

Implicit TVD Schemes for Hyperbolic Conservation Laws in Curvilinear Coordinates

H. C. Yee*

NASA Ames Research Center, Moffett Field, California
and

A. Harten†

Tel Aviv University, Tel Aviv, Israel

A one-parameter family of explicit and implicit upwind second-order-accurate, total variation diminishing (TVD) schemes has been developed by Harten. These TVD schemes have the property of not generating spurious oscillations when applied to one-dimensional nonlinear scalar hyperbolic conservation laws and constant coefficient hyperbolic systems. The goal of this work is to extend these methods to the multidimensional hyperbolic conservation laws in curvilinear coordinates. Various ways of linearizing the implicit operator and solution strategies to improve the computation efficiency of the implicit algorithm are discussed. Numerical experiments with some AGARD test cases for steady-state airfoil calculations show that the proposed linearized implicit upwind TVD schemes are quite robust and accurate.

I. Introduction

IN recent years, many second-order-accurate shock-capturing finite-difference schemes for the computation of the Euler equations of gasdynamics have been developed. See, for example, the recent proceedings in Refs. 1–3. Of special interest are the methods that generate nonoscillatory but sharp approximations to shocks and contact discontinuities.^{4–12} This interest stems from the fact that even with the recent advances in high-speed supercomputing, grid generation, automatic adaptive grid procedures, etc., the lack of robust and accurate numerical schemes is a major stumbling block for the success of computational fluid dynamics. Most of these schemes^{4–12} are very different in form, methodology, and design principle. However, from the standpoint of numerical analysis, these schemes are total variation diminishing (TVD) for nonlinear scalar hyperbolic conservation laws and for constant coefficient hyperbolic systems. The notion of TVD schemes was introduced by Harten.^{4–5} Some of these methods can also be viewed as three-point central difference schemes with a “smart” numerical dissipation or smoothing mechanism. “Smart” here means automatic feedback mechanism to control the amount of numerical dissipation for nonlinear problems. In general, TVD schemes can be divided into two categories, namely, upwind and symmetric TVD schemes. A way of distinguishing an upwind from a symmetric TVD scheme is that the numerical dissipation term corresponding to an upwind TVD scheme is upwind-weighted^{4–9} as opposed to the numerical dissipation term corresponding to a symmetric TVD scheme that is centered.^{10–12}

Harten’s method of constructing high-resolution TVD schemes involves starting with a first-order TVD scheme and applying it to a modified flux. The modified flux is chosen

so that the scheme is second-order at regions of smoothness and first-order at points of extrema. This technique is sometimes referred to as the modified flux approach. Although the scheme is an upwind scheme, it is written in a symmetric form; i.e., central difference plus an appropriate numerical dissipation term. This special form is especially advantageous for systems of higher than one space dimension. It results in less storage and a smaller operation count than its upwind form.¹³ The modified flux approach is relatively simple to understand and easy to implement into a new or existing computer code. One can modify a standard three-point central difference code by simply changing the conventional numerical dissipation term into the one designed for the TVD scheme. However, for non-Cartesian grids, care must be taken to preserve the freestream. A formulation closer to finite volume is more desirable.

In Ref. 14, a preliminary study was completed on the implicit TVD scheme for a two-dimensional gasdynamics problem in a Cartesian coordinate. It was found that further improvement in computation efficiency and convergence rate is required for practical application.

The objective of this paper is to extend this TVD scheme (via the modified flux approach) to generalized coordinate systems and to discuss the various solution strategies for the implicit TVD schemes for more efficient two-dimensional steady-state applications. Numerical experiments with some of the 1984 AGARD Fluid Dynamics Panel Working Group 07 airfoil test cases¹⁵ are included.

Because of space limitations, the material was divided into two separate papers. In the present paper, applications are stressed rather than detailed derivations of the various linearized forms of the implicit TVD schemes. Interested readers should refer to our companion paper¹⁶ for more details.

II. Description of the Algorithm for a System of Conservation Laws

Before going further, it should be emphasized that all the second-order TVD schemes are constructed so that no spurious oscillations are generated for one-dimensional nonlinear scalar hyperbolic conservation laws and constant coefficient hyperbolic systems. None of the theory says anything about nonlinear systems or two-dimensional scalar hyperbolic conservation laws. But, in practice, it is straightforward to extend the scheme formally to one- or two-

Presented as Paper 85-1513 at the AIAA Seventh Computational Fluid Dynamics Conference, Cincinnati, OH, July 15–17, 1985; received Aug. 19, 1985; revision submitted April 8, 1986. Copyright © 1986 American Institute of Aeronautics and Astronautics, Inc. No copyright is asserted in the United States under Title 17, U.S. Code. The U.S. Government has a royalty-free license to exercise all rights under the copyright claimed herein for Governmental purposes. All other rights are reserved by the copyright owner.

*Research Scientist, Computational Fluid Dynamics Branch.

†Associate Professor, School of Mathematical Sciences; also Visiting Professor, University of California, Los Angeles, CA.

dimensional nonlinear hyperbolic systems. Therefore, for the rest of the paper it is understood that the properties of all the schemes under discussion are for one-dimensional nonlinear scalar hyperbolic conservation laws and one-dimensional constant coefficient hyperbolic systems. The schemes are then formally extended to one- or two-dimensional systems of conservation laws and are evaluated by numerical experiments.

Extension of the scalar TVD scheme to systems of conservation laws can be accomplished by defining at each point a "local" system of characteristic fields, and then applying the scheme to each of the m scalar characteristic equations. Here m is the dimension of the hyperbolic system. Extension of these schemes to system cases is not unique; see Refs. 13 and 16 for more details. The formulation described here is valid for both two- and three-dimensional systems of conservation laws. Only the two-dimensional case will be described. For three-dimensional formulations, one has only to add an extra dimension and the corresponding numerical flux.

The notion of TVD schemes, including formulation and extension to system cases (in uniform Cartesian grids), can be found in Refs. 14 and 16. Here the extension of the implicit second-order-accurate TVD scheme for hyperbolic systems of conservation laws in curvilinear coordinates is described.

Consider a two-dimensional system of hyperbolic conservation laws

$$\frac{\partial Q}{\partial t} + \frac{\partial F(Q)}{\partial x} + \frac{\partial G(Q)}{\partial y} = 0 \quad (1)$$

Here Q , $F(Q)$, and $G(Q)$ are column vectors of m components.

A generalized coordinate transformation of the form $\xi = \xi(x, y)$ and $\eta = \eta(x, y)$, which maintains the strong conservation law form of Eq. (1) is given by

$$\frac{\partial \hat{Q}}{\partial t} + \frac{\partial \hat{F}(\hat{Q})}{\partial \xi} + \frac{\partial \hat{G}(\hat{Q})}{\partial \eta} = 0 \quad (2)$$

where $\hat{Q} = Q/J$, $\hat{F} = (\xi_x F + \xi_y G)/J$, $\hat{G} = (\eta_x F + \eta_y G)/J$, and $J = \xi_x \eta_y - \xi_y \eta_x$, the Jacobian transformation. Let $A = \partial F / \partial Q$ and $B = \partial G / \partial Q$; then the Jacobians \hat{A} and \hat{B} of \hat{F} and \hat{G} can be written as

$$\hat{A} = (\xi_x A + \xi_y B) \quad (3a)$$

$$\hat{B} = (\eta_x A + \eta_y B) \quad (3b)$$

Let the eigenvalues of \hat{A} be

$$(a_\xi^1, a_\xi^2, \dots, a_\xi^m) \quad (4a)$$

and the eigenvalues of \hat{B} be

$$(a_\eta^1, a_\eta^2, \dots, a_\eta^m) \quad (4b)$$

Denote R_ξ and R_η as the matrices whose columns are eigenvectors of \hat{A} and \hat{B} , and denote R_ξ^{-1} and R_η^{-1} as the inverses of R_ξ and R_η .

