Acta Mech
DOI 10.1007/s00707-009-0144-9

M. H. Djavareshkian · M. H. Abdollahi Jahdi

Shock-capturing method using characteristic-based dissipation filters in pressure-based algorithm

Received: 23 February 2008 / Revised: 31 December 2008

Abstract In this paper, an efficient blending procedure based on the pressure-based algorithm is presented to solve the compressible Euler equations on a non-orthogonal mesh with collocated finite volume formulation. The boundedness criteria for this procedure are determined from total variation diminishing (TVD) schemes with and without applying of artificial compression method (ACM) of Harten as a control switch of dissipation. The fluxes of the convected quantities including mass flow rate are approximated by using the characteristic-based TVD and TVD/ACM methods. The algorithm is tested for steady-state inviscid flows at different Mach numbers ranging from the transonic to the supersonic regime and the results are compared with the existing numerical solutions. The comparisons show that the ACM is a useful technique to modify standard high-resolution schemes, which prevents the smearing of discontinuities and improves the resolution of shocks in the pressure-based algorithm. Aside from having the ability of accurately capturing shocked flows, this approach also accelerates the convergence rate of the solution in the supersonic flows with only a maximum increase of 5% in the operations with respect to standard second-order TVD schemes.

1 Introduction

The capturing of sharp gradients in compressible aerodynamic flows as well as shock waves and contact discontinuities has been the subject of much research and development. The work has resulted in the devising of various high-resolution schemes, notably the total variation diminishing (TVD) technique [1] and Jameson's artificial dissipation method [2]. Most of these schemes have been implemented in density-based numerical algorithms. Computational methods used in aerospace applications for the computation of compressible flows have reached a very high level of maturity with respect to accuracy and efficiency.

Nowadays, there are two main approaches advocated in the development of algorithms for the computation of flow at all speeds; first, there is the modification of compressible solvers (density-based) downward to low Mach numbers; second, extending of the classical incompressible solvers (pressure-based) towards the compressible flow.

Density-based algorithms without modification become impractical for low Mach number flows. Turkel et al. [3,4] and Guillard and Viozat [5] introduced the preconditioning techniques and identified that in the low Mach numbers the discretized solution of the compressible fluid flow equations may fail to provide an accurate approximation to the incompressible flow equations. Also, in a recent work, Razavi and Zamzamian [6] presented an artificial compressibility method to solve the incompressible flow equations using a density-based algorithm.

In contrast, pressure-based methods were originally conceived to solve incompressible flows, adopting pressure as a primary variable. The first implementation of pressure-based schemes for compressible flow is

M. H. Djavareshkian · M. H. Abdollahi Jahdi (✉)
Department of Mechanical Engineering, Tabriz University, Tabriz, Iran
E-mail: jahdi@tabrizu.ac.ir
Tel.: +98-91-44034529

Published online: 13 March 2009

widely attributed to the early contribution of Harlow and Amsden [7], based on a semi-implicit finite difference algorithm. In a separate work, Karki and Patankar [8] extended the classical incompressible pressure-based method to the compressible flow regime, thus overcoming the shortcoming of the applicability restriction of the methodology.

Recently, several authors have implemented the TVD technique in pressure-based algorithms. Lien and Leschziner [9] introduced a MUSCL (van Leer [10]) type of TVD scheme into their pressure-based procedure. The slope limiter in their work relies on the gradients of the solved-for dependent variables. Shyy and Thakur [11] developed what they call the controlled variation scheme (CVS), which is based on the formalism of the TVD concept, introduced in the context of incompressible flow. These authors revealed that the existing TVD schemes are not generalized to the pressure-based methods for two reasons: the first is related to the contemporary sequential-iteration methods of solution, which treat the pressure–gradients as source terms in the momentum equations; this is unlike the simultaneous algorithms, which treat gradients as part of the flux vectors. The second reason is the lack of definition of local characteristics on which the flux limiters of the TVD schemes are based. Thakur et al. [12] generalized their CVS scheme to the compressible flows including shocks as well as the incompressible flows.

In the cited works, the gradients of either conserved or primitive variables are used in formulating the flux-limiting function. Indeed, Mulder and Van Leer [13] and Lin and Chieng [14], who carried out extensive numerical experiments, found that, at least for one-dimensional flow, the best accuracy was obtained by using the Riemann variables. This may be because only one of these variables will undergo a small change through a wave or a contact front, whereas large changes take place in conserved or primitive variables. Issa and Javareshkian [15,16] were among the first to implement a high-resolution TVD scheme with characteristic-variables-based flux limiters into a pressure-based finite volume method that solves Euler and Navier–stokes equations in the local coordinate system. The scheme was applied to transient one-dimensional and steady-state two-dimensional compressible flows. Contemporary with the work of Issa and Javareshkian [16] is the work of Kobayashi and Pereira [17], where characteristic-based flux computations were introduced into pressure-correction solution procedures. They used the essentially non-oscillatory (ENO) scheme for the flux calculation and incorporated into a steady state solution method.

The main drawback of the high-resolution schemes such as TVD method is the smearing of discontinuities. The accuracy of these schemes when across a sharp gradient is reduced to first-order to prevent the non-physical oscillations and instabilities. Harten [18,19] introduced a method for calculation of discontinuous solutions of hyperbolic systems of conservation laws by standard first-order schemes, which deal effectively with both shock and contact discontinuities. The method consists of two stages; in the first stage a standard first-order scheme provides a monotonic variation of the solution near sharp gradients, and in the second stage an artificial compression is applied to sharpen transitions at discontinuities. Also, Harten formulated the ACM method based on primitive variables. Yee et al. [20] extended the Harten's first-order ACM scheme to TVD method by different context so that instead of switching from a second-order scheme to a first-order for shock capturing, the order of accuracy maintains acceptable and being capable of accurately capturing of steep gradients. Also, Yee et al. [21] employed the ACM switch directly into the dissipation term of TVD scheme and applied it to the computation of vortex convections, and shock, shear and turbulence interactions. In their work, the flow was viscous and the discontinuities, including shocks, were computed by using direct numerical simulation (DNS) computations. Also, they applied the entropy splitting method in the discretization of equations to overcome the instabilities. Therefore, the reduction of dissipation term does not induce any problem regarding the stability of the solution. Besides, Lie and Noelle [22] applied the ACM as a limiter to the high-order central difference scheme to improve the resolution of linear discontinuities. They used this method for computation of linear advection and unsteady one-dimensional gas dynamic equations and found these limiters disadvantageous when applied to nonlinear fields.

The aforementioned TVD and artificial compression methods have been used in the density-based algorithm and there is no report on the convergency of the solution using these methods. The object of this research is to extend the artificial compression in a high-resolution TVD scheme with characteristic-variables-based flux limiters into a pressure-based algorithm blended with Riemann problem. The equations of compressible and inviscid flows are solved using collocated finite volume and non-orthogonal mesh in the general coordinate system. Unlike the work of Lie and Noelle [22] where the ACM has been considered as a limiter, in this procedure the ACM is a switch and acts in the limiter of TVD schemes to reduce the limiter dissipation. In addition, this switch is applied to both linear and non-linear fields. In this work, we will analyze the influence of the ACM on the flow computation from the viewpoint of accuracy, computational cost and convergency of the solution. The scheme is tested for steady flows at different Mach numbers ranging from transonic to

supersonic flows on a bump and supersonic flow on a ramp in the channel geometry. Also, in this work unlike the other works based on the pressure-based method, the upper limit of the Mach number in the supersonic test cases is extended to 2.5. Below, we will show that the ACM not only improves the resolution of shocks in all cases, but also accelerates the convergency of solutions in supersonic flows.

