES}(\boldsymbol{Q}^n)$ comprises the spatial differences in Eq. (11) evaluated at the nth time level with the provisos as discussed in the preceding section.

With approximate factorization (AF), a standard noniterative solution algorithm[15] for Eq. (15) is written as

$$[I - \psi\tau(A^{-}\Delta_{i+1/2} - A^{+}\Delta_{i-1/2})]^n \times [I - \psi\tau(B^{-}\Delta_{j+1/2} - B^{+}\Delta_{j-1/2})]^n \Delta Q^n = -\tau \cdot \mathrm{RES}(Q^n) \tag{16}$$

By using a similarity transformation such as $C = T\Lambda T^{-1}$, $\Lambda = \mathrm{diag}(\lambda_m)$ (diag = diagonal matrix) with an orthonormal set of left and right eigenvectors $TT^{-1} = I$, a diagonalized AF algorithm for Eq. (16) is written as

$$\{I - \psi\tau[(\Lambda^{-})^A\Delta_{i+1/2} - (\Lambda^{+})^A\Delta_{i-1/2}]\}^n \Delta\tilde{\tilde{Q}} = -(T^{-1})^A\tau \cdot \mathrm{RES}(Q^n) \tag{17a}$$

$$\{I - \psi\tau[(\Lambda^{-})^B\Delta_{j+1/2} - (\Lambda^{+})^B\Delta_{j-1/2}]\}^n \Delta\tilde{Q} = M\Delta\tilde{\tilde{Q}} \tag{17b}$$

$$Q^{n+1} = Q^n + T^B\Delta\tilde{Q} \tag{17c}$$

with $M = (T^{-1})^B T^A =$

$$\begin{bmatrix} m_3 & m_2\delta/\sqrt{2} & 0 & m_4 \\ -m_2/(\delta\sqrt{2}) & m_1 & 0 & -m_2/(\delta\sqrt{2}) \\ 0 & 0 & 1 & 0 \\ m_4 & m_2\delta/\sqrt{2} & 0 & m_3 \end{bmatrix}$$

where

$$m_1 = \hat{\alpha}^A\hat{\alpha}^B + \hat{\beta}^A\hat{\beta}^B \qquad m_3 = (m_1 + 1)/2$$

$$m_2 = \hat{\alpha}^A\hat{\beta}^B - \hat{\alpha}^B\hat{\beta}^A \qquad m_4 = (m_1 - 1)/2$$

The time-implicit algorithm entailed in Eqs. (17) requires some provisions along shock fronts. Since shocks are explicitly computed and all differences across shocks are suppressed, the fitted shocks act like internal boundaries, even though they are allowed to float between the fixed grid points. The fact that the solution along the high-pressure side of shocks is updated in a separate step similar to a time-explicit updating of the boundary conditions along the perimeter of the computational domain suggests the modifications to the time-implicit algorithm around shock points. When the boundary conditions are updated in a time-explicit fashion, the changes $\Delta\tilde{Q}$ and $\Delta\tilde{\tilde{Q}}$ are usually set to zero along the perimeter of the integration domain.[18,19] This is done for reasons of convenience and generality, albeit this locally reduces the time accuracy to first order.[18] Particularly, the generality aspect lends itself to a simple adoption of this approach to the treatment of shocks as internal boundaries by writing

$$-\psi\Delta t(\Lambda^{+})^A(\Delta\tilde{\tilde{Q}})_{i-1} + [I + \psi\Delta t \mid \Lambda^A \mid](\Delta\tilde{\tilde{Q}})_i - \psi\Delta t(\Lambda^{-})^A(\Delta\tilde{\tilde{Q}})_{i+1} = -(T^{-1})^A\Delta t \cdot \mathrm{RES}(Q^n) \tag{18a}$$

$$-\psi\Delta t(\Lambda^{+})^B(\Delta\tilde{Q})_{j-1} + [I + \psi\Delta t \mid \Lambda^B \mid](\Delta\tilde{Q})_j - \psi\Delta t(\Lambda^{-})^B(\Delta\tilde{Q})_{j+1} = M\Delta\tilde{\tilde{Q}} \tag{18b}$$

Finally, Q^{n+1} is computed from Eq. (17c). In the absence of shocks or for unsteady problems, where a global time step is chosen such that the maximum Courant–Friedrichs–Lewy (CFL) number is less than unity, $\Delta t = \tau$. In the presence of shocks and by employing local time stepping with CFL > 1 to expedite convergence to a steady-state solution, Δt has to be evaluated such that CFL < 1 at the shock points (i.e., points adjacent to the shock and on the high pressure side). This constraint comes about because of the time-explicit shock treatment.

V. Two-Dimensional Shocks

Thus far, floating-shock-fitting methods have exclusively relied on a single concept to compute the shock orientation.[1,3,8,16] After shock points are generated, they are reorganized such that they form successive chains with no link being larger than one mesh interval. This grouping process always requires computationally "if" statements. The orientation of the shock front is usually computed from centered differences between the coordinates of shock points along the shock front. If a shock point is without an immediate neighbor above or below, one-sided differences are employed. If a shock point is isolated, it needs to be dropped. Along inviscid walls, the shock normal is assumed to run parallel to the wall surface. Sometimes "wrinkled" shocks are encountered, which are numerically ironed out via some ad hoc manipulations.[3,8]

Adopting Davis'[10] approach for computing the local shock orientation eliminates the need for the aforementioned precautions and special procedures. Since the local shock orientation is computed independently from neighboring shock points, successive links of shock points need no longer be assembled. Note that this quality forms a keystone for a straightforward extension of floating shock fitting to three dimensions. Moreover, isolated shock points are no longer an issue; they either form a nucleus for a nascent shock or they will fade away. Wrinkles have never been observed with Davis' simple and effective approach that flows from the continuity of the tangential velocity component across a shock.