Let the grid spacing be denoted by $\Delta\xi$ and $\Delta\eta$ such that $\xi = j\Delta\xi$ and $\eta = k\Delta\eta$. Denote $Q_{j+\frac{1}{2},k}$ as some symmetric average of $Q_{j,k}$ and $Q_{j+1,k}$ (for example, $Q_{j+\frac{1}{2},k} = 0.5*(Q_{j+1,k} + Q_{j,k})$, or the Roe's average¹⁷ for gasdynamics, see Sec. IV for a discussion). Let $a_{j+\frac{1}{2}}^1$, $R_{j+\frac{1}{2}}$, and $R_{j+\frac{1}{2}}^{-1}$ denote the quantities a_ξ^1 , R_ξ , and R_ξ^{-1} related to \hat{A} evaluated at $Q_{j+\frac{1}{2},k}$. Similarly, let $a_{k+\frac{1}{2}}^1$, $R_{k+\frac{1}{2}}$, and $R_{k+\frac{1}{2}}^{-1}$ denote the quantities a_η^1 , R_η , and R_η^{-1} related to \hat{B} evaluated at $Q_{j,k+\frac{1}{2}}$.

Define

$$\alpha_{j+\frac{1}{2}} = R_{j+\frac{1}{2}}^{-1} \frac{Q_{j+1,k} - Q_{j,k}}{0.5*(J_{j+1,k} + J_{j,k})} \quad (5a)$$

as the difference of the characteristic variables in the local ξ direction, and define

$$\alpha_{k+\frac{1}{2}} = R_{k+\frac{1}{2}}^{-1} \frac{Q_{j,k+1} - Q_{j,k}}{0.5*(J_{j,k+1} + J_{j,k})} \quad (5b)$$

as the difference of the characteristic variables in the local η direction. The $J_{j,k}$ is the Jacobian transformation evaluated at $(j\Delta\xi, k\Delta\eta)$. The averaged Jacobians are used here in order to preserve the freestream. Here, for example, the ξ direction characteristic variables W_ξ with respect to the state \hat{Q} is defined by

$$W_\xi = R_\xi^{-1} \hat{Q} \quad (6)$$

With the above notation, a one-parameter family of TVD schemes can be written as

$$\begin{aligned} \hat{Q}_{j,k}^{n+1} + \lambda^\xi \theta [\tilde{F}_{j+\frac{1}{2},k}^{n+1} - \tilde{F}_{j-\frac{1}{2},k}^{n+1}] + \lambda^\eta \theta [\tilde{G}_{j,k+\frac{1}{2}}^{n+1} - \tilde{G}_{j,k-\frac{1}{2}}^{n+1}] \\ = \hat{Q}_{j,k}^n - \lambda^\xi (1-\theta) [\tilde{F}_{j+\frac{1}{2},k}^n - \tilde{F}_{j-\frac{1}{2},k}^n] \\ - \lambda^\eta (1-\theta) [\tilde{G}_{j,k+\frac{1}{2}}^n - \tilde{G}_{j,k-\frac{1}{2}}^n] \end{aligned} \quad (7a)$$

where θ is a parameter, $\lambda^\xi = \Delta t / \Delta\xi$, and $\lambda^\eta = \Delta t / \Delta\eta$. A particular form of the numerical flux function $\tilde{F}_{j+\frac{1}{2},k}$ can be expressed as

$$\tilde{F}_{j+\frac{1}{2},k} = \frac{1}{2} [\hat{F}_{j,k} + \hat{F}_{j+1,k} + R_{j+\frac{1}{2}} \Phi_{j+\frac{1}{2}}] \quad (7b)$$

where the elements of the $\Phi_{j+\frac{1}{2}}$ denoted by $\phi_{j+\frac{1}{2}}^l$, $l=1, \dots, m$ are

$$\phi_{j+\frac{1}{2}}^l = \frac{1}{2} \psi(a_{j+\frac{1}{2}}^l)(g_j^l + g_{j+1}^l) - \psi(a_{j+\frac{1}{2}}^l + \gamma_{j+\frac{1}{2}}^l) \alpha_{j+\frac{1}{2}}^l \quad (7c)$$

with

$$g_j^l = S \cdot \max[0, \min(|\alpha_{j+\frac{1}{2}}^l|, S \cdot \alpha_{j-\frac{1}{2}}^l)] \quad (7d)$$

$$S = \text{sign}(\alpha_{j+\frac{1}{2}}^l)$$

$$\begin{aligned} \psi(z) &= |z| & |z| &\geq \epsilon \\ &= (z^2 + \epsilon^2)/2\epsilon & |z| &< \epsilon \end{aligned} \quad (7e)$$

Here ϵ is a small positive number (see Ref. 18 for a formula of ϵ), and

$$\begin{aligned} \gamma_{j+\frac{1}{2}}^l &= \frac{1}{2} \psi(a_{j+\frac{1}{2}}^l) = (g_{j+1}^l - g_j^l) / \alpha_{j+\frac{1}{2}}^l & \alpha_{j+\frac{1}{2}}^l &\neq 0 \\ &= 0 & \alpha_{j+\frac{1}{2}}^l &= 0 \end{aligned} \quad (7f)$$

where $\alpha_{j+\frac{1}{2}}^l$ are elements of Eq. (5a). Similarly, we can define the numerical flux $\tilde{G}_{j,k+\frac{1}{2}}$.

This one-parameter family of schemes contains implicit as well as explicit schemes. When $\theta=0$, Eq. (7) is an explicit method; when $\theta \neq 0$, Eq. (7) is an implicit scheme. For example, if $\theta = \frac{1}{2}$, the time differencing is the trapezoidal formula, and scheme (7) is second-order in time and space. If $\theta=1$, the time differencing is the backward Euler method and scheme (7) is first-order in time but second-order in space. Second-order accuracy in time for $\theta=1$ can be achieved by a slight modification of Eqs. (7c) and (7f) (see Ref. 16 for details). Although second-order accuracy can be achieved in space and time for $\theta=1$, the resulting scheme is

good only for time-accurate solutions since steady-state solutions are dependent on the time step Δt . Equations (7c) and (7f) are chosen mainly for steady-state applications. It has the property that the steady-state solution is independent of the time step Δt .

III. Linearized Forms of the Algorithm

In order to solve for \hat{Q}^{n+1} , a set of nonlinear algebraic equations needs to be solved. For computational efficiency, consider the following solution strategy. First, linearize the implicit operator in two ways; then, construct an alternating direction implicit (ADI) form for the linearized implicit algorithm.

Here two linearized forms of Eq. (7) are proposed. The first method will preserve the conservative form, but the resulting scheme may no longer be unconditionally TVD. This method is referred to as the linearized conservative implicit (LCI) form. The second method will destroy the conservative property (in time) but preserve the unconditionally TVD property. The latter method is referred to as the linearized nonconservative implicit (LNI) form. For a detailed derivation in Cartesian coordinates, see Ref. 16.

Linearized Conservative Implicit (LCI) Form

The LCI form of Eq. (7) in Delta form can be written as

$$\begin{aligned} & [I + \theta(\lambda^\xi H_{j+\frac{1}{2},k}^\xi - \lambda^\xi H_{j-\frac{1}{2},k}^\xi + \lambda^\eta H_{j,k+\frac{1}{2}}^\eta - \lambda^\eta H_{j,k-\frac{1}{2}}^\eta)] \\ & \times (\hat{Q}^{n+1} - \hat{Q}^n) = -\lambda^\xi [\tilde{F}_{j+\frac{1}{2},k}^\eta - \tilde{F}_{j-\frac{1}{2},k}^\eta] \\ & - \lambda^\eta [\tilde{G}_{j,k+\frac{1}{2}}^\eta - \tilde{G}_{j,k-\frac{1}{2}}^\eta] \end{aligned} \quad (8a)$$

where

$$H_{j+\frac{1}{2},k}^\xi = \frac{1}{2} [\hat{A}_{j+1,k}^\xi + \Omega_{j+\frac{1}{2},k}^\xi] \quad (8b)$$

$$H_{j,k+\frac{1}{2}}^\eta = \frac{1}{2} [\hat{B}_{j,k+1}^\eta + \Omega_{j,k+\frac{1}{2}}^\eta] \quad (8c)$$

with \hat{A} and \hat{B} the Jacobians of the fluxes \hat{F} and \hat{G} , and

$$\Omega_{j+\frac{1}{2},k}^\xi = \left(R_\xi \text{diag} [\beta^l - \psi(a^l + \gamma^l)] R_\xi^{-1} \right)_{j+\frac{1}{2}} \Delta_{j+\frac{1}{2}} \quad (8d)$$

$$\Omega_{j,k+\frac{1}{2}}^\eta = \left(R_\eta \text{diag} [\beta^l - \psi(a^l + \gamma^l)] R_\eta^{-1} \right)_{k+\frac{1}{2}} \Delta_{k+\frac{1}{2}} \quad (8e)$$

Here $\text{diag}(z^l)$ denotes a diagonal matrix with diagonal elements z^l . The $\beta_{j+\frac{1}{2}}^l$ and $\beta_{k+\frac{1}{2}}^l$ are defined as

$$\beta_{j+\frac{1}{2}}^l = \frac{(g_j^l + g_{j+1}^l)}{\alpha_{j+\frac{1}{2}}^l}, \quad \beta_{k+\frac{1}{2}}^l = \frac{(g_k^l + g_{k+1}^l)}{\alpha_{k+\frac{1}{2}}^l} \quad (8f)$$

The nonstandard notation

$$\begin{aligned} & H_{j+\frac{1}{2},k}^\xi (\hat{Q}^{n+1} - \hat{Q}^n) = \frac{1}{2} [\hat{A}_{j+1,k}^\eta (\hat{Q}^{n+1} - \hat{Q}^n)_{j+1,k} \\ & + (\Omega_{j+\frac{1}{2},k}^\eta)^n (\hat{Q}^{n+1} - \hat{Q}^n)] \end{aligned} \quad (8g)$$

is used and the difference operator $\Omega_{j+\frac{1}{2},k}^\xi$ is defined in Eq. (8d). The operator $\Delta_{j+\frac{1}{2}}$, operating on \hat{Q} , means $\Delta_{j+\frac{1}{2}} \hat{Q} = \hat{Q}_{j+1,k} - \hat{Q}_{j,k}$.

Notice that in each coordinate direction, Eq. (8) is a spatially second-order-accurate, five-point scheme and yet the iteration matrix associated with Eq. (8) in that direction is a block tridiagonal matrix. This linearized construction is nontrivial since the matrix associated with a five-point stencil scheme is normally a block pentadiagonal matrix. A block tridiagonal matrix is obtained because of the special linearized procedure.

Linearized Nonconservative Implicit (LNI) Form

The LNI form of Eq. (7) in Delta form can be written as

$$\begin{aligned} & [I - \lambda^\xi \theta K_{j+\frac{1}{2},k}^- \Delta_{j+\frac{1}{2}} + \lambda^\xi \theta K_{j-\frac{1}{2},k}^+ \Delta_{j-\frac{1}{2}} - \lambda^\eta \theta L_{j,k+\frac{1}{2}}^- \Delta_{k+\frac{1}{2}} \\ & + \lambda^\eta \theta L_{j,k-\frac{1}{2}}^+ \Delta_{k-\frac{1}{2}}] (\hat{Q}^{n+1} - \hat{Q}^n) \\ & = -\lambda^\xi [\tilde{F}_{j+\frac{1}{2},k}^\eta - \tilde{F}_{j-\frac{1}{2},k}^\eta] - \lambda^\eta [\tilde{G}_{j,k+\frac{1}{2}}^\eta - \tilde{G}_{j,k-\frac{1}{2}}^\eta] \end{aligned} \quad (9a)$$

where

$$K_{j+\frac{1}{2},k}^\pm = \left(R_\xi \text{diag} (C_\xi^\pm) R_\xi^{-1} \right)_{j+\frac{1}{2},k}^n \quad (9b)$$

$$L_{j,k+\frac{1}{2}}^\pm = \left(R_\eta \text{diag} (C_\eta^\pm) R_\eta^{-1} \right)_{j,k+\frac{1}{2}}^n \quad (9c)$$

$$(C_\xi^\pm)_{j+\frac{1}{2},k}^\eta = \frac{1}{2} [\psi(a_\xi^l + \gamma_\xi^l) \pm (a_\xi^l + \gamma_\xi^l)]_{j+\frac{1}{2},k}^\eta \quad (9d)$$

$$(C_\eta^\pm)_{j,k+\frac{1}{2}}^\eta = \frac{1}{2} [\psi(a_\eta^l + \gamma_\eta^l) \pm (a_\eta^l + \gamma_\eta^l)]_{j,k+\frac{1}{2}}^\eta \quad l=1, \dots, m \quad (9e)$$

Again, in each coordinate direction, Eq. (9) is a spatially second-order-accurate, five-point scheme and yet the iteration matrix associated with Eq. (9) in that direction is a block tridiagonal matrix. Note that because of this specific linearization, Eq. (9) is conservative in space but not in time. Consequently, Eq. (9) is suitable only for steady-state applications.

Simplification for Steady-State Applications

To calculate Eqs. (8b) and (8c) or (9b) and (9c) at every time step is quite costly. For steady-state applications, Eqs. (8) and (9) can be simplified even more, since one only wants the scheme to be second-order after it reaches steady state. The time integration and the entire implicit operator can be viewed as a relaxation procedure for the steady-state solution.

One way to simplify Eq. (8) is to use a spatially first-order implicit operator; i.e., by redefining Eqs. (8d) and (8e) as

$$\Omega_{j+\frac{1}{2},k}^\xi = \left(R_\xi \text{diag} [-\psi(a^l)] R_\xi^{-1} \right)_{j+\frac{1}{2}} \Delta_{j+\frac{1}{2}} \quad (10a)$$

$$\Omega_{j,k+\frac{1}{2}}^\eta = \left(R_\eta \text{diag} [-\psi(a^l)] R_\eta^{-1} \right)_{k+\frac{1}{2}} \Delta_{k+\frac{1}{2}} \quad (10b)$$

The computation can be reduced even more if $\Omega_{j+\frac{1}{2},k}^\xi$ and $\Omega_{j,k+\frac{1}{2}}^\eta$ are simplified to diagonal matrices. For example, one can again redefine Eqs. (10a) and (10b) as

$$\Omega_{j+\frac{1}{2},k}^\xi = \left(\text{diag} [-\max_l \psi(a^l)] \right)_{j+\frac{1}{2}} \Delta_{j+\frac{1}{2}} \quad (11a)$$

$$\Omega_{j,k+\frac{1}{2}}^\eta = \left(\text{diag} [-\max_l \psi(a^l)] \right)_{k+\frac{1}{2}} \Delta_{k+\frac{1}{2}} \quad (11b)$$

From here on, algorithm (8a-8c), together with Eq. (11) is referred to as the linearized conservative diagonal form. Similarly, Eqs. (9b) and (9c) can be simplified by setting $\gamma=0$ in Eqs. (9d) and (9e).