2 Governing equations and finite volume discretization

The inviscid flow is governed by Euler equations that can be obtained setting viscous terms equal to zero in Navier–Stokes equations. The differential form of Euler equations, which describe the conservation of mass, momentum, and energy can be expressed in Cartesian tensor form as

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u_j)}{\partial x_j} = 0, \tag{1}$$

$$\frac{\partial (\rho u_i)}{\partial t} + \frac{\partial (\rho u_j u_i + p\delta_{ij})}{\partial x_j} = S_i^u, \tag{2}$$

$$\frac{\partial (\rho e)}{\partial t} + \frac{\partial (\rho u_j H)}{\partial x_j} = S^h. \tag{3}$$

Here ρ is the density, u_i and u_j the velocity components, ρu_i the i-component of momentum per unit volume, p is the pressure, H the total enthalpy, $e = \varepsilon + (|V|^2)/2$ the total energy per unit volume, ε is the specific internal energy, and $V = [u, v]^T$, where u and v are the velocity components in the x- and y-directions, respectively. The basic equations (1), (2) and (3) can be expressed in general tensor form as

$$\frac{\partial \varphi}{\partial t} + \frac{\partial F_{ij}}{\partial x_j} = S_\varphi, \tag{4}$$

where φ is the conservative variable and S_φ is the source term based on the property φ.

The preceding conservation equation (4) is discretized through integration over control volumes such as that shown in Fig. 1, and application of the Gauss Divergence Theorem yields a balance involving the rate of change in φ, face fluxes and volume integrated net source as

$$\int_{\mathrm{d}v} \frac{\partial \varphi}{\partial t} \mathrm{d}v + \int_{\mathrm{d}s} F \cdot \vec{n} \mathrm{d}s = \int_{\mathrm{d}v} S_\varphi \mathrm{d}v \tag{5}$$

where $\varphi = \begin{bmatrix} \rho \\ \rho \vec{V} \\ \rho e \end{bmatrix}$, $F = \begin{bmatrix} \rho \vec{V} \\ \rho \vec{V} \otimes \vec{V} + p\bar{I} \\ \rho \vec{V} H \end{bmatrix}$ and $\bar{I}$ representing the unit matrix.

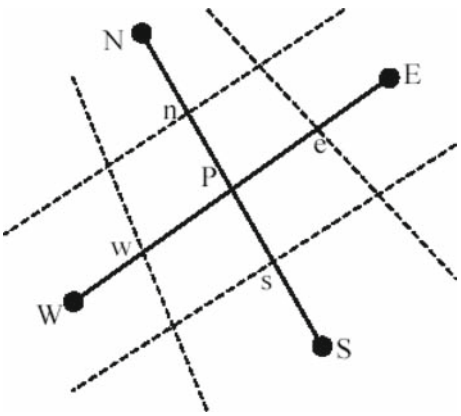

Fig. 1 Finite volume and storage arrangement

For any variable φ, the result of the integration is

$$\frac{\delta v}{\delta t}\left[(\rho\varphi)_p^{n+1} - (\rho\varphi)_p^n\right] + I_e - I_w + I_n - I_s = S_\varphi \delta v, \tag{6}$$

where I is the combined cell-face convection (I_C) and diffusion (I_D) fluxes. The diffusion flux is approximated by central differences and can be written as an example for a cell-face e of the control volume in Fig. 1 as

$$I_{D,e} = D_e(\varphi_P - \varphi_E) - S_e^\varphi, \tag{7}$$

where S_e^φ stands for cross-derivatives arising from mesh non-orthogonality.

3 Convective fluxes

The expressions for the convective fluxes (mass, momentum, and energy) are determined by the TVD scheme used for interpolation from the nodes at the neighboring points, for example nodes E and P. The numerical flux function I_e^C can be written, in general, as

$$I_{C,e} = \frac{1}{2}[I_{C,P} + I_{C,E} + R_e\Phi_e] \tag{8}$$

where $R_e\Phi_e$ is a dissipation term based on the characteristic field decomposition of the flux difference.

For two-dimensional cases with arbitrary grid shapes, the grids cannot generally be aligned with the flow (see Fig. 2).

Therefore, in this work the numerical fluxes in generalized coordinates are described based on the flow parameters and grid-geometry. In Fig. 2, $\hat{\kappa}_x$ and $\hat{\kappa}_y$ are the Cartesian components of the unit vector $\vec{\mathrm{K}}$ along the direction of ζ and normal to the grid cell-face. For the above grid, the trigonometric relations can be written as

$$\begin{cases} \vec{\mathrm{K}} = \dfrac{\vec{\nabla}\zeta}{\left|\vec{\nabla}\zeta\right|} = \hat{\kappa}_x\vec{i} + \hat{\kappa}_y\vec{j}, \\ l = \sqrt{(\delta_x^\zeta)^2 + (\delta_y^\zeta)^2}, \\ \hat{\kappa}_x = \delta_x^\zeta / l, \\ \hat{\kappa}_y = \delta_y^\zeta / l. \end{cases} \tag{9}$$

The flux vectors of mass, momentum and energy conservation can be expressed in any neighbor node of the above cell-face as

$$I_C^\xi = \begin{bmatrix} \rho U^\zeta \\ \rho U^\zeta u + P\delta_x^\zeta \\ \rho U^\zeta v + P\delta_y^\zeta \\ \rho U^\zeta H \end{bmatrix}, \quad U^\zeta = u\delta_x^\zeta + v\delta_y^\zeta. \tag{10}$$

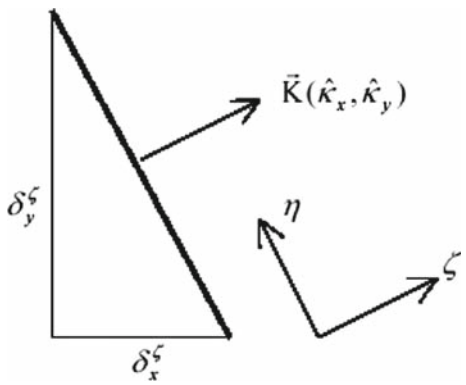

Fig. 2 Grid cell-face in local coordinates

In a one-dimensional system of hyperbolic conservation laws, the Jacobian matrix is defined by $A = \partial F(U)/\partial U$, where U is the vector of conservative variables and $F(U)$ is the vector of fluxes. Let the eigenvalues of A be $(a^1, a^2, \ldots, a^m)$. Denote R as the matrix whose columns are eigenvectors of A, and R^{-1} as the inverse of R. Let a_e^l, R_e, R_e^{-1} denote the quantities of a^l, R, R^{-1} related to A evaluated at U_e based on Roe's averaging method [23]. Define

$$\alpha_e = R_e^{-1}(U_E - U_P) \tag{11}$$

as the difference of the local characteristic variables, between the nodes of E and P. According to Hirsch [24], the eigenvectors and eigenvalues of A in a general two-dimensional coordinate system are defined by

$$R_e = \begin{bmatrix} 1 & 0 & \frac{\tilde{\rho}}{2\tilde{c}} & \frac{\tilde{\rho}}{2\tilde{c}} \\ \tilde{u} & \tilde{\rho}\hat{\kappa}_y & \frac{\tilde{\rho}}{2\tilde{c}}(\tilde{u} + \tilde{c}\hat{\kappa}_x) & \frac{\tilde{\rho}}{2\tilde{c}}(\tilde{u} - \tilde{c}\hat{\kappa}_x) \\ \tilde{v} & -\tilde{\rho}\hat{\kappa}_x & \frac{\tilde{\rho}}{2\tilde{c}}(\tilde{v} + \tilde{c}\hat{\kappa}_y) & \frac{\tilde{\rho}}{2\tilde{c}}(\tilde{v} - \tilde{c}\hat{\kappa}_y) \\ \frac{\tilde{\vec{V}}^2}{2} & \tilde{\rho}(\tilde{u}\hat{\kappa}_y - \tilde{v}\hat{\kappa}_x) & \frac{\tilde{\rho}}{2\tilde{c}}(\tilde{H} + \tilde{c}\tilde{\vec{V}} \cdot \vec{\mathrm{K}}) & \frac{\tilde{\rho}}{2\tilde{c}}(\tilde{H} - \tilde{c}\tilde{\vec{V}} \cdot \vec{\mathrm{K}}) \end{bmatrix}, \tag{12}$$

$$\begin{cases} a^1 = \tilde{U}^\zeta \times l, \\ a^2 = \tilde{U}^\zeta \times l, \\ a^3 = (\tilde{U}^\zeta + \tilde{c}) \times l, \\ a^4 = (\tilde{U}^\zeta - \tilde{c}) \times l, \end{cases} \qquad \tilde{U}^\zeta = \tilde{\vec{V}} \cdot \vec{\mathrm{K}} = \tilde{u}\hat{\kappa}_x + \tilde{v}\hat{\kappa}_y, \tag{13}$$

where l is the length of the grid cell face, c is the sound speed, and $\tilde{\rho}, \tilde{u}, \tilde{v}, \ldots$ are Roe average quantities of the flow variables at the grid cell-face. According to Yee [25], a spatially second-order upwind formula for the components of Φ_e is given by

$$\phi_e^l = \frac{1}{2}\psi(a_e^l)(g_E^l + g_P^l) - \psi(a_e^l + \gamma_e^l)\alpha_e^l, \tag{14}$$

where the superscript l denotes the various characteristics. The flux limiter g may be defined in any way chosen. For the present work, the minmod limiter due to Harten [1] is used; thus,

$$g_P^l = \min \mathrm{mod}(\alpha_e^l, \alpha_w^l), \tag{15}$$

where

$$\min \mathrm{mod}(x, y) = \mathrm{sign}(x) \cdot \max\left\{0, \min\left[|x|, y\,\mathrm{sign}(x)\right]\right\}.$$

For γ one can take [25]

$$\gamma_e^l = \frac{1}{2}\psi(\alpha_e^l) \begin{cases} \frac{g_E^l - g_P^l}{\alpha_e^l} & \alpha_e^l \neq 0, \\ 0 & \alpha_e^l = 0. \end{cases} \tag{16}$$

The entropy function ψ is required to prevent nonphysical solutions, such as expansion shocks, and introduces a small amount of dissipation. Following the suggestion of Harten and Hyman [26], it is taken as

$$\psi(z) = \begin{cases} |z| & |z| \geq \varepsilon_1, \\ \left[z^2 + \varepsilon_1^2\right]/2\varepsilon_1 & |z| < \varepsilon_1, \end{cases} \tag{17}$$

where ε_1 is usually set to zero for problems containing unsteady shocks, and for steady shocks is a small number.

The accuracy of TVD schemes when across a discontinuity is reduced to first order. We obtain a second-order approximation by applying the TVD scheme to the modified flux limiter. The application of the standard TVD scheme to new flux has the effect of reducing the error due to the numerical viscosity [20]. Therefore, we use nonlinear dissipation term in conjunction with Harten's switch applied to each characteristic wave as a filter of the numerical fluxes. The switch acts in discontinuities and enhances the accuracy in sharp gradients

[18]. This is done by increasing the size of the limiter in Eq. (15) by adding a term that is a switch designed by Harten [19,20]. Thus,

$$\bar{g}_P^l = (1 + \omega^l \theta_P^l) g_P^l, \quad \omega > 0 \tag{18}$$

with

$$\theta_P^l = \frac{\left|\alpha_e^l - \alpha_w^l\right|}{\left|\alpha_e^l\right| + \left|\alpha_w^l\right| + \varepsilon}. \tag{19}$$

This serves to limit the action of the TVD dissipation to the immediate vicinity (within one grid cell) of the discontinuity, as detected by steep gradients in the characteristic variables. ω^l is a problem-dependent variable and can be different from one characteristic field to another. In all the computations, $\varepsilon = 10^{-7}$ is chosen to avoid a singularity in the actual coding.

With the preceding assumption, the discretized equations take the form

$$A_P \varphi_P = \sum_{m=E,W,N,S} A_m \varphi_m + S'_\varphi, \tag{20}$$

where the coefficients A are given by

$$\begin{aligned} A_E &= D_e - (F_E/2), \qquad A_N = D_n - (F_N/2), \\ A_W &= D_w - (F_W/2), \qquad A_S = D_s - (F_S/2), \end{aligned} \tag{21}$$

the with $A_P = A_E + A_W + A_N + A_S + (\rho \delta v/\delta t)$, and F is the mass flux.

The term S'_φ in Eq. (20) contains quantities arising from non-orthogonality, numerical dissipation terms, external sources, and $(\rho \delta v/\delta t)\varphi_P$ of the old time-step/iteration level.

4 Solution algorithm

The set of Eq. (20) is solved for the primitive variables (velocity components and energy) together with continuity utilizing pressure-based implicit sequential solution methods. The techniques used are either the PISO scheme for time-dependent problems [15] or the SIMPLE scheme for the steady-state problems presented herein. In both of these techniques, the methodology has to be adapted to handle the way in which the fluxes are computed in Eqs. (8)–(19). The adapted SIMPLE scheme consists of a predictor and corrector sequence of steps at every iteration. The predictor step solves the implicit momentum equation using the old pressure field. Thus, for example, for the u component of velocity, the momentum predictor stage can be written as

$$u^* = H(u^*) - D \nabla p^0 + S'_u, \tag{22}$$

where H contains all terms relating to the surrounding nodes and superscripts $*$ and 0 denote intermediate and previous iteration values, respectively. Note that the pressure–gradient term is now written out explicitly; it is extruded from the total momentum flux by simple subtraction and addition. The corrector-step equation can be written as

$$u^{**} = H(u^*) - D \nabla p^* + S'_u. \tag{23}$$

Hence, from Eqs. (22) and (23),

$$u^{**} - u^* = -D \nabla (p^{**} - p^*), \quad \text{or} \quad \delta u = -D \nabla \delta p. \tag{24}$$

Now the continuity equation demands that

$$\nabla(\rho^* u^{**}) = 0 \tag{25}$$

for steady-state flows. For compressible flows it is essential to account for the effect of change of density on the mass flux as the pressure changes. This is accounted for by linearizing the mass fluxes as follows (as in [8]):

$$\rho^* u^{**} \approx \rho^0 u^* + \rho^0 \delta u + u^* \delta \rho \tag{26}$$

or

$$\rho^* u^{**} \approx \rho^0 u^* - \rho^0 D \nabla \delta p + u^* \left(\frac{\mathrm{d}\rho}{\mathrm{d}p}\right) \delta p, \tag{27}$$

where Eq. (24) is invoked to eliminate δu, and $\delta\rho$ is related to δp by the appropriate equation of state. Substitution of Eq. (27) into Eq. (25) yields a pressure-correction equation of the form

$$A_P \cdot \delta p_P^* = A_E \cdot \delta p_E^* + A_W \cdot \delta p_W^* + A_N \cdot \delta p_N^* + A_S \cdot \delta p_S^* + S_P, \tag{28}$$

where S_P is the finite difference analog of $\nabla(\rho^0 u^*)$, which vanishes when the solution is convergent.

The coefficients A in Eq. (28) take the form (the expression for A_E is given as an example)

$$A_E = (\rho^0 \widehat{a} D)_e - \lambda_e (\widehat{a} u^*)_e \cdot \left(\frac{\mathrm{d}\rho}{\mathrm{d}p}\right)_e, \tag{29}$$

where λ is a factor whose significance is explained subsequently, and $\widehat{a}$ is cell face area.

Because the mass flux at a cell face is computed directly (via Eq. (8)) from nodal values of density and velocity, the cell face values ρ_e^0 and u_e^* in Eq. (29) are not readily available. To compute those values, assumptions concerning the variations of ρ need to be made. For example, if upwinding is chosen, then λ would take the value of 1 when u is positive; otherwise it would be zero. Alternatively, if a central difference formula is used, then $\lambda = \frac{1}{2}$. It is important to recognize, however, that such assumptions have no influence whatsoever on the final solution because they affect only the pressure-correction coefficients, and as δp goes to zero at convergence, the solution is, therefore, independent of how those coefficients are formulated; however, they do influence the convergence behavior. What is important is how $\nabla(\rho^0 u^*)$ in Eq. (28) is computed (as this does determine the solution), and this is based on the TVD principle outlined earlier.

The structure of the coefficients in Eq. (28) simulates the hyperbolic nature of the equation system. Indeed, a closer inspection of expression (29) would reveal an upstream bias of the coefficients (A decreases as u increases), and this bias is proportional to the square of the Mach number. Also note that the coefficients reduce identically to their incompressible form in the limit of Mach number zero.

The overall solution procedure follows the same steps as in the standard SIMPLE algorithm, with the exception of solving the hyperbolic-like pressure-correction equation (28). To ensure convergence of the iterative process, under-relaxation factors between 0.1 and 0.2 for pressure correction and between 0.1 and 0.4 for the other variables are employed.

5 Boundary conditions

At the inlet of the domain, all flow variables are specified if the flow is supersonic. For subsonic inlet flow, only three of the four variables need to be prescribed: the total temperature, the angle of attack, and the total pressure. The pressure is obtained by zeroth-order extrapolation from interior points. At the outlet, all of the flow variables are obtained by linear extrapolation for supersonic velocities; the pressure is fixed when the outlet is subsonic. Slip boundary conditions are used on the lower and upper walls of the bump in the inviscid flow test cases.

6 Results

The first validation is the flow over a bump in the channel geometry in which transonic and supersonic flows were computed by TVD and ACM methods and compared together and with the other existing numerical predictions, including pressure-based or density-based methods. It is then validated for a supersonic internal flow through a ramp in channel geometry for which the results of TVD and ACM methods are compared together.

Flow over an arc bump is a typical problem used to validate various algorithms for all Mach number flows. In the transonic case, the thickness of the circular arc "bump" on the lower wall is 10% of the bump length, while in the supersonic cases the thickness of the arc bump is only 4% of its length.

In Fig. 3a the geometry of a 10% thick bump on a channel wall is shown together with the algebraic mesh (98×26) used to compute steady two-dimensional flow. The transonic case was considered for this geometry

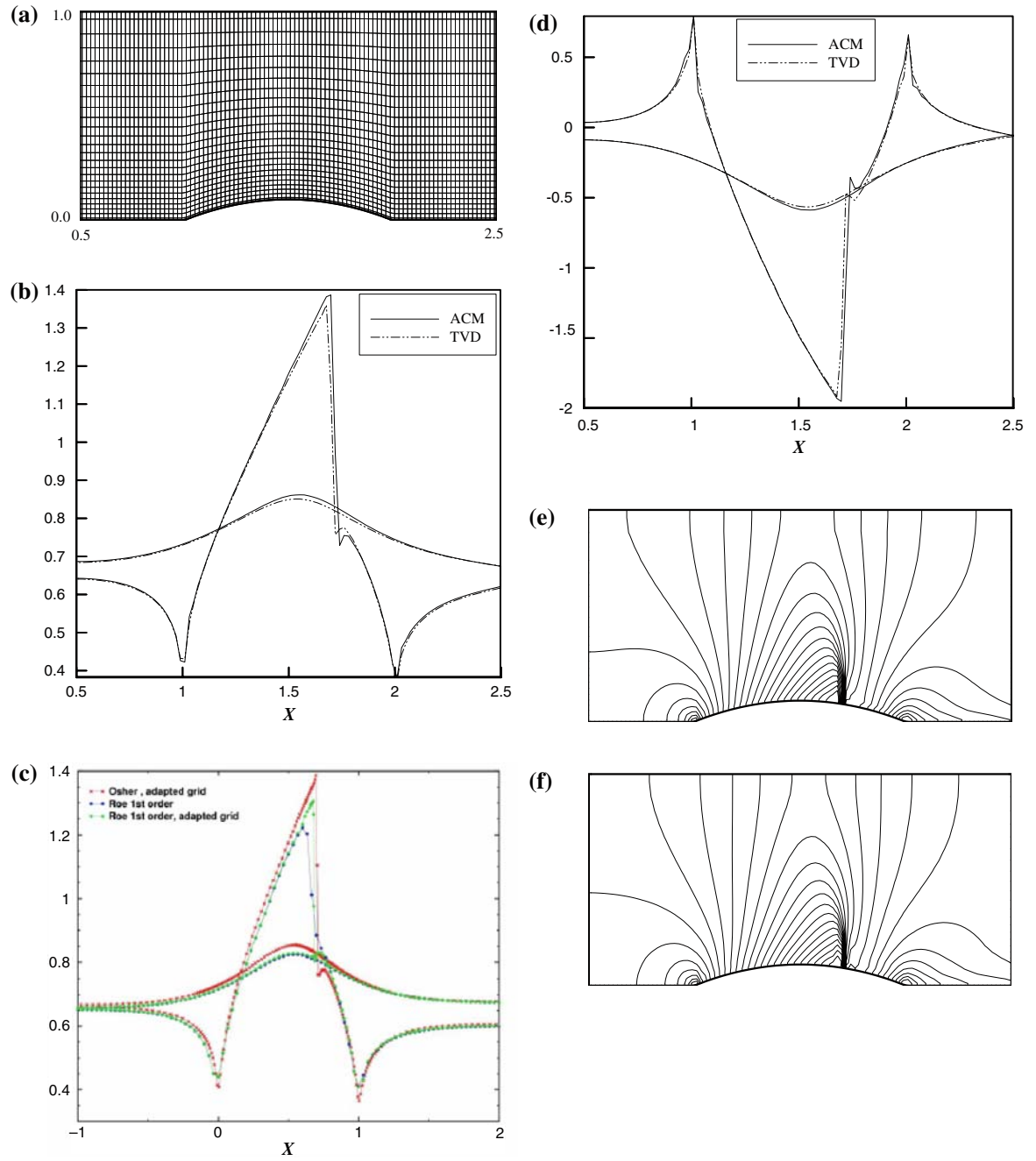

Fig. 3 Transonic flow over 10% thick bump, inlet M =0.675. **a** Bump geometry and 96 × 28 mesh. **b** Mach number on walls (present work). **c** Mach number on walls [27]. **d** Pressure coefficient on walls (present work). **e** Mach contours (present TVD) **f** Mach contours (present ACM)

with an inlet Mach number of 0.675 leading to flow over the bump. Figure 3 presents the results of transonic flow in a similar geometry (Fig. 3a). The Mach number and pressure distributions on the lower and upper walls for TVD and ACM schemes with $\omega = 3.0$ for both the linear and nonlinear fields are shown in Fig. 3b and d, respectively. Kozel [27] used a density-based method with Osher scheme in adaptive grid (Fig. 3c). Compared with this work, it is found that the agreement between the two solutions is close, thus verifying the validity of the present ACM method. Results show that the new ACM scheme has a sharper shock compared to the TVD procedure. Also, Fig. 3e and f show that the resolution of shock waves has been clearly improved by the ACM method.

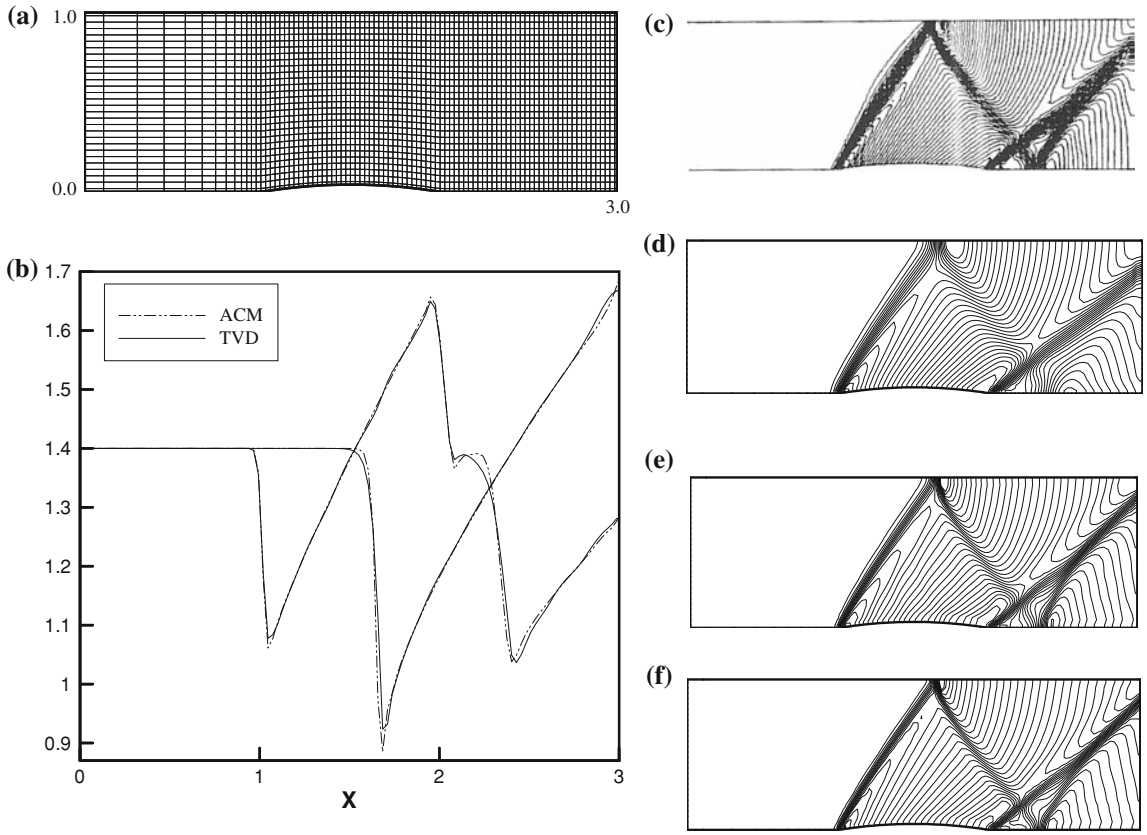

Fig. 4 Supersonic flow over 4% thick bump, inlet M=1.4. **a** Supersonic bump geometry and 90 × 30 mesh. **b** Mach number on walls (present work). **c** Mach contours [28]. **d** Mach contours (TVD) [16]. **e** Mach contours (present TVD). **f** Mach contours (present ACM)

Figures 4 and 5 present the solutions for two supersonic flows ($M = 1.4$ and 1.65) past the 4% thick circular arc bumps in a channel, respectively. Figure 4a shows the geometry and the grid comprising 90×30 control volumes used in the computations. Although the mathematical model is simpler than the Navier–Stokes equations, the presence of discontinuities and their interactions (shock–shock interaction and shock reflection) serves as a good test for the capabilities of the method to resolve shock waves and their interactions. The results of the TVD and ACM schemes in terms of Mach number on upper and lower walls are shown in Figs. 4b and 5a for the cases $M = 1.4$ and $M = 1.65$, respectively. The results show that the capture of shocks in the ACM method is sharper than the same solution in the TVD method, especially in shock reflection zones. The constant Mach number contours are chosen to compare the resolution of shock waves for these schemes. The results of Jameson's scheme in the density-based algorithm on a grid of 145×33 nodes [28], and a TVD scheme in a pressure-based algorithm on a grid of 90×30 [16] for the case $M = 1.4$, are shown in Fig. 4c and d, respectively. The result of the ENO scheme in a pressure-based algorithm on a grid of 80×30 nodes [17] is shown in Fig. 5b for the case $M = 1.65$. Figure 4e and f show the results of Mach contours for $M = 1.4$ case, and Fig. 5c and d shows the results for $M = 1.65$ case obtained with the present methodology. As it can be seen from these figures, all shock structures are very well resolved by the ACM method. Also, by comparison of the present TVD result (Fig. 4e) with the same scheme result in ref. [16] (Fig. 4d) it is found that the present flow calculations (general coordinate considerations) have a better prediction in shock reflection and shock–shock interaction compared to the other method (local coordinate considerations).

Figure 6 shows the grid and presents the solution results for the two supersonic flows ($M = 2.0$ and 2.5) past a 10° ramp in the channel geometry on a grid of 60×30 nodes, respectively. The comparisons indicate that the resolution of shock waves has been clearly improved by applying the ACM scheme.

Figures 7, 8, 9, 10 and 11 show the residual history and convergency of transonic and supersonic flow calculations in present work. The number of solution iterations required to drive the residuals (velocities, pressure and density) down by five orders of magnitude or given accuracy is shown. For simplicity, the residual history

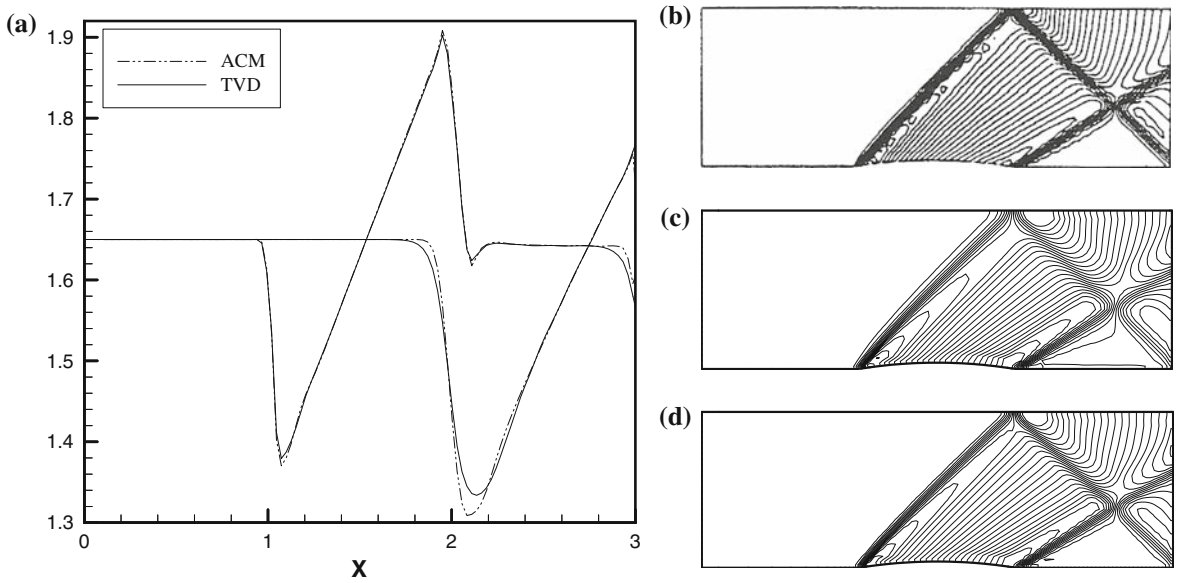

Fig. 5 Supersonic flow over 4% thick bump, inlet M =1.65. **a** Mach number on walls (present work). **b** Mach contours [17]. **c** Mach contours (present TVD). **d** Mach contours (present ACM)

is shown only for the streamwise velocity (u). Tables 1, 2, 3, 4 and 5 list the computing times on a 2.4 GHz Pentium IV processor to reach a reduction of the residual of five orders of magnitude or to carry out 3,000 iterations.

The ACM coefficients ($\omega_{1,2}$ for linear and $\omega_{3,4}$ for nonlinear fields) are functions of the physics of the problem. Their optimum values are required to have sharper shock waves and faster convergence rate which differ from one case to another. In general, by applying the ACM we will have an increase in accuracy and for each case the sharper shock is achieved at relatively high values of ACM coefficients. The convergence rate is dependent on the choice of values of ω_1, ω_2, ω_3 and ω_4. The best convergence rate, however, depends on the optimum values of these coefficients, which can be obtained by using the trial and error procedure. The resolution of discontinuities is improved by increasing the coefficients. However, increasing the coefficients by much affects the convergence process and instability may occur.

In the transonic case, only for small ACM coefficients for nonlinear fields (ω_3 and ω_4) the convergence rate and CPU time are slightly better than for the TVD method (Table 1). For the other coefficient values, the CPU time is larger than that required in the TVD method. To clearly indicate the difference between the convergence histories of ACM and TVD methods, the large coefficients have been chosen in the transonic regime (see Fig. 7). For the other coefficients shown in Table 1, the convergence histories nearly coincide. In addition, in this case the solution is less sensitive to the choice of ACM coefficients from the convergence point of view.

In supersonic flows with strong shocks the solution is very sensitive to the numerical dissipation and we cannot choose larger values of coefficients. Especially at high Mach numbers, if the coefficients of nonlinear fields (ω_3 and ω_4) are chosen larger than coefficients of linear fields (ω_1 and ω_2), the convergence may be postponed. For $M = 1.4$, due to the aforementioned restrictions the coefficients have been chosen less than 2.0. According to Table 2, we have minimum CPU time when the coefficients lie in the range of 0.6–0.7 for all fields. Also, Fig. 8 shows that for a certain criteria the best convergence is achieved at $\omega = 0.7$ for all fields. In higher Mach numbers, if the coefficients are chosen greater than unity, the convergence is destroyed.

The coefficients shown in Figs. 7, 8, 9, 10 and 11 and Tables 1, 2, 3, 4 and 5 are chosen arbitrarily and the optimal values of the coefficients are those with which the solution has the most rapid convergence compared to the other coefficients. According to Fig. 9 and Table 3 for $M = 1.65$, the optimal values of the coefficients are set equal to 0.6 for all fields or 0.7 for linear and 0.6 for nonlinear fields. For the case of $M = 2.0$ the convergence improvement is very considerable. Figure 10 shows that the value of $\omega = 0.6$ for all fields is a good choice from the convergence point of view. The value of 0.6 is only a typical value and according to Table 4 there are other values that impose less computational costs. Also, Fig. 11 and Table 5 show that for the case $M = 2.5$, the influence of different ACM coefficients on the convergence and computational cost are almost the same.

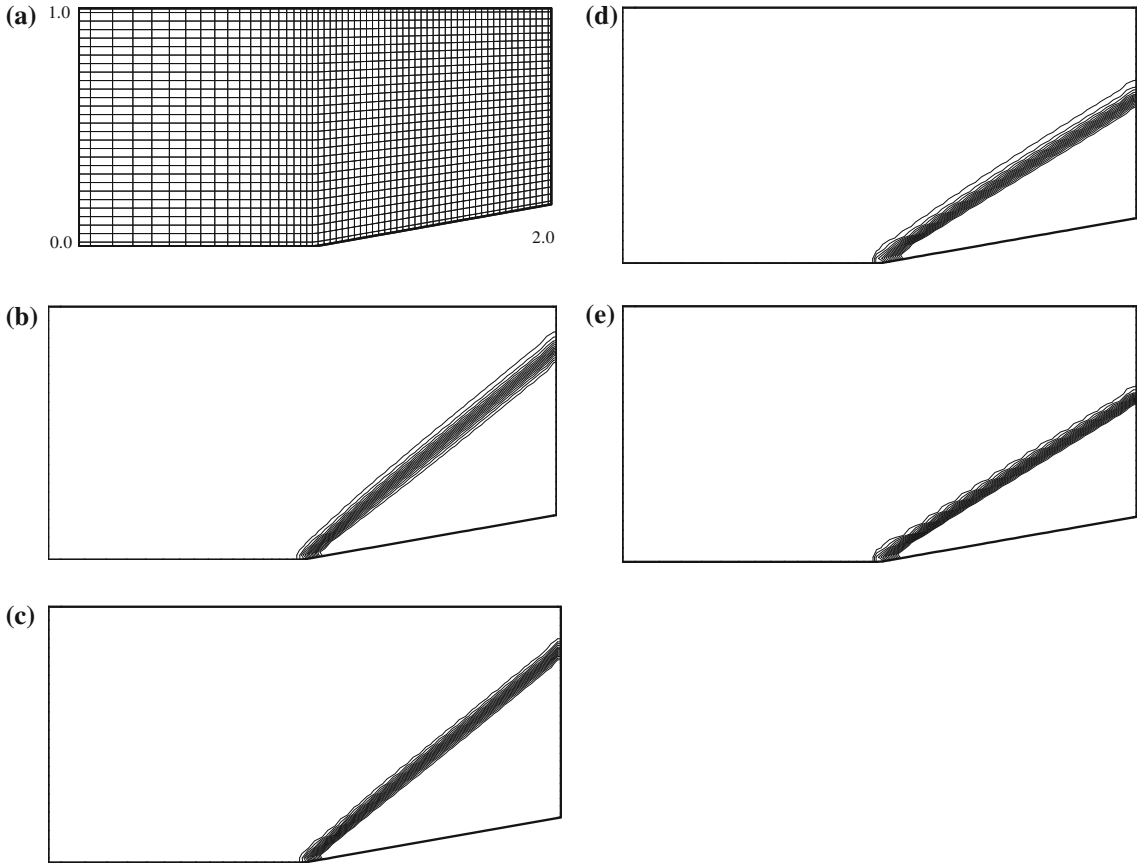

Fig. 6 Supersonic flow over 10° ramp. **a** Supersonic ramp geometry and 60×20 mesh. **b** Mach contours (present TVD), $M = 2.0$. **c** Mach contours (present ACM), $M = 2.0$. **d** Mach contours (present TVD), $M = 2.5$. **e** Mach contours (present ACM), $M = 2.5$

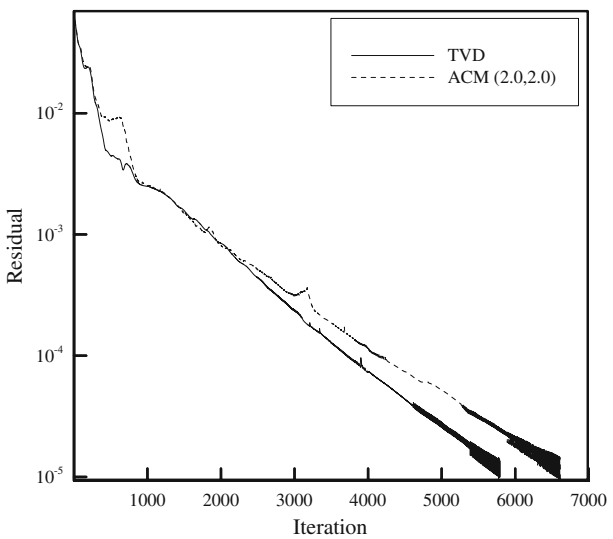

Fig. 7 Residual history of velocity 'u' for transonic case. The numbers in parenthesis (2.0, 2.0) correspond to ($\omega_{1,2}$, $\omega_{3,4}$) for ACM, where ω_1 and ω_2 are considered for linear and ω_3 and ω_4 for nonlinear fields

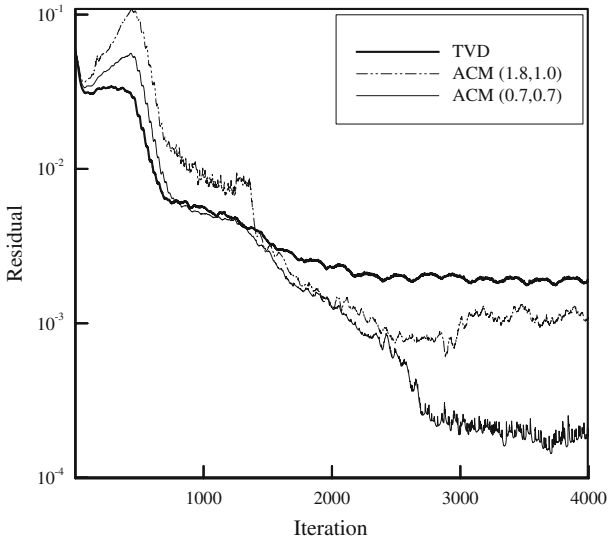

Fig. 8 Residual history of velocity 'u' for $M = 1.4$. The numbers in parenthesis (1.8, 1.0) and (0.7, 0.7) correspond to ($\omega_{1,2}$, $\omega_{3,4}$) for ACM, where ω_1 and ω_2 are considered for linear and ω_3 and ω_4 for nonlinear fields

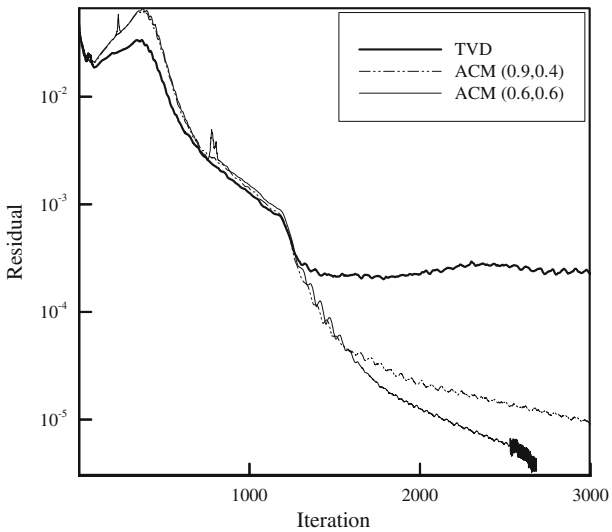

Fig. 9 Residual history of velocity 'u' for $M = 1.65$. The numbers in parenthesis (0.9, 0.4) and (0.6, 0.6) correspond to ($\omega_{1,2}$, $\omega_{3,4}$) for ACM, where ω_1 and ω_2 are considered for linear and ω_3 and ω_4 for nonlinear fields

Overall, the results show that in supersonic flows although we have an increase in the operations for the ACM, with the fixed number of steps its rate of convergence is faster than for TVD and less CPU time is needed to reach a certain accuracy. Therefore, in this procedure the computational costs are small compared to the TVD method.

Shock capturing improvement may be attributed to the fact that by using the modified limiter in ACM makes the numerical characteristic speeds more convergent and hence improves the resolution of the computed shocks. Also, the convergency improvement by increasing the Mach number might be because of the characteristic-based and hyperbolic behavior of the ACM method, which is consistent with supersonic flow. This improvement is not considerable in the transonic regime where only a small part of flow is supersonic.

It is worth mentioning that in each case the TVD method and the ACM with different coefficients are compared using the same under-relaxation factor for any flow parameter.

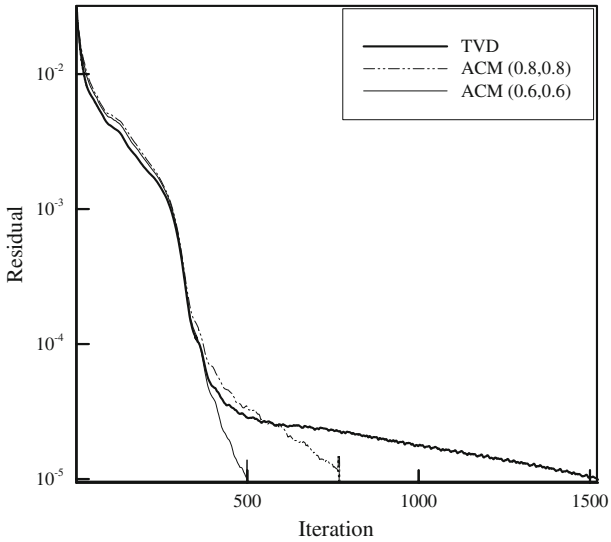

Fig. 10 Residual history of velocity 'u' for $M = 2.0$. The numbers in parentheses is (0.8, 0.8) and (0.6, 0.6) correspond to $(\omega_{1,2}, \omega_{3,4})$ for ACM, where ω_1 and ω_2 are considered for linear and ω_3 and ω_4 for nonlinear fields

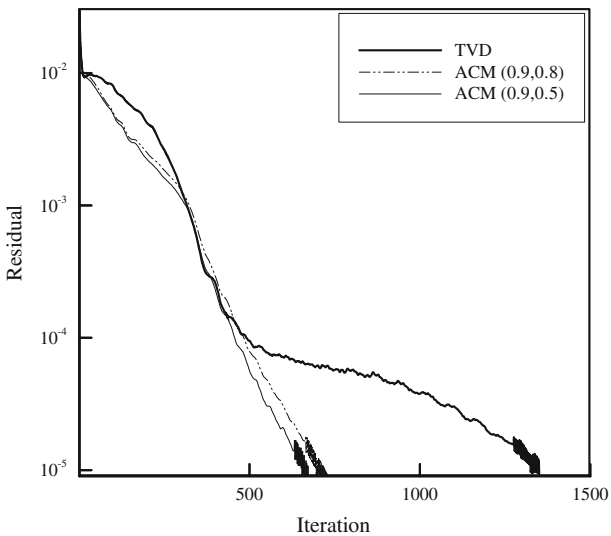

Fig. 11 Residual history of velocity 'u' for $M = 2.5$. The numbers in parentheses is (0.9, 0.8) and (0.9, 0.5) correspond to $(\omega_{1,2}, \omega_{3,4})$ for ACM, where ω_1 and ω_2 are considered for linear and ω_3 and ω_4 for nonlinear fields

Table 1 CPU time (s) for transonic case to reduce residuals by five orders of magnitude

Case	Method					
	TVD	ACM	ACM	ACM	ACM	ACM
		$\omega_{1,2} = 2.0$ $\omega_{3,4} = 1.0$	$\omega_{1,2} = 2.0$ $\omega_{3,4} = 0.5$	$\omega_{1,2} = 1.5$ $\omega_{3,4} = 0.06$	$\omega_{1,2} = 0.8$ $\omega_{3,4} = 0.8$	$\omega_{1,2} = 0.2$ $\omega_{3,4} = 0.2$
Transonic, $M = 0.675$	725	773	752	724	754	736

Table 2 CPU time (s) for supersonic case until 3,000 iterations

Case	Method					
	TVD	ACM	ACM	ACM	ACM	ACM
		$\omega_{1,2} = 1.5$ $\omega_{3,4} = 1.0$	$\omega_{1,2} = 1.5$ $\omega_{3,4} = 0.8$	$\omega_{1,2} = 0.7$ $\omega_{3,4} = 0.7$	$\omega_{1,2} = 0.6$ $\omega_{3,4} = 0.6$	$\omega_{1,2} = 0.6$ $\omega_{3,4} = 0.5$
Supersonic, $M = 1.4$	409	425	427	421	421	426

Table 3 CPU time (s) for supersonic case until 3,000 iterations

Case	Method					
	TVD	ACM	ACM	ACM	ACM	ACM
		$\omega_{1,2} = 1.0$ $\omega_{3,4} = 1.0$	$\omega_{1,2} = 0.9$ $\omega_{3,4} = 0.8$	$\omega_{1,2} = 0.8$ $\omega_{3,4} = 0.7$	$\omega_{1,2} = 0.7$ $\omega_{3,4} = 0.6$	$\omega_{1,2} = 0.7$ $\omega_{3,4} = 0.2$
Supersonic, $M = 1.65$	374	396	395	394	393	398

Table 4 CPU time (s) for supersonic case to reduce residuals by five orders of magnitude

Case	Method					
	TVD	ACM	ACM	ACM	ACM	ACM
		$\omega_{1,2} = 0.9$ $\omega_{3,4} = 0.9$	$\omega_{1,2} = 0.8$ $\omega_{3,4} = 0.8$	$\omega_{1,2} = 0.6$ $\omega_{3,4} = 0.6$	$\omega_{1,2} = 0.4$ $\omega_{3,4} = 0.4$	$\omega_{1,2} = 1.0$ $\omega_{3,4} = 0.0$
Supersonic, $M = 2.0$	129	120	69	48	46	40

Table 5 CPU time (s) for supersonic case to reduce residuals by five orders of magnitude

Case	Method					
	TVD	ACM	ACM	ACM	ACM	ACM
		$\omega_{1,2} = 0.95$ $\omega_{3,4} = 0.9$	$\omega_{1,2} = 0.9$ $\omega_{3,4} = 0.8$	$\omega_{1,2} = 0.8$ $\omega_{3,4} = 0.8$	$\omega_{1,2} = 0.7$ $\omega_{3,4} = 0.7$	$\omega_{1,2} = 0.9$ $\omega_{3,4} = 0.5$
Supersonic, $M = 2.5$	115	64	67	66	65	61

7 Conclusion

The application of the artificial compression method of Harten to a pressure-based implicit procedure blended with the Riemann solver has been presented. The method incorporates bounded high resolution of discontinuities and is therefore well suited to all flow regimes ranging from transonic to supersonic. The boundedness criteria for this procedure are determined using TVD and TVD/ACM schemes, which are applied to the fluxes of the convected quantities, including mass, at the cell faces. A collocated finite volume method is then used to solve the governing equations in a general coordinate system. The flux limiter and ACM switch is based on the characteristic variables. The main findings can be summarized as follows:

(1) The agreement between the results of the presented implementation of the ACM scheme and those of the other schemes such as density-based methods is excellent, as perhaps expected.
(2) The presented TVD and ACM schemes based on the new approach (i.e. using general coordinate system considerations) generate a better resolution of discontinuities compared to the other TVD schemes employed in the pressure-based algorithms (see Fig. 4 for a close-up comparison).
(3) With the same algorithm, the presented ACM method has an excellent resolution of flows under serve conditions (simple shocks, shock reflection, shock/shock interaction, etc.) in comparison to the TVD methods.
(4) The ACM method accelerates the convergency of the solution in the supersonic flows with respect to the standard second-order TVD schemes.

References

1. Harten, A.: High resolution scheme for hyperbolic conservation laws. J. Comput. Phys. **49**(3), 357–393 (1983)
2. Jameson, A., Schmidt, W., Turkel, E.: Numerical solutions of the Euler equations by finite-volume methods using Runge–Kutta time-stepping schemes. AIAA, Paper 81-1259, June (1981)
3. Turkel, E.: Preconditioned methods for solving the incompressible and low speed compressible equations. J. Comput. Phys. **72**, 277–298 (1987)
4. Turkel, E., Radespiel, R., Kroll, N.: Assessment of preconditioning methods for multidimensional aerodynamics. Comp. Fluids **26**(6), 613–634 (1997)
5. Guillard, H., Viozat, C.: On the behaviour of upwind schemes in the low Mach number limit. Comp. Fluids **28**(1), 63–86 (1999)
6. Zamzamian, K., Razavi, S.E.: Multidimensional upwinding for incompressible flows based on characteristics. J. Comput. Phys. **227**(19), 8699–8713 (2008)

7. Harlow, F.H., Amsden, A.: A numerical fluid dynamics calculation method for all flow speeds. J. Comput. Phys. **8**(2), 197–213 (1971)
8. Karki, K.C., Patankar, S.V.: Pressure-based calculation procedure for viscous flows at all speeds in arbitrary configurations. AIAA J. **27**(9), 1167–1174 (1989)
9. Lien, F.S., Leschziner, M.A.: A pressure–velocity solution strategy for compressible flow and its application to shock boundary-layer interaction using second-moment turbulence closure. J. Fluids Eng. **115**(4), 717–725 (1993)
10. Van Leer, B.: Towards the ultimate conservation difference scheme. II. Monotonicity and conservation combined in second-order scheme. J. Comput. Phys. **14**(4), 361–370 (1974)
11. Shyy, W., Thakur, S.: Controlled variation scheme in a sequential solver for recirculating flows. Heat Transf. **25**(3, Pt.B), 273–286 (1994)
12. Thakur, S., Wright, J., Shyy, W., Liu, J., Ouyang, H., Vu, T.: Development of pressure-based composite multigrid methods for complex fluid flows. Prog. Aerospace Sci. **32**(4), 313–375 (1996)
13. Mulder, W.A., Van Leer, B.: Experiments with implicit upwind methods for the Euler equations. J. Comput. Phys. **59**, 232–246 (1985)
14. Lin, H., Chieng, C.C.: Characteristic-based flux limiters of an essentially third-order flux-splitting method for hyperbolic conservation laws. Int. J. Numer. Methods Fluids **13**(3), 287–307 (1991)
15. Issa, R.I., Javareshkian, M.H.: Application of TVD schemes in pressure-based finite-volume methods. In: Proceedings of the Fluids Engineering Division Summer Meeting, vol. 3, pp 159–164. American Society of Mechanical Engineers, New York (1996)
16. Issa, R.I., Javareshkian, M.H.: Pressure-based compressible calculation method utilizing total variation diminishing schemes. AIAA J. **36**(9), 1652–1657 (1998)
17. Kobayashi, M.H., Pereira, J.C.F.: Characteristic-based pressure correction at all speeds. AIAA J. **34**(2), 272–280 (1996)
18. Harten, A.: The artificial compression method for computation of shocks and contact discontinuities. I. Single conservation laws. Comm. Pure & Appl. Math. **30**, 611–637 (1977)
19. Harten, A.: The artificial compression method for computation of shocks and contact discontinuities: III. Self-adjusting hybrid schemes. Math. Comput. **32**(142), 363–389 (1978)
20. Yee, H., Warming, R.F., Harten, A.: Implicit total variation diminishing (TVD) schemes for steady state calculations. J. Comput. Phys. **57**(2), 327–360 (1985)
21. Yee, H.C., Sandham, N.D., Djomeri, M.J.: Low dissipative high-order shock-capturing methods using characteristic-based filters. J. Comput. Phys. **150**, 199–238 (1999)
22. Lie, K.A., Noelle, S.: On the artificial compression method for second-order nonoscillatory central difference schemes for systems of conservation laws. SIAM J. Sci. Comput. **24**(4), 1157–1174 (2003)
23. Roe, P.L.: Approximate riemann solvers, parameter vectors and difference schemes. J. Comput. Phys. **43**, 357–372 (1981)
24. Hirsch, C.H.: Numerical Computation of Internal and External Flows, vol. 2. Wiley, New York (1990)
25. Yee, H.C.: Numerical experiments with a symmetric high-resolution shock-capturing scheme. NASA TM-88325 (June 1986)
26. Harten, A., Hyman, J.M.: A self-adjusting grid for the computation of weak solutions of hyperbolic conservation laws. J. Comput. Phys. **50**, 235–269 (1983)
27. Kozel, K.: Numerical solution of some problems of external and internal aerodynamics with applications. QNET Workshop, May 29–30, Prague (2003)
28. Arnone, A., Swanson, R.C.: A navier–stokes solver for cascade flows. NASA CR 181682, ICASE Report No. 88-32 (July 1988)