Figure 2 illustrates the present approach to compute the shock orientation and to update the solution across a shock via the Rankine–Hugoniot relations. Some oblique shock is bracketed by two points, A and B, that lie on a η = const line. Furthermore, assume that $V^* = \bar{\eta}_y u - \bar{\eta}_x v \geq 0$ and that B belongs to the high-pressure region. Then, the direction cosines of the normal shock in a local reference frame are

$$N = \Delta q / \mid \Delta q \mid \tag{19}$$

where $N = (N_1, N_2)^T$ and $q = (V, V^*)^T$ with

$$N_1 = -\Delta_{i-1/2}V/\sqrt{(\Delta_{i-1/2}V)^2 + (\Delta_{i-1/2}V^*)^2}$$

$$N_2 = -\Delta_{i-1/2}V^*/\sqrt{(\Delta_{i-1/2}V)^2 + (\Delta_{i-1/2}V^*)^2}$$

$$V = \bar{\eta}_x u + \bar{\eta}_y v$$

$$V^* = \bar{\eta}_y u - \bar{\eta}_x v \tag{20}$$

Thus, V is the scaled contravariant velocity component normal to η = const lines, and V^* is perpendicular to V and in the ξ direction. The notation $\bar{\eta}_x$ and $\bar{\eta}_y$ indicates that the metric coefficients are either locally evaluated at the midpoint of the interval $(i, i-1)$, or that they are computed as averages from the nodal values at i and $i-1$. Thus, the shock orientation as defined in Eq. (19) depends only on the change in the normal velocity component across a shock.

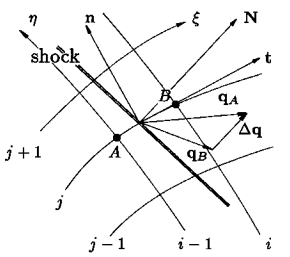

Fig. 2 Updating of a shock point.

The velocity components normal and tangential to the shock, $\tilde{u}$ and $\tilde{v}$, are computed from

$$\tilde{u} = VN_1 + V^*N_2 \tag{21a}$$

$$\tilde{v} = VN_2 - V^*N_1 \tag{21b}$$

These quantities together with entropy and the speed of sound are used to update the solution at point B using the Rankine-Hugoniot relations

$$a_B = a_A \frac{\sqrt{(\gamma M^2 - \delta)(1 + \delta M^2)}}{(1 + \delta) \mid M \mid}$$

$$\tilde{u}_B = \tilde{u}_A - a_A \frac{M^2 - 1}{(1 + \delta) \mid M \mid}$$

$$\tilde{v}_B = \tilde{v}_A$$

$$s_B = s_A + \left[\ell n \frac{\gamma M^2 - \delta}{1 + \delta} - \gamma \, \ell n \frac{(1 + \delta)M^2}{1 + \delta M^2} \right] \Big/ (2\delta\gamma) \tag{22}$$

The symbol M represents the relative shock Mach number

$$\text{sign}(a_B - a_A) \mid M \mid = \frac{\tilde{u}_a - W}{a_A} \tag{23}$$

where W is the shock speed. Moretti[16] developed an implicit function relating the shock Mach number M to a shock parameter Σ

$$\Sigma = \frac{\sqrt{(\gamma M^2 - \delta)(1 + \delta M^2)} + \delta(M^2 - 1)N_2}{(1 + \delta) \mid M \mid} \tag{24}$$

To merge this approach to computing the shock Mach number M with the present approach to determine the shock orientation, the shock parameter Σ is here defined as

$$\Sigma = (a_B - \delta \Delta_{i-1/2} v^*)/a_A \tag{25}$$

The shock parameter is basically the normalized difference of a Riemann variable (i.e., $a/\delta - v^*$) taken at two consecutive grid points. Moretti and DiPiano[6] introduced this shock parameter because it was kind of natural within the framework of Moretti's λ-scheme and because it was demonstrated to be a reliable and sufficiently sensitive shock sensor. The present method is based on the SCM method that is equivalent to the λ-scheme only for two independent variables. In spite of this incongruence of these two baseline schemes, the shock parameter Σ still proved to be useful in the present extension of the SCM method. That is because it comprises two functions: 1) It still provides a reliable means in detecting shock points, and 2) it inherently carries information about the shock Mach number crucial in evaluating the Rankine-Hugoniot relations.

Once the shock parameter Σ of the shock point between A and B is known, M is computed iteratively[16] starting with an initial guess for M, that is,

$$M_0 = 1 + \epsilon_3[\Sigma_{\text{Eq.(24)}} - 1] \tag{26}$$

where

$$\epsilon_1 = \sqrt{(16\gamma - \delta)(1 + 16\delta)}/[4(1 + \delta)]$$

$$\epsilon_2 = 15\delta/[4(1 + \delta)]$$

$$\epsilon_3 = 3/(\epsilon_1 - 1 + \epsilon_2 N_2) \tag{27}$$

The initial guess in Eq. (26) is based on the approximation of $M(\Sigma)$ by a linear function between $M = 1$ and $M = 4$, as in Ref. 16. The three coefficients in Eq. (27) are computed from evaluating Eq. (24) at $M = 4$, which allows for the construction of the following simple and efficient procedure[16]: 1) update Σ by inserting the most recent value for M in Eq. (24), 2) update M with $M = M + \epsilon_3[\Sigma_{\text{Eq.(25)}} - \Sigma_{\text{Eq.(24)}}]$. A handful of iterations suffice to approximate M with an error margin of 10^{-5}.

The final values at B are obtained by assuming B lying right at the shock and applying a linear interpolation between the shock front and the second node to the right of the shock. This interpolation improves accuracy in general and ensures that subsonic downstream conditions are properly communicated to the high-pressure side of the shock.

Finally, the shock is moved along the η = const line by the increment

$$\Delta s = (W/N_2)\Delta t$$

The calculation of the relative shock Mach number does not change at all for $V^* < 0$. If the high-pressure side lies to the left of the shock in Fig. 2, then the framework established by Eqs. (19-27) still holds true, provided the subscripts A and B in Eq. (25) are interchanged. The formulas for a shock crossing a ξ = const line are obtained from interchanging the velocity components u, u^* and v, v^* and from straightforward redefinitions for N_1 and N_2 in Eq. (20).

VI. Detection and Tracking of Shocks

As proposed by Moretti,[3,16] the information pertaining to shock points (node points adjacent to a shock and carrying a B label) is stored in one-dimensional arrays, with the shock points being numbered with a counter J for identification but not necessarily stored in any particular order. Thus, shock information is stored as shock points are generated, and shock points denoted with J and $J + 1$ may be far apart in space. To relate a shock point J to a node (i, j), two-dimensional arrays are introduced: IJA for shocks transversing η = const lines and IJB for shocks crossing ξ = const lines. Initially ($t = 0$), all elements of IJA and IJB are set to zero. When a shock point J is detected at a node (i, j), then its value (i.e., J) is inserted in the corresponding position in either the IJA or IJB arrays, depending on which shock family it belongs to.

The detection of shocks intersecting η = const lines is performed first. Marching in the ξ direction, along each line η = const, Eq. (25) is evaluated at every grid point. If Σ exceeds some tolerance Σ_0 (typically $\Sigma_0 = 1.05$, which translates into a shock Mach number of about 1.08), then Σ is stored in a one-dimensional array at location J; if it is a new shock point, then 1) $J = J + 1$, 2) $IJA(i, j) = J$, 3) the initial guess for the shock location is computed from $\text{SPL}(J) = \sqrt{(\Delta_{i-1/2}x)^2 + (\Delta_{i-1/2}y)^2}/2$, and 4) the location of the high-pressure side (whether B lies to the right or left of the shock front) is indicated by setting a flag in another one-dimensional array.

If a shock point moves into an adjacent mesh interval downstream, the values upstream are extended to the new upstream node. Likewise, the most recent values at the B point are assigned to the new downstream node when the shock point moves into an adjacent upstream mesh interval.

The application of the shock detection algorithm to shocks intersecting ξ = const lines is performed in a similar manner, and the logic for shock tracking applies equally to both shock point families. Finally, it should be mentioned that a shock point may simultaneously belong to both shock point families; the Rankine-Hugoniot relations are then twice applied—the updated values on the high-pressure side are the same for both updates.

VII. Boundary Conditions

For supersonic farfields, all dependent variables are specified. For subsonic flow, the farfield boundary conditions are determined from characteristic-based formulations as readily found, for instance, in Refs. 19 and 20.

Along nonporous walls, the normal velocity is set to zero. The tangential velocity components are computed by linear extrapolation of the Cartesian velocity components onto the surface and subsequent subtraction of the pertinent Cartesian component of the contravariant velocity component normal to the surface. As in Barton and Pulliam,[21] entropy on the wall is computed from simple extrapolation. These boundary conditions for a nonporous wall allow computing the speed of sound along the wall from the momentum equations by simplifying η_x (ξ momentum) + η_y(η momentum):

$$
\begin{aligned}
&-2\delta U(\eta_x u_\xi + \eta_y v_\xi) \\
&= (\xi_x\eta_x + \xi_y\eta_y)(a^2)_\xi + (\eta_x^2 + \eta_y^2)(a^2)_\eta \\
&= \sqrt{\eta_x^2 + \eta_y^2}(a^2)_n
\end{aligned} \tag{28}
$$

where n is the direction normal to the body surface, which is here assumed to coincide with an η = const line. Updated values for a are obtained from Eq. (28) by central differencing $(a^2)_\xi$ and forward differencing $(a^2)_\eta$, and solving a tridiagonal system of equations along the body surface by a point-Jacobi relaxation.

VIII. Results

Riemann's Problem

The numerical solution of a one-dimensional shock-tube flow is chosen to demonstrate the capability of a nonconservative Euler code to accurately track transient shock waves. In Riemann's problem, a diaphragm separates a perfect gas with different entropy but uniform temperature. With the rupture of the diaphragm, an expansion propagates into the low-entropy gas, whereas a shock wave followed by a contact discontinuity runs into the high-entropy gas (see Liepmann and Roshko[22] for details and the exact solution).

In Fig. 3, the solution with the shock-fitting method is compared with the numerical solution by Steger and Warming[17] and the exact solution. The initial density ratio is 10, and as in Steger and Warming,[17] $\tau/\Delta x = 0.4$, which corresponds to CFL ≈ 0.95. The shock-fitting method yields a crisp representation of the shock, which is in phase with the exactly computed shock. The shock-capturing method by Steger and Warming smears the shock over three mesh intervals and reveals some spurious pre- and postshock oscillations. Both numerical methods smear the contact discontinuity. For the shock-fitting method, this deficiency can be eliminated by explicitly computing, that is, fitting, the contact discontinuity.[1,6] The shock-fitting method also discloses some "wavy" behavior around the transition from the expansion fan into the still undisturbed gas. This slight aberration can be corrected by fitting this gradient discontinuity.[6]

Regular Shock Reflection

This is a standard test case[10,23,24] used to assess a code's accuracy in computing shocks that are not aligned with cell interfaces. This is a crucial issue since, with few exceptions, all multidimensional upwind schemes are constructed with the assumption that the shock normals are parallel to the cell surface normals (see, for instance, Ref. 25).

Results are presented for supersonic flow ($M_\infty = 2.9$) along a nonporous wall with a shock impinging on that wall at an angle of 29 deg. The calculations are started by assuming uniform flow with $M_\infty = 2.9$ everywhere except along the top boundary, where the variables are consistently overspecified from the jump conditions. The comparison of computed and "exact" pressures along the line $y = 0.4$ in Fig. 4 shows again the sharpest possible discrete resolution of both the impinging and reflected shock. Note that the results are "grid-independent": The correct answer is calculated on a uniform grid of just 16 × 6 grid points; this is only 1/16 of the usually[10,23,24] employed 61 × 21 grid points. Figure 5 shows pressure contours and grid lines for computations on three successively refined grids. In all three cases, the shocks are confined to just one mesh interval. (The jagged appearance of the shocks is due to the contouring routine's use of bilinear interpolation between the known nodal values.)

Figure 6 summarizes the convergence performance of the shock-fitting method. The L_2 norm of all residuals is driven to machine zero in 100–200 iterations, depending on the fineness of the grid. The computing time per grid point and per iteration using 64-bit-word arithmetic is 27 μs on the CYBER-205 at NASA Langley, 14 μs on the Cray-2 at NAS, 11 μs on the Cray-2s at NASA Langley, and 7.5 μs on the Cray Y-MP at NAS.

These computing times are obtained on the 61 × 21 grid with the same computer program, which is written in standard Fortran 77 to make it portable. For completeness, it is mentioned that only one processor has been used on any of the Cray computers.

Supersonic Flow Past a Cylinder

This benchmark test[14,23,26] involves computations on a curvilinear, body-fitted grid that is not aligned with a strong detached shock. The results in Figs. 7–9 are computed for $M_\infty = 8.0$. As frequently done,[14,27,28] the calculations are ini-

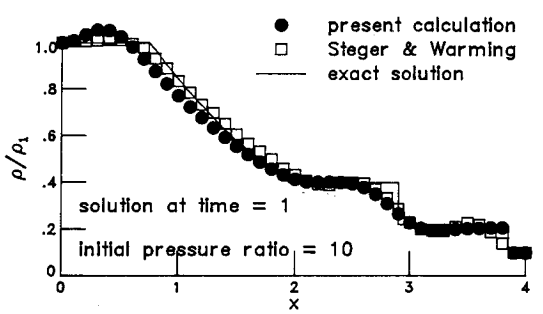

Fig. 3 Solutions for Riemann's problem.

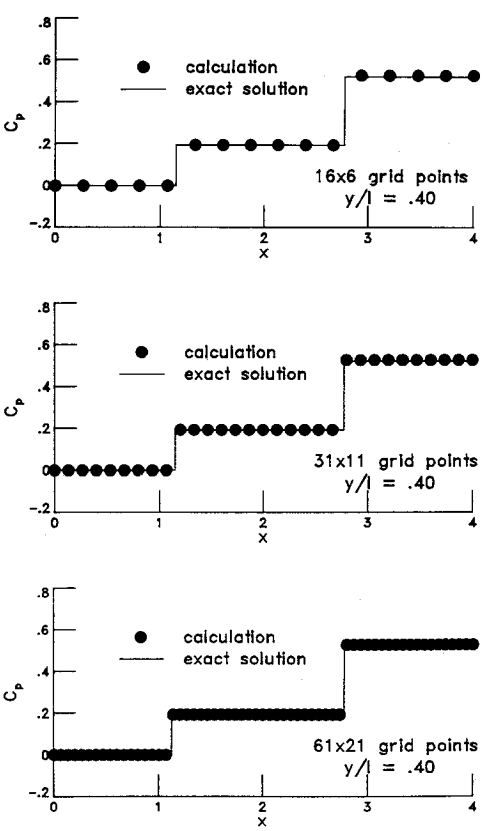

Fig. 4 Pressure coefficients for shock reflection problem.