ADI Form

Even with the above simplifications, it is still very costly to solve the two-dimensional difference equations (8) or (9). An ADI form of Eqs. (8) and (9) will be adopted, which can be

expressed as

$$[I + \lambda^\xi \theta H_{j+\frac{1}{2},k}^\xi - \lambda^\xi \theta H_{j-\frac{1}{2},k}^\xi] D^* \\ = -\lambda^\xi [\tilde{F}_{j+\frac{1}{2},k}^\eta - \tilde{F}_{j-\frac{1}{2},k}^\eta] - \lambda^\eta [\tilde{G}_{j,k+\frac{1}{2}}^\eta - \tilde{G}_{j,k-\frac{1}{2}}^\eta] \quad (12a)$$

$$[I + \lambda^\eta \theta H_{j,k+\frac{1}{2}}^\eta - \lambda^\eta \theta H_{j,k-\frac{1}{2}}^\eta] D = D^* \quad (12b)$$

$$\hat{Q}^{n+1} = \hat{Q}^n + D \quad (12c)$$

and

$$[I - \lambda^\xi \theta K_{j+\frac{1}{2},k}^- \Delta_{j+\frac{1}{2}} + \lambda^\xi \theta K_{j-\frac{1}{2},k}^+ \Delta_{j-\frac{1}{2}}] D^* \\ = -\lambda^\xi [\tilde{F}_{j+\frac{1}{2},k}^\eta - \tilde{F}_{j-\frac{1}{2},k}^\eta] - \lambda^\eta [\tilde{G}_{j,k+\frac{1}{2}}^\eta - \tilde{G}_{j,k-\frac{1}{2}}^\eta] \quad (13a)$$

$$[I - \lambda^\eta \theta L_{j,k+\frac{1}{2}}^- \Delta_{k+\frac{1}{2}} + \lambda^\eta \theta L_{j,k-\frac{1}{2}}^+ \Delta_{k-\frac{1}{2}}] D = D^* \quad (13b)$$

$$\hat{Q}^{n+1} = \hat{Q}^n + D \quad (13c)$$

A numerical experiment for a one-dimensional gasdynamics problem shows that the LNI form (9) with $\gamma \neq 0$ or $\gamma = 0$ on the left-hand side has a fairly rapid convergence rate and gives good shock resolution. However, numerical experiments with Eq. (13)¹⁴ show that the ADI construction of the LNI form does not have a good convergence rate. One possible way of solving Eq. (9) is by a different type of relaxation method. This will be the subject of a future investigation. The rest of the paper will concentrate on the linearized conservative implicit form (12) only.

Observe that Eq. (12) together with Eq. (11) are the original Beam and Warming¹⁹ algorithm if $\Omega_{j+\frac{1}{2},k}^x = \Omega_{j,k+\frac{1}{2}}^y = 0$ in Eqs. (8b) and (8c) and $\Phi_{j+\frac{1}{2}} R_{j+\frac{1}{2}}$ in Eq. (7b) is replaced by the conventional fourth-order dissipation term. Therefore, the implementation of this ADI scheme into an existing central difference code (such as the code based on the Beam and Warming algorithm) is relatively simple. All one has to do is add the extra matrices $\Omega_{j+\frac{1}{2},k}^x$ and $\Omega_{j,k+\frac{1}{2}}^y$ for the implicit operator and use a more sophisticated dissipation term $\Phi_{j+\frac{1}{2}} R_{j+\frac{1}{2}}$ for the explicit operator. For the case of the linearized conservative diagonal form (12) together with Eq. (11), no extra work is involved on the implicit operator since Ω^x and Ω^y in Eq. (11) are diagonal matrices with equal elements and can be saved while computing for the right-hand side.

From our numerical experiments with the NACA-0012 airfoil steady-state calculations, the linearized conservative diagonal ADI form is the most efficient scheme (in terms of CPU time) among the various proposed linearized methods for the case of $\theta = 1$. No comparison has been made for time-accurate calculations or for any other values of θ . In the next section some numerical results for the linearized conservative ADI algorithm (12), together with Eq. (11), with $\theta = 1$ (the backward Euler time differencing) will be shown.

IV. Numerical Results for the Euler Equations

The results here utilized a coordinate transformation transforming a general curvilinear physical space into a rectangle with uniform spacing of unit length. For the airfoil calculations, the actual geometry is mapped onto the computational rectangle such that all the boundary surfaces are edges of the rectangle. A C-mesh topology in which a branch cut (wake cut) is employed at the trailing edge of the airfoil is used for all the studies.

Treatment of the boundary condition is similar to Pulliam and Steger.²⁰ All the numerical boundary conditions are treated explicitly. At the airfoil surface the tangency condition must be satisfied, requiring that the velocity in the normal direction be zero. The tangential velocity is obtained at the body surface through linear extrapolation. The pressure on the body surface is obtained from the normal momentum equation. For steady uniform incoming flow, freestream

stagnation enthalpy H_0 is held constant along the body. Using the equation for enthalpy $H_0 = (e + p)/\rho$ and the computed velocities and pressure, a value of density is obtained at the body. The total energy e is then obtained from the equation of state. Along the wake cut in the C-mesh, averaging is used to provide a continuous flow variable. All the upstream boundaries are fixed to freestream values. Since this is a five-point scheme in each coordinate direction, the values g_j^l and g_k^l of Eq. (7d) are obtained by zeroth-order extrapolation at the boundaries.

Since the actual grids have widely varying cell sizes, a space-varying Δt similar to Pulliam and Steger²⁰ is used as a vehicle to improve the convergence rate. This particular form of the space-varying time step is a purely geometric variation of the time step, and it is of the form

$$\Delta t = \bar{\Delta t} / (1 + \sqrt{J}) \quad (14)$$

where $\bar{\Delta t}$ is the fixed time step at input and J is the Jacobian transformation.

The ϵ in Eq. (7) is set equal to 0.125 for all cases. Numerical experiments show that the solutions are insensitive to the value of ϵ between 0.06 and 0.25. A more sophisticated ϵ has been proposed in Ref. 18. Computational experiments show no visible improvement in efficiency or accuracy in using the more sophisticated formula while more computations are required. No artificial compression term, as discussed in Ref. 14, is needed in all the airfoil calculations.

The cases considered here are the NACA-0012 airfoil with 1) $M_\infty = 0.8$, $\alpha = 1.25$, 2) $M_\infty = 0.85$, $\alpha = 1.00$, 3) $M_\infty = 0.95$, $\alpha = 0.0$, and 4) $M_\infty = 1.2$, $\alpha = 7.0$. Here M_∞ is the freestream Mach number and α is the angle of attack. These are four of the cases considered in the AGARD Fluid Dynamics Panel Working Group 07.¹⁵

To show the accuracy of the scheme, we use a 249×41 C-grid with no special clusterings on the upper or lower surface near the vicinity of the shocks. Figure 1 shows the grid distribution around the airfoil. The outer boundary is 24 chord lengths away from the body. Each case was initialized with a uniform freestream flow at the prescribed Mach number and angle of attack and used the same grid as shown in Fig. 1. Figures 2-5 show the pressure coefficient distributions, Mach contours, and pressure contours for all four cases. The plus symbol on the pressure coefficient plots is used to indicate the computed values. The solid (upper surface) and dashed (lower surface) lines are just connectors between grid points. The value C_p^* in all the pressure coefficient figures indicates the critical pressure coefficient. One can see in all cases that shocks can be captured within 1-2 grid points. When the same cases are run on the FLO52R

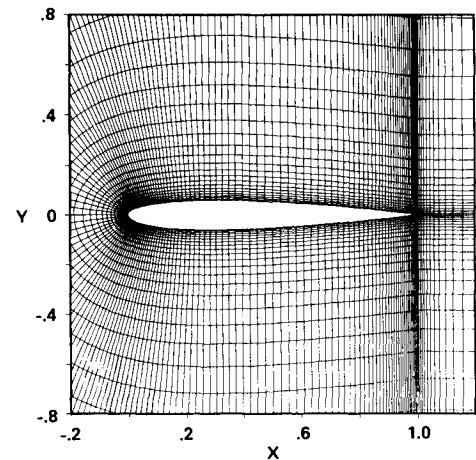


Fig. 1 The 249×41 C-grid for the NACA-0012 airfoil.

code of Jameson²¹ and the improved ARC2D code (version 150) of Pulliam and Steger,²⁰ 3–4 points in the shock transition are generally observed. Away from the shocks, the three methods produce almost identical results. As a side remark, the accuracy and efficiency of FLO52R and the improved version of ARC2D are comparable; see Ref. 20 for more details. Both codes use central difference in space with similar numerical dissipation terms, but they use different time-stepping methods for steady-state applications. These two codes are widely circulated. Figures 6 and 7 show the pressure coefficients, Mach contours, and pressure contours computed by ARC2D (version 150) for cases 1 and 2, using the same mesh as shown in Fig. 1. Figures 8 and 9 show a comparison of the pressure coefficients of the current scheme with ARC2D for cases 1 and 2. One can see that the

current method captures the shocks better than ARC2D, especially on the lower surface.

Here, as a guideline, the results of Pulliam and Barton¹⁵ (using ARC2D, version 150) and the not yet published results of the AGARD Fluid Dynamics Panel Working Group 07 are used as the “exact” solutions. Pulliam and Barton computed all the cases with very fine grids of 561×65 and with very dense clustering near the shocks. Figures 10 and 11 show the grid they used and their computed results for cases 1 and 2. Even though no grid refinements have been conducted on the present scheme for the above cases, shock strengths and shock locations of the current computations on the 249×41 grid coincide very well with the fine grid results of Pulliam and Barton. Furthermore, the present numerical experiments indicate that in order to produce the same accuracy as this TVD scheme, special clustering of grid points

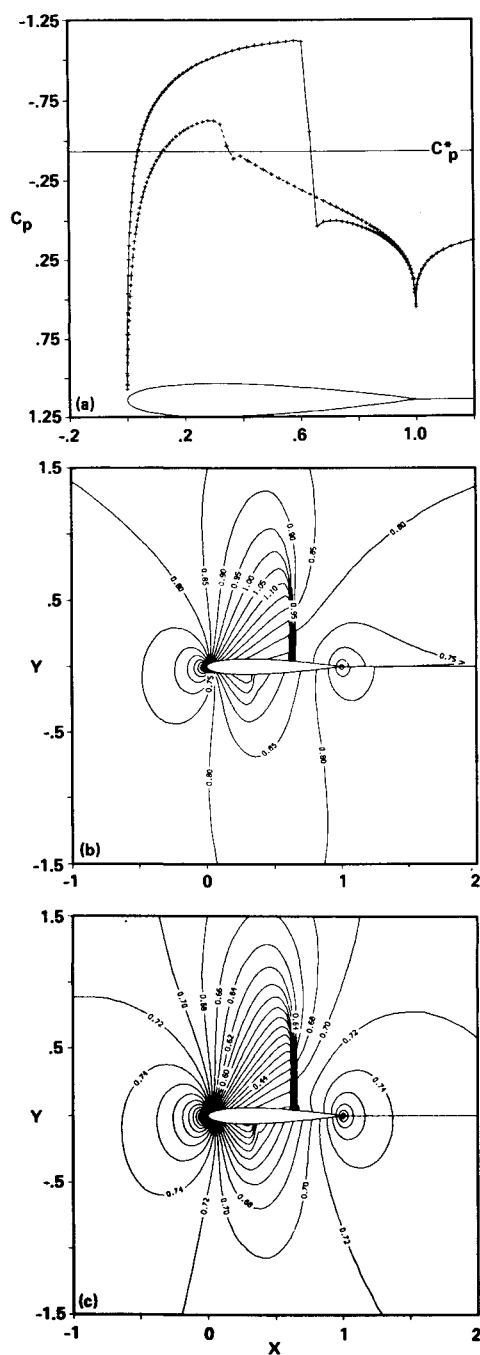


Fig. 2 a) Pressure coefficient distributions, b) Mach contours, and c) pressure contours for the NACA-0012 airfoil with $M_\infty = 0.8$, $\alpha = 1.25$.

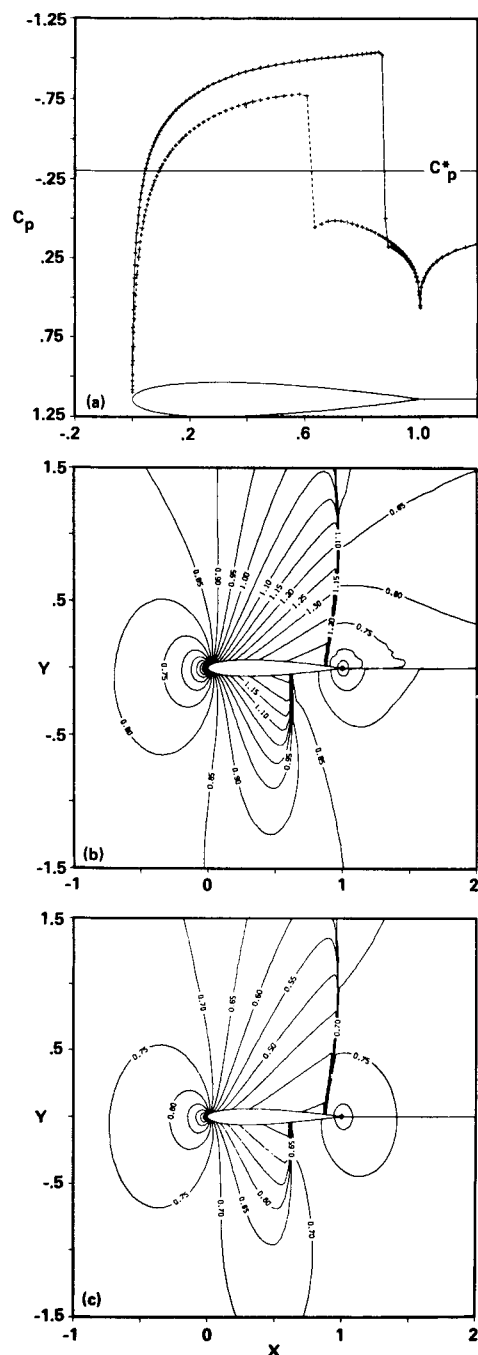


Fig. 3 a) Pressure coefficient distributions, b) Mach contours, and c) pressure contours for the NACA-0012 airfoil with $M_\infty = 0.85$, $\alpha = 1.0$.

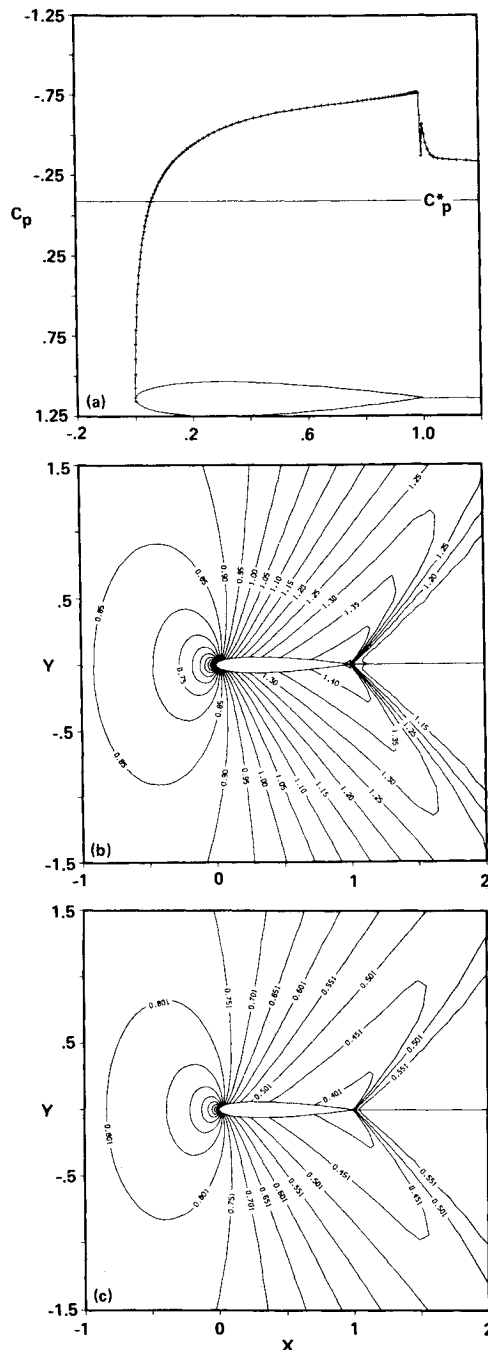


Fig. 4 a) Pressure coefficient distributions, b) Mach contours, and c) pressure contours for the NACA-0012 airfoil with $M_\infty = 0.95$, $\alpha = 0.0$.

near the shocks and/or a denser grid have to be employed by FLO52R and ARC2D. Thus, the advantage of the current scheme over FLO52R and ARC2D is that the ill condition in computations that arises from the high aspect ratio of grid spacing can be avoided. Moreover, the present method requires less storage and operation count than most other upwind TVD schemes.^{3,7,9} See Ref. 13 for an explanation.

For cases 1 and 4, the L_2 -norm residual (of the explicit operator) reaches machine zero at around 3000 steps. A residual of 10^{-7} can be reached in around 800 steps. Cases 2 and 3 are slightly more difficult. The shocks appear to be oblique and not aligned with the C-grid coordinate system, and the convergence rates are slower. Since the computer code is not fully vectorized and is not coded in an optimized fashion, it requires 0.7 s/time step on the Cray-XMP at the

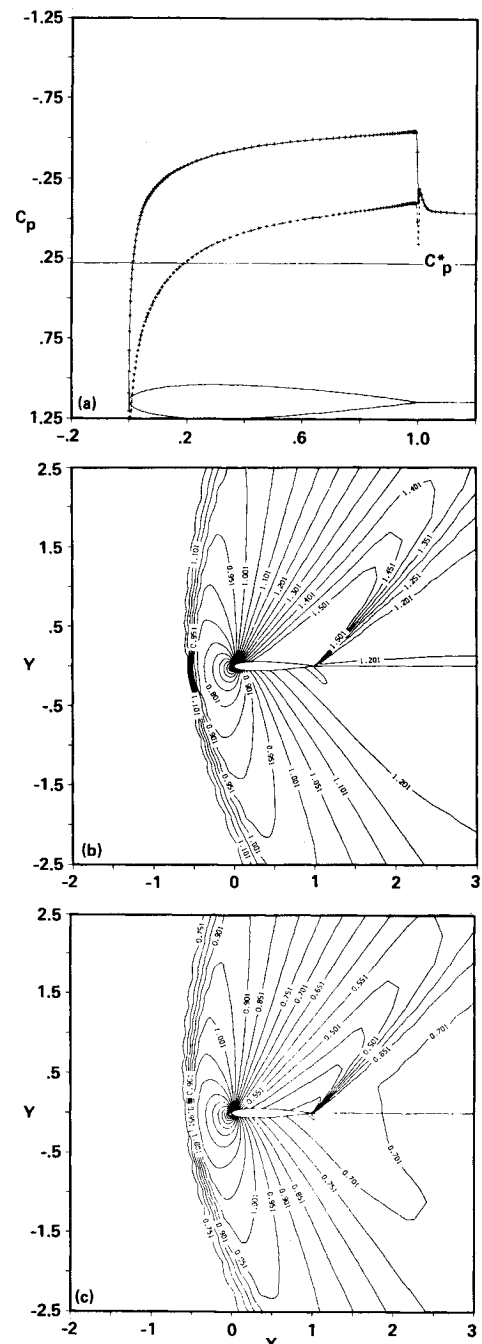


Fig. 5 a) Pressure coefficient distributions, b) Mach contours, and c) pressure contours for the NACA-0012 airfoil with $M_\infty = 1.2$, $\alpha = 7.0$.

NASA Ames Research Center (based on the 249×41 C-grid). A careful recording could possibly increase the speed by a factor of 2 or more. When the vectorized option is turned off, the current code requires 1.69 s/time step, while ARC2D requires 1.01 s/time step. This indicates that the present method requires 68% more computation time than ARC2D.

Evaluation of the Symmetric Averages $U_{j+\frac{1}{2},k}$ and $U_{j,k+\frac{1}{2}}$ for the Euler Equations

For a perfect gas, numerical experiments have been performed with two types of averaging for $U_{j+\frac{1}{2},k}$ and $U_{j,k+\frac{1}{2}}$. The simplest form of $U_{j+\frac{1}{2},k}$ is the arithmetic average $U_{j+\frac{1}{2},k} = 0.5 \cdot (U_{j+1,k} + U_{j,k})$. The other, Roe's averaging,¹⁷ is only applicable to a perfect gas. It has the computational advantage of perfectly resolving stationary discontinuities.

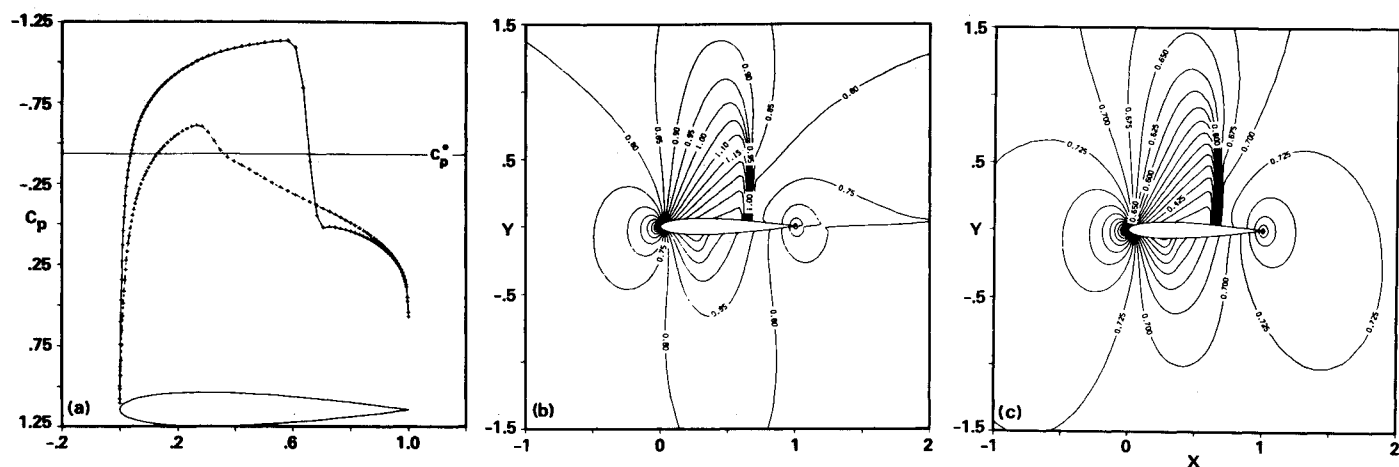


Fig. 6 a) Pressure coefficient distributions, b) Mach contours, and c) pressure contours computed by ARC2D (version 150) for the NACA-0012 airfoil with $M_\infty = 0.8$, $\alpha = 1.25$.

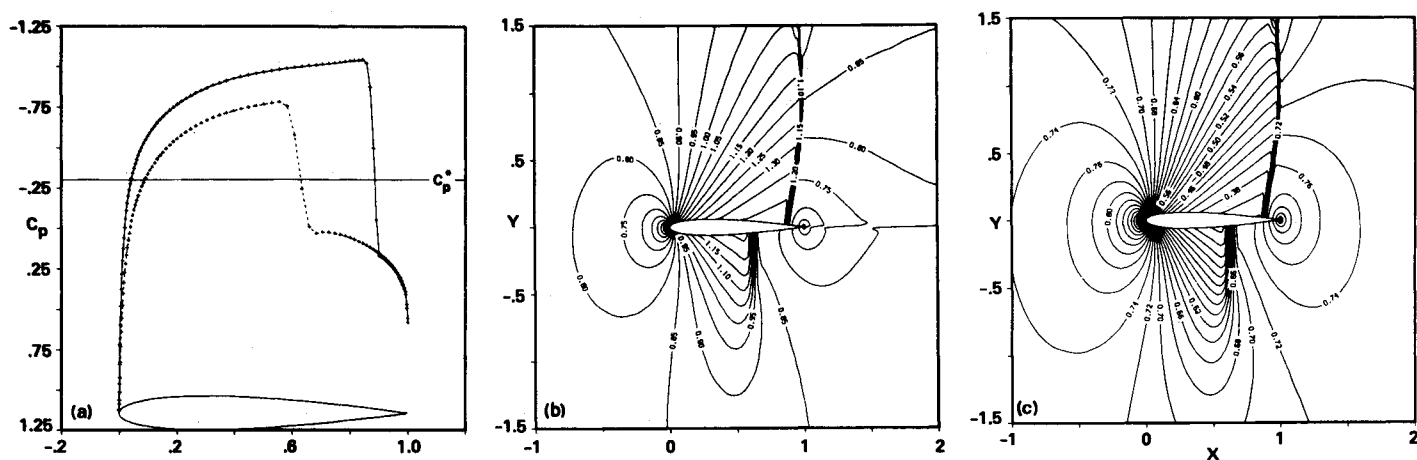


Fig. 7 a) Pressure coefficient distributions, b) Mach contours, and c) pressure contours computed by ARC2D (version 150) for the NACA-0012 airfoil with $M_\infty = 0.85$, $\alpha = 1.0$.

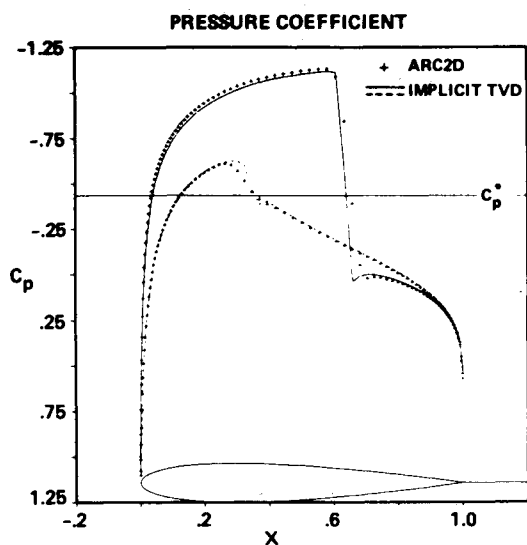


Fig. 8 Comparison of the current scheme with ARC2D (version 150) for the NACA-0012 airfoil with $M_\infty = 0.8$, $\alpha = 1.25$.

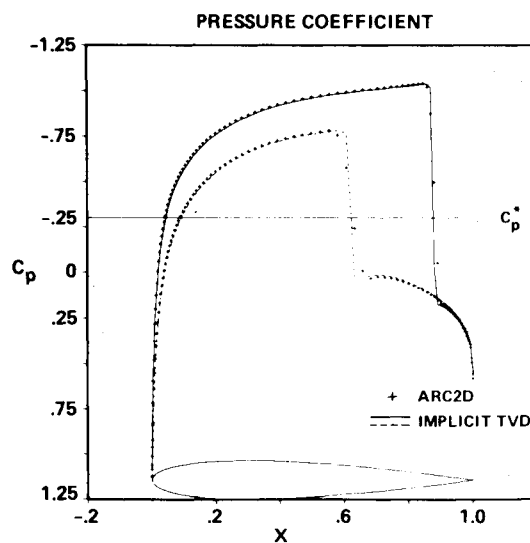


Fig. 9 Comparison of the current scheme with ARC2D (version 150) for the NACA-0012 airfoil with $M_\infty = 0.85$, $\alpha = 1.0$.

Fig. 10 a) The 561×65 grid, b) pressure coefficient distributions, c) Mach contours, and d) pressure contours computed by ARC2D (version 150) for the NACA-0012 airfoil with $M_\infty = 0.8$, $\alpha = 1.25$.

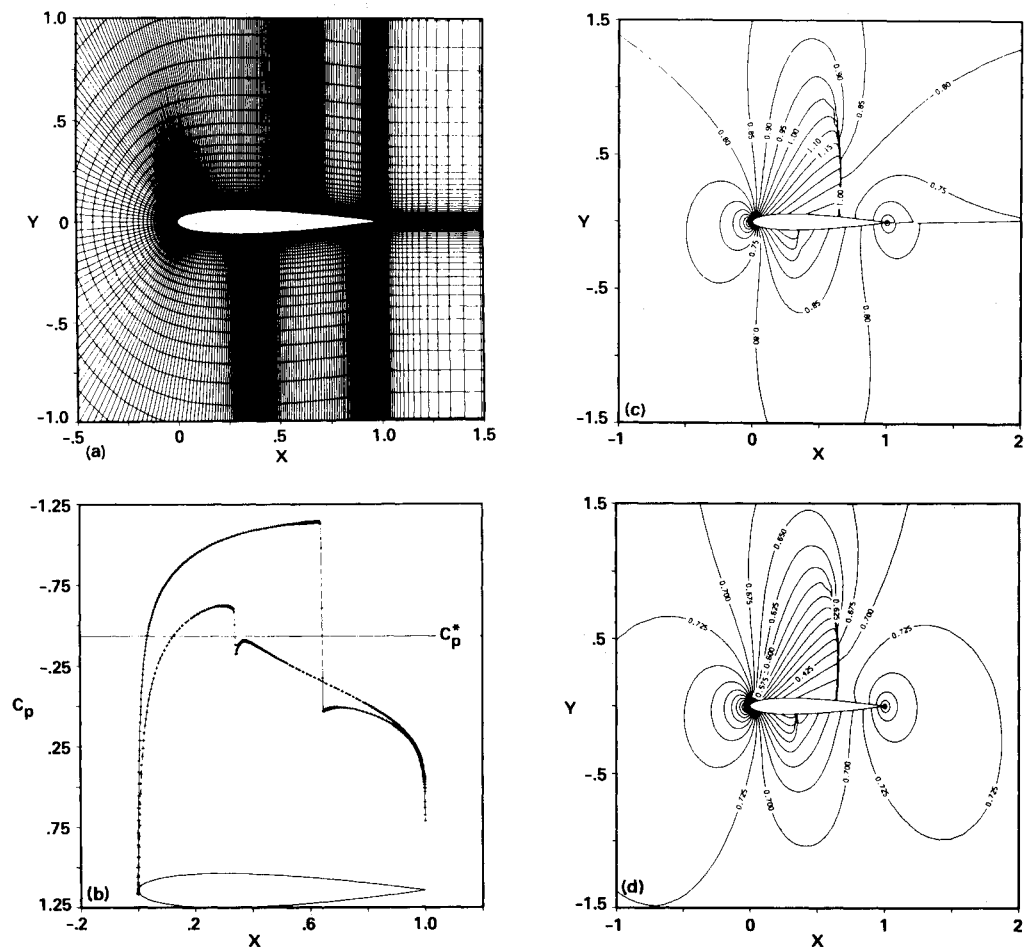
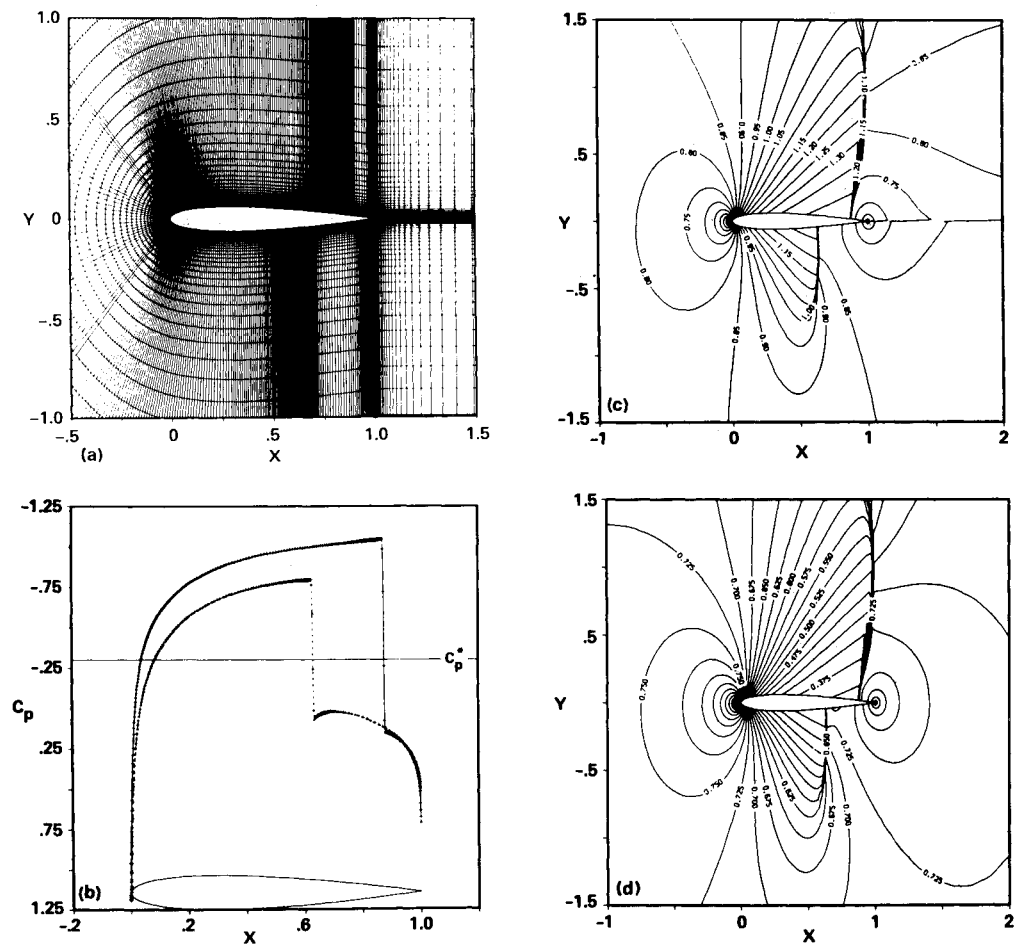


Fig. 11 a) The 561×65 grid, b) pressure coefficient distributions, c) Mach contours, and d) pressure contours computed by ARC2D (version 150) for the NACA-0012 airfoil with $M_\infty = 0.85$, $\alpha = 1.0$.



For detailed implementation, see Ref. 14. However, under certain conditions, such as highly irregular grids or special flow conditions, the characteristic speeds $a_{j+1/2}^l$ can lie outside the interval (a_j, a_{j+1}) . Consequently, under this situation, the direction of upwinding, which is determined solely by the sign of the $a_{j+1/2}^l$ on Roe's schemes, might be the opposite of what is desired. This special property of Roe's averaging was first observed by M. Vinokur of NASA Ames Research Center. Roe's averaging has been tested on a variety of one- and two-dimensional gasdynamics problems,^{16,22,23} and no sign of ill condition or instability was encountered. Numerical experiments with these two averages show no visible difference in numerical solutions for the above airfoil test cases. However, Roe's average requires slightly more computation.

V. Conclusion

Numerical experiments for the airfoil calculations show that the application of the linearized version of the second-order implicit upwind TVD schemes generate good shock resolution for steady-state computations. The numerical dissipation is built in and has an automatic feedback mechanism. This algorithm requires less storage and operation count than most other upwind TVD schemes.^{3,7,9} The method is quite robust and can easily be implemented into an existing central difference code.

Acknowledgments

The authors wish to thank Thomas Pulliam for his generous assistance on the airfoil geometries. The permission to use his well-documented computer code ARC2D as a base code for the development of a research code for the present study is gratefully acknowledged. Thanks also to Garry Carofano, U.S. Armament Research and Development Command, Watervliet, NY, for suggesting the use of the averaged Jacobian value in Eq. (5).

References

- ¹Proceedings of the Eighth International Conference on Numerical Methods in Fluid Dynamics, Lecture Notes in Physics, Vol. 170, edited by E. Krause, Springer-Verlag, New York, 1982.
- ²Proceedings of the AIAA 6th Computational Fluid Dynamics Conference, Danvers, MA, July 1983.
- ³Proceedings of the AMS-SIAM Summer Seminar on Large Scale Computations in Fluid Mechanics, Lectures in Applied Mathematics, Vol. 22, 1985.
- ⁴Harten, A., "A High Resolution Scheme for the Computation of Weak Solutions of Hyperbolic Conservation Laws," NYU Rept., Oct. 1981; *Journal of Computational Physics*, Vol. 49, 1983, pp. 357-393.
- ⁵Harten, A., "On a Class of High Resolution Total-Variation-Stable Finite-Difference Schemes," NYU Report, Oct. 1982; *SIAM Journal of Numerical Analysis*, Vol. 21, 1984, pp. 1-23.
- ⁶van Leer, B., "Towards the Ultimate Conservative Difference Scheme. V. A Second-Order Sequel to Godunov's Method," *Journal of Computational Physics*, Vol. 32, 1979, pp. 101-136.
- ⁷Colella, P. and Woodward, P. R., "The Piecewise-Parabolic Method (PPM) for Gas-Dynamical Simulations," LBL Rept. 14661, July 1982.
- ⁸Roe, P. L., "Some Contributions to the Modeling of Discontinuous Flows," *Lectures in Applied Mathematics*, Vol. 22, Pt. II, 1985, pp. 163-193.
- ⁹Osher, S., "Shock Modeling in Transonic and Supersonic Flow," *Recent Advances in Numerical Methods in Fluids*, Vol. 4, *Advances in Computational Transonics*, edited by W. G. Habashi, Pineridge Press, Swansea, U.K., 1985.
- ¹⁰Yee, H. C., "Generalized Formulation of a Class of Explicit and Implicit TVD Schemes," NASA-TM-86775, July 1985.
- ¹¹Roe, P. L., "Generalized Formulation of TVD Lax-Wendroff Schemes," ICASE Rept. 84-53, Oct. 1984.
- ¹²Davis, S. F., "TVD Finite Difference Schemes and Artificial Viscosity," ICASE Rept. 84-20, June 1984.
- ¹³Yee, H. C., "On the Implementation of a Class of Upwind Schemes for System of Hyperbolic Conservation Laws," NASA TM, in press.
- ¹⁴Yee, H. C., Warming, R. F., and Harten, A., "Implicit Total Variation Diminishing (TVD) Schemes for Steady-State Calculations," *Journal of Computational Physics*, Vol. 57, 1985, pp. 327-360.
- ¹⁵Pulliam, T. H. and Barton, J. T., "Euler Computations of AGARD Working Group 07 Airfoil Test Cases," AIAA Paper 85-0018, 1985.
- ¹⁶Yee, H. C., "Linearized Form of Implicit TVD Schemes for the Multidimensional Euler and Navier-Stokes Equations," *Advances in Hyperbolic Partial Differential Equations, International Journal on Computers and Mathematics with Applications*, Vol. 12A, Nos. 4/5, 1986, pp. 413-432.
- ¹⁷Roe, P. L., "Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes," *Journal of Computational Physics*, Vol. 43, 1981, pp. 357-372.
- ¹⁸Harten, A. and Hyman, J. M., "A Self-Adjusting Grid for the Computation of Weak Solutions of Hyperbolic Conservation Laws," *Journal of Computational Physics*, Vol. 50, 1983, pp. 235-269.
- ¹⁹Beam, R. and Warming, R. F., "An Implicit Finite-Difference Algorithm for Hyperbolic Systems in Conservation Law Form," *Journal of Computational Physics*, Vol. 22, 1976, pp. 87-110.
- ²⁰Pulliam, T. H. and Steger, J., "Recent Improvements in Efficiency, Accuracy and Convergence for Implicit Approximate Factorization Algorithms," AIAA Paper 85-0360, 1985.
- ²¹Jameson, A., Schmidt, W., and Turkel, E., "Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes," AIAA Paper 81-1259, 1981.
- ²²Yee, H. C., Warming, R. F., and Harten, A., "Application of TVD Schemes for the Euler Equations of Gasdynamics," *Lectures in Applied Mathematics*, Vol. 22, Pt. II, 1985, pp. 357-377.
- ²³Yee, H. C. and Kutler, P., "Application of Second-Order-Accurate Total Variation Diminishing (TVD) Schemes to the Euler Equations in General Geometries," NASA TM-85845, Aug. 1983.