AIAA JOURNAL
Vol. 41, No. 12, December 2003

Extension of a Compressible Code Toward the Incompressible Limit

Cord-Christian Rossow*
Deutsches Zentrum für Luft- und Raumfahrt, D-38108 Braunschweig, Germany

Pressure is used as a dependent variable for consistent extension toward incompressible flows. The proposed method exploits the fact that the divergence-free constraint on the velocity field for incompressible flow derives from the energy equation. In the compressible regime the implicit elliptic pressure equation transforms into an explicit hyperbolic equation, leading to the common formulation used in explicit compressible codes. The method strictly respects the strong conservation form, and reliable shock capturing is established by the use of a proven approximate Riemann solver for flux computation. For low-Mach-number flows the present approach showed improved robustness and efficiency compared to standard preconditioning techniques.

Introduction

THE extension of compressible codes toward the limit of incompressible flow is still a challenge in the development of numerical algorithms for fluid flow simulation. The main reason for the difficulties encountered is tied to a change in character of the governing equations from hyperbolic in time to elliptic, when the Mach number tends to zero. Furthermore, as was shown in Ref. 1, in the incompressible limit the discrete equations contain pressure fluctuations of the order of the Mach number, which is not the case for the continuous equations. To cope with these problems, the system of unsteady, compressible equations is preconditioned with a suitable matrix such that the disparity in the eigenvalues of the system is reduced and the integration with a time-marching process is still applicable.[2–4] Preconditioning of the unsteady equations to apply compressible codes to incompressible flow problems has received widespread attention, and a variety of suitable preconditioning matrices was derived.[2–7] However, this low-Mach-number preconditioning has not proven to be fully satisfying in the incompressible limit. In regions of very low local Mach number, for example, near stagnation points and in recirculation regions the preconditioning matrix might become singular[8] and the robustness of the computation may be impaired.

As an alternative, codes primarily designed for incompressible flows are extended toward the compressible flow regime.[9–12] In this approach, usually the continuity equation is manipulated using the discrete divergence of the momentum equations to yield a Poisson equation for a pressure correction. For the solution of incompressible flows, this formulation leads naturally to respecting the constraint of a divergence-free velocity field. This condition is not directly accounted for in the preconditioning approach when used in combination with explicit time-integration methods such as Runge–Kutta schemes.[7] The extension to compressible flows is achieved by incorporating convective terms into the pressure correction equation and to use Laplace's equation to transform the time derivative of density stemming from the continuity equation into a time derivative of pressure.[9–11] However, in most such methods the strong conservation of the governing equations is not strictly respected,[9–11] and the computation of cell interface fluxes is usually not achieved with established Riemann solvers.

To avoid the shortcomings of preconditioning and pressure-based schemes, in the present work, a method will be developed that employs a unique formulation throughout the Mach-number regime, and it shall exhibit the following features: the governing equations are solved in strong conservation form, the cell interface flux computation is based on the approximate solution of the Riemann problem, the divergence-free constraint is accounted for in the incompressible limit, and the solution method adapts to the change in character of the governing equations from hyperbolic to elliptic.

As a basis for the present work, the MAPS+ flux-splitting scheme developed in Ref. 13 will be used in combination with the findings of Ref. 14, where it was shown that the divergence-free constraint derives from the energy equation, and not from the continuity equation as usually assumed in pressure-based methods. The present approach is rigorously based on the approximate solution of the Riemann problem.[15] The performance of the present method will be assessed by computing steady incompressible and compressible, inviscid flow through nozzles and around airfoils and viscous airfoil flow.

Governing Equations

We consider the two-dimensional Navier–Stokes equations for compressible flow. The system of partial differential equations in strong conservation form is given by

$$\frac{\partial \boldsymbol{W}}{\partial t} + \frac{\partial \boldsymbol{F}}{\partial x} + \frac{\partial \boldsymbol{G}}{\partial y} = 0 \tag{1}$$

where $\boldsymbol{W}$ represents the vector of conservative variables and $\boldsymbol{F}$ and $\boldsymbol{G}$ denote the flux-density vectors for the x and y directions, respectively. Setting the viscous parts of the flux-density tensor to zero, the Euler equations governing inviscid flow are obtained:

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u)}{\partial x} + \frac{\partial (\rho v)}{\partial y} = 0$$

$$\frac{\partial (\rho u)}{\partial t} + \frac{\partial (\rho u^2 + p)}{\partial x} + \frac{\partial (\rho uv)}{\partial y} = 0$$

$$\frac{\partial (\rho v)}{\partial t} + \frac{\partial (\rho uv)}{\partial x} + \frac{\partial (\rho v^2 + p)}{\partial y} = 0$$

$$\frac{\partial (\rho E)}{\partial t} + \frac{\partial (\rho uH)}{\partial x} + \frac{\partial (\rho vH)}{\partial y} = 0$$

Presented as Paper 2003-0432 at the AIAA 41st Aerospace Sciences Meeting, Reno, NV, 6–9 January 2003; received 24 February 2003; revision received 17 June 2003; accepted for publication 3 July 2003.