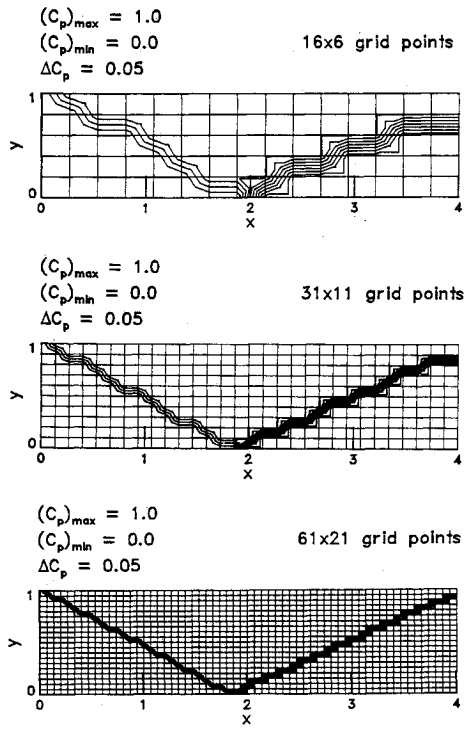

Fig. 5 Pressure distributions for shock reflection problem.

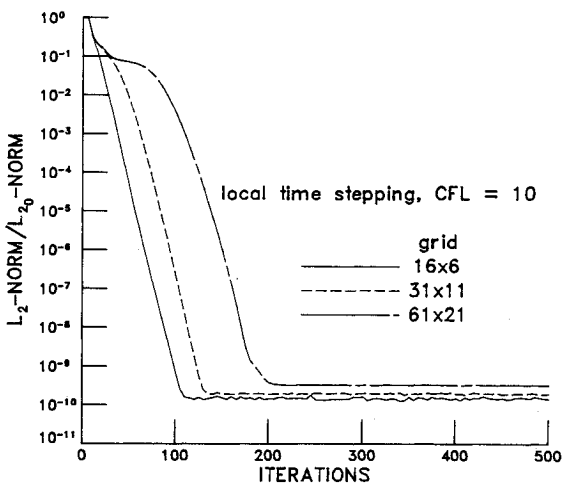

Fig. 6 Convergence summary for shock reflection problem.

tialized by assuming adiabatic flow on the cylinder surface, which implies constant total enthalpy. According to inviscid theory, the body surface is created by a line of constant entropy, with a value equal to that downstream of a normal shock with a shock Mach number equal to M_∞. The initial velocity magnitude along the cylinder surface is determined by assuming a Newtonian pressure distribution[29] in conjunction with the constancy of the total enthalpy. The initial shock shape and shock location are computed using some empirical formulas.[30] Once the Rankine–Hugoniot relations have been applied, the initial flowfield between shock and body surface is computed from linear interpolation. The solutions are calculated without the usual assumption of bilateral symmetry. They demonstrate that the present method is free of producing "carbuncles" or "protrusions" along the stagnation streamline.[31] The surface pressure results compare very well with the "exact" solution by Lyubimov and Rusanov[26] (Fig. 7); so do the shape and location of the fitted shock (Fig. 8). Figure 9 demonstrates that a sharp representation of the shock is achieved on a grid with just 21 × 11 grid points, without having the grid adapted to the shock shape.

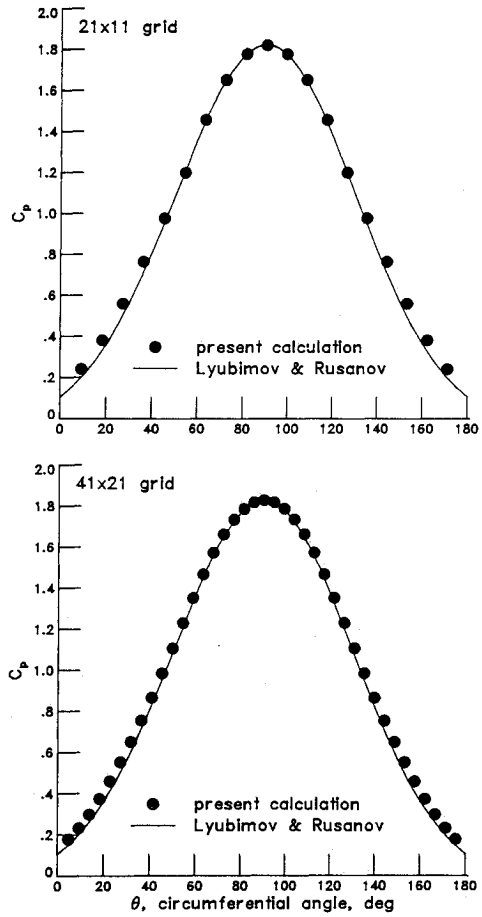

Fig. 7 Surface pressure distributions for detached shock problem.

The convergence summary in Fig. 10 indicates that although the solutions converge rapidly to a steady state, the L_2 norm of all residuals drops only about four orders of magnitude. This limit cycle behavior is due to the crude grids and the subsonic pocket between bow shock and cylinder. Small inaccuracies in the subsonic region trigger a minute, but never damped oscillation of the shock front, which, in turn, prevents the subsonic solution from converging to machine zero.

Transonic Airfoil

Supercritical flow ($M_\infty = 0.8$) past a NACA 0012 airfoil at zero incidence has been chosen to demonstrate the capability of the present code to handle transonic flows with embedded shocks. These solutions are computed on a coarse and a standard C-type mesh. The standard mesh with 161 × 33 grid points has been closely patterned after that used by Pulliam et al.[32] for computing subcritical shockless flows. A coarser grid (81 × 17) has been derived by dropping every other grid point in each coordinate direction. The computational study by Pulliam et al. on transonic flows over the same airfoil has been selected because, with its thorough documentation, it allows for a well-defined comparison between a quite accurate shock-capturing method whose accuracy has been further enhanced by using adapted meshes, with the present method being used on "unadapted" meshes.

For the chosen parameters, an upper and a lower shock develop at $x/C \approx 0.5$, which coincides with the coarsest chordwise resolution of the "subcritical" grids. The spatial step size $\Delta x/C$ in the neighborhood of the fitted shocks is about twice (161 × 33 grid) and four times (81 × 17 grid) as large as in the comparable computations by Pulliam et al.,[32] who employed an adapted 161 × 33 mesh with 16 ξ = const lines clustered along the shock fronts with a regular spacing of $\Delta x/C = 0.01$. Even with this handicap, the surface pressure distributions and the shock locations as predicted by the pre-

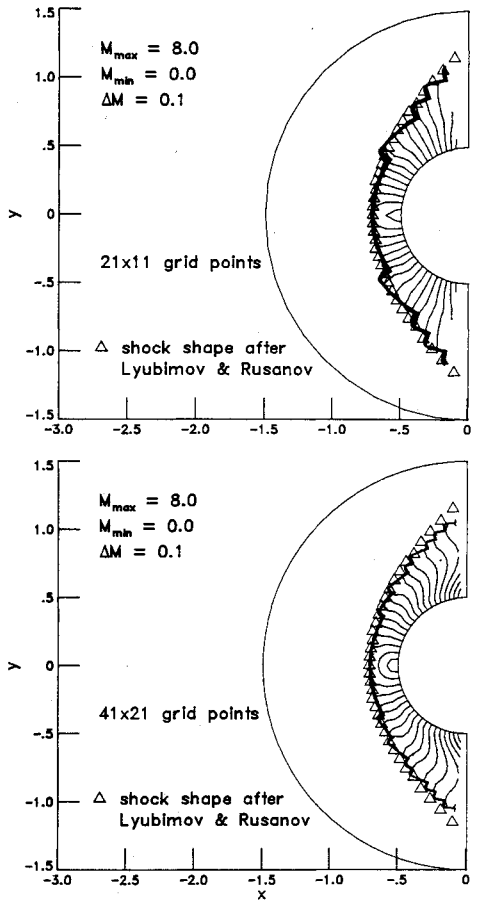

Fig. 8 Mach number distribution for detached shock problem.

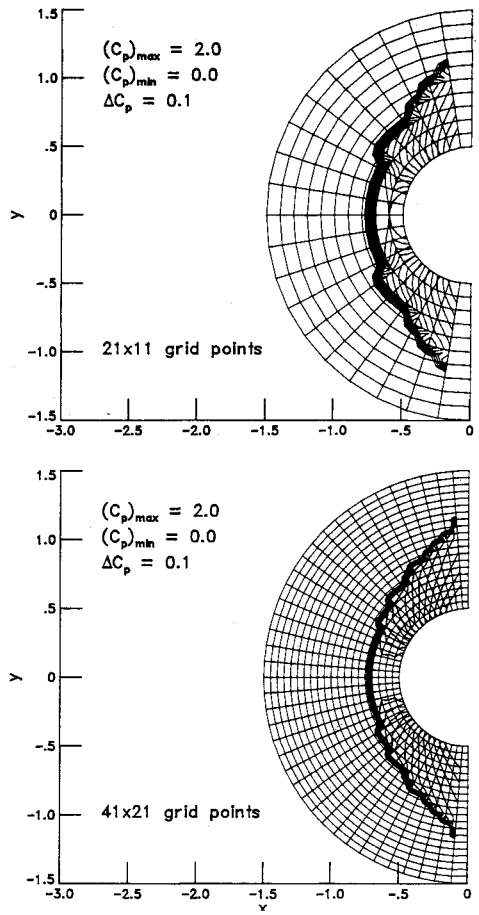

Fig. 9 Pressure distribution for detached shock problem.

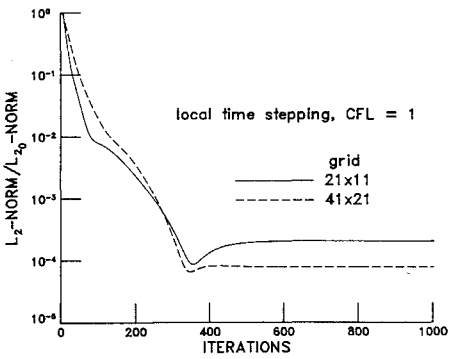

Fig. 10 Convergence summary for detached shock problem.

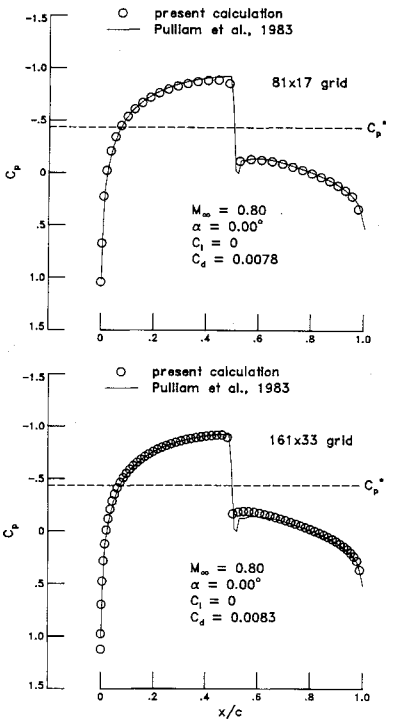

Fig. 11 Surface pressure distributions for supercritical flow past a NACA 0012 airfoil.

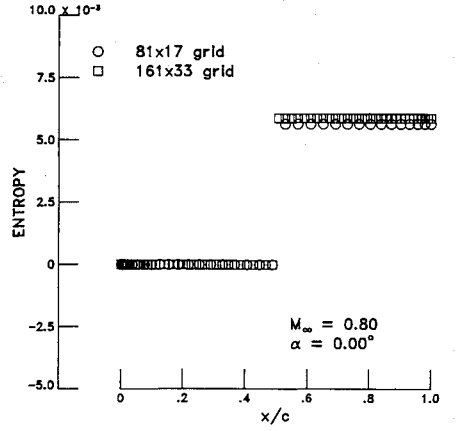

Fig. 12 Surface entropy distributions for supercritical flow past a NACA 0012 airfoil.

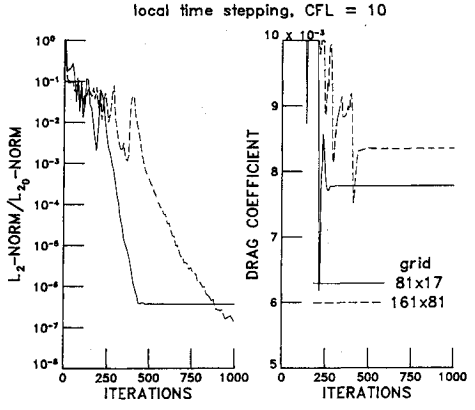

Fig. 13 Convergence summary for supercritical flow past a NACA 0012 airfoil.

sent method on both grids agree quite well with the shock-capturing results on the adapted grid (Fig. 11). (Note that despite their crisp appearance, the captured shocks are still resolved over four mesh intervals.) The surface entropy distributions in Fig. 12 provide another means of demonstrating the accuracy of the present results: 1) Entropy is only generated across the shocks, 2) away from shocks entropy is merely convected and thus it remains at a constant level, and 3) the fitted shocks are completely devoid of any spurious oscillations. As indicated by the convergence summary in Fig. 13, it takes roughly 400 (900) iterations to reduce the L_2 norm of all residuals by about seven orders of magnitude for the coarse (standard) grid case. Starting from freestream conditions, it takes less than 500 iterations to establish asymptotic values for the drag, which differ by just five counts for the two cases.

There are several reasons for some minor differences between the present airfoil results and those published in Ref. 33. First, a sharp trailing edge has been produced not by bevelling (as before), but by extending the chord by roughly 1% as in Lock.[34] Furthermore, different farfield boundary conditions, which are more general than those in Ref. 33, have been used in the present calculations. Finally, the shock parameter [see Eq. (25)] has been redefined such that it no longer requires orthogonal grids.

IX. Conclusions

A floating shock-fitting technique has been devised and combined with a second-order-accurate upwind scheme based on the SCM method, and with a time-implicit, diagonalized AF algorithm. The result is a fast and robust, two-dimensional Euler code that produces accurate results for shocked flows on crude meshes that are not adapted to the shock fronts. Future work will concentrate on implementing a contact-discontinuity fitting capability and an extension to three-dimensional flows.

Acknowledgments

NASA Langley Research Center sponsored this work under Contract NAS1-18585. The contributions by Mohamed M. Hafez, James M. Luckring, Frank Marconi, Gino Moretti, Manuel D. Salas, and, in particular, Michael J. Hemsch in many fruitful discussions are gratefully acknowledged.

References

[1]Salas, M. D., "Shock-Fitting Method for Complicated Two-Dimensional Supersonic Flows," *AIAA Journal,* Vol. 14, No. 5, 1976, pp. 583–588.

[2]Dadone, A., and Moretti, G., "Fast Euler Solver for Transonic Airfoils, Part II: Applications," *AIAA Journal,* Vol. 26, No. 4, 1988, pp. 417–424.

[3]Moretti, G., "Efficient Euler Solver with Many Applications," *AIAA Journal,* Vol. 26, No. 6, 1988, pp. 655–660.

[4]Morton, K. W., and Paisley, M. F., "A Finite Volume Scheme with Shock Fitting for the Steady Euler Equations," *Journal of Computational Physics,* Vol. 8, No. 1, 1989, pp. 168–203.

[5]Lax, P. D., *Hyperbolic Systems of Conservation Laws and the Mathemtical Theory of Shock Waves,* SIAM, Philadelphia, PA, 1972.

[6]Moretti, G., and DiPiano, M., "An Improved Lambda-Scheme for One-Dimensional Flows," NASA CR 3712, Sept. 1983.

[7]Marconi, F., "Supersonic Conical Separation due to Shock Vorticity," *AIAA Journal,* Vol. 22, No. 8, 1984, pp. 1048–1055.

[8]Dadone, A., and Moretti, G., "Fast Euler Solver for Transonic Airfoils, Part I: Theory," *AIAA Journal,* Vol. 26, No. 4, 1988, pp. 409–416.

[9]Moretti, G., "The λ-Scheme," *Computers & Fluids,* Vol. 7, No. 3, Sept. 1979, pp. 191–205.

[10]Davis, S. F., "A Rotationally Biased Upwind Difference Scheme for the Euler Equations," *Journal of Computational Physics,* Vol. 56, No. 1, Oct. 1984, pp. 65–92.

[11]Powell, K. G., and van Leer, B., "A Genuinely Multi-Dimensional Upwind Cell Vertex Scheme for the Euler Equations," AIAA Paper 89-0095, Jan. 1989.

[12]Morton, K. W., and Rudgyard, M. A., "Shock Recovery and the Cell Vertex Scheme for Steady Euler Equations," *Lecture Notes in Physics,* Vol. 323, Springer-Verlag, New York, 1989, pp. 424–428.

[13]Verhoff, A., and O'Neil, P. J., "Accurate, Efficient Prediction Method for Supersonic/Hypersonic Inviscid Flow," AIAA Paper 87-1165, June 1987.

[14]Chakravarthy, S. R., Anderson, D. A., and Salas, M. D., "The Split-Coefficient Matrix Method for Hyperbolic Systems of Gasdynamic Equations," AIAA Paper 80-0268, Jan. 1980.

[15]Pulliam, T. H., and Chaussee, D. S., "A Diagonal Form of an Implicit Approximate-Factorization Algorithm," *Journal of Computational Physics,* Vol. 39, No. 2, 1981, pp. 347–363.

[16]Moretti, G., "A Technique for Integrating Two-Dimensional Euler Equations," *Computers & Fluids,* Vol. 15, No. 1, 1987, pp. 59–75.

[17]Steger, J. L., and Warming, R. F., "Flux Vector Splitting of the Inviscid Gasdynamic Equations with Application to Finite-Difference Methods," *Journal of Computational Physics,* Vol. 40, No. 2, April 1981, pp. 263–293.

[18]Steger, J. L., "Implicit Finite-Difference Simulation of Flow About Arbitrary Two-Dimensional Geometries," *AIAA Journal,* Vol. 16, No. 7, 1978, pp. 679–686.

[19]Pulliam, T. H., and Steger, J. L., "Recent Improvements in Efficiency, Accuracy, and Convergence for Implicit Approximate Factorization Algorithms," AIAA Paper 85-0360, Jan. 1985.

[20]Thomas, J. L., and Salas, M. D., "Farfield Boundary Conditions for Transonic Lifting Solutions to the Euler Equations," *AIAA Journal,* Vol. 24, No. 7, 1986, pp. 1074–1080.

[21]Barton, J. T., and Pulliam, T. H., "Airfoil Computation at High Angles of Attack, Inviscid and Viscous Phenomena," *AIAA Journal,* Vol. 24, No. 5, 1986, pp. 705–714.

[22]Liepmann, H. W., and Roshko, A., *Elements of Gasdynamics,* Wiley, New York, 1957.

[23]Chakravarthy, S. R., and Osher, S., "High Resolution Applications of the Osher Upwind Scheme for the Euler Equations," AIAA Paper 83-1943, June 1983.

[24]Thomas, J. L., Walters, R. W., Rudy, D. H., and Swanson, R. C., "Upwind Relaxation Algorithms for Euler/Navier-Stokes Equations," NASA CP 2398, Vol. I, April 1985, pp. 89–108.

[25]Roe, P. L., "Characteristic-Based Schemes for the Euler Equations," *Annual Review of Fluid Mechanics,* Vol. 18, 1986, pp. 337–365.

[26]Lyubimov, A. N., and Rusanov, V. V., "Gas Flows Past Blunt Bodies," NASA TT-F715, Feb. 1973.

[27]Kutler, P., Chakravarthy, S. R., and Lombard, C. P., "Supersonic Flow over Ablated Nosetips Using an Unsteady Implicit Numerical Procedure," AIAA Paper 78-213, Jan. 1978.

[28]Kutler, P., Pedelty, J. A., and Pulliam, T. H., "Supersonic Flow over Ablated Nosetips Using an Unsteady Implicit Numerical Procedure," AIAA Paper 80-0063, Jan. 1980.

[29]Hayes, W. D., and Probstein, R. F., *Hypersonic Flow Theory,* Academic Press, New York, 1959.

[30]Billig, S. F., "Shock-Wave Shapes Around Spherical- and Cylindrical-Nosed Bodies," *Journal of Spacecraft and Rockets,* Vol. 4, 1967, pp. 822, 823.

[31]Peery, K. M., and Imlay, S. T., "Blunt-Body Flow Simulations," AIAA Paper 88-2904, July 1988.

[32]Pulliam, T. H., Jespersen, D. C., and Childs, R. E., "An Enhanced Version of an Implicit Code for the Euler Equations," AIAA Paper 83-0344, Jan. 1983.

[33]Hartwich, P. M., "Fresh Look at Floating Shock Fitting," AIAA Paper 90-0108, Jan. 1990.

[34]Lock, R. C., "Test Cases for Numerical Methods in Two-Dimensional Transonic Flows," AGARD Rept. 575, 1970.