*Professor, Institut für Aerodynamik und Strömungstechnik, Lilienthalplatz 7; cord.rossow@dlr.de. Senior Member AIAA.

or in divergence form

$$\frac{\partial \rho}{\partial t} + \mathrm{div}(\rho \boldsymbol{q}) = 0, \qquad \frac{\partial (\rho u)}{\partial t} + \mathrm{div}(\rho u \boldsymbol{q}) + \frac{\partial p}{\partial x} = 0$$

$$\frac{\partial (\rho v)}{\partial t} + \mathrm{div}(\rho v \boldsymbol{q}) + \frac{\partial p}{\partial y} = 0, \qquad \frac{\partial (\rho E)}{\partial t} + \mathrm{div}(\rho H \boldsymbol{q}) = 0 \tag{2}$$

where x and y denote the Cartesian coordinates; ρ, u, v, E, p, and H represent density, Cartesian velocity components, specific total energy, pressure, and specific total enthalpy, respectively; and $\boldsymbol{q}$ is the vector of the local flow velocity.

To close this system, the equation of state

$$p/\rho = R \cdot T \tag{3}$$

is used with R as specific gas constant and T as temperature. To respect the role of pressure, in this work pressure will not be computed from the equation of state, but an equation exclusively for pressure is established. If conservation shall not be violated, pressure changes $\delta(p)$ have to be computed from changes of conservative variables $\delta(\boldsymbol{W})$ by employing the chain rule:

$$\delta(p) = (\gamma - 1)[(\boldsymbol{q}^2/2)\delta(\rho) - u\delta(\rho u) - v\delta(\rho v) + \delta(\rho E)] \tag{4}$$

Using definition (4), the energy equation in the system (2) is replaced with

$$\frac{\partial p}{\partial t} + (\gamma - 1) \times \left\{ \frac{\boldsymbol{q}^2}{2}\mathrm{div}(\rho \boldsymbol{q}) - u\left[\mathrm{div}(\rho u \boldsymbol{q}) + \frac{\partial p}{\partial x}\right] - v\left[\mathrm{div}(\rho v \boldsymbol{q}) + \frac{\partial p}{\partial y}\right] + \mathrm{div}(\rho H \boldsymbol{q})\right\} = 0 \tag{5}$$

or in abbreviated form introducing $\boldsymbol{Q}(\boldsymbol{W})$ as the vector of conservative flux balances with

$$\frac{\partial p}{\partial t} + (\gamma - 1)\left[\frac{\boldsymbol{q}^2}{2}Q(\rho) - uQ(\rho u) - vQ(\rho v) + Q(\rho E)\right] = 0 \tag{6}$$

The equation of state (3) is then used to compute specific total energy or enthalpy when required. When the Mach number M approaches zero, all equations of system (2) are of order $\mathcal{O}(M^2)$ (see Ref. 1). Because of the scaling of the residuals of continuity and momentum equations with $\boldsymbol{q}^2$, u, v in Eq. (5), for $M \to 0$ the contributions of continuity and momentum equations become negligible compared to the contribution of the energy equation, and Eq. (5) reduces to

$$\frac{\partial p}{\partial t} + (\gamma - 1)[\mathrm{div}(\rho H \boldsymbol{q})] = 0 \tag{7}$$

Derivation of Computational Approach

The present method is derived by considering the discrete flux through a computational cell. Using the basic flux-difference-splitting (FDS) scheme of Ref. 15, the inviscid flux density vector $\boldsymbol{F}$ normal to a cell interface can be written as

$$\boldsymbol{F} = \tfrac{1}{2}(\boldsymbol{F}^L + \boldsymbol{F}^R) - \tfrac{1}{2}|\boldsymbol{A}| \cdot \Delta \boldsymbol{W} \tag{8}$$

where $\boldsymbol{F}^L$ and $\boldsymbol{F}^R$ are the left and right states of the inviscid flux density vector normal to the cell interface, $\boldsymbol{A}$ is the corresponding flux Jacobian, and $\Delta \boldsymbol{W}$ denotes the differences in conservative variables between left and right states of a cell interface. The expression $|\boldsymbol{A}| \cdot \Delta \boldsymbol{W}$ can be expanded in terms of the interface Mach number M_0, with M_0 defined as[13]

$$M_o = \min(|M|, 1) \cdot \mathrm{sign}(M) \tag{9}$$

The resulting expressions of $|\boldsymbol{A}| \cdot \Delta \boldsymbol{W}$ are summarized here as flux differences $\Delta \boldsymbol{F}$ for the continuity, momentum, and energy equations:

$$\Delta F_\rho = (1/c)(1 - |M_0|)\Delta p + \rho |M_0| \Delta q_n + |q_n| \Delta \rho$$

$$\Delta F_{\rho u} = n_x |M_0| \Delta p + (1/c)u(1 - |M_0|)\Delta p + n_x \rho c (1 - |M_0|)\Delta q_n + \rho u |M_0| \Delta q_n + |q_n| \Delta \rho u$$

$$\Delta F_{\rho v} = n_y |M_0| \Delta p + (1/c)v(1 - |M_0|)\Delta p + n_y \rho c (1 - |M_0|)\Delta q_n + \rho v |M_0| \Delta q_n + |q_n| \Delta \rho v$$

$$\Delta F_{\rho H} = (1/c)H(1 - |M_0|)\Delta p - |q_n|(1 - |M_0|)\Delta p + q_n \rho c (1 - |M_0|)\Delta q_n + \rho H |M_0| \Delta q_n + |q_n| \Delta \rho H$$

For compressible flows the terms scaled linearly with $|M_0|$ become dominant, whereas for incompressible flows these terms vanish and all terms scaled by $(1 - |M_0|)$ dominate.[13] To obtain the proper scaling of the numerical dissipation in the incompressible limit, the speed of sound c in the preceding equations has to be replaced by an artificial speed of sound c', which is of the order of the mean flow speed.[1,2,13] Thus, the dominance of the pressure differences scaled by $1/c'(1 - |M_0|)$ is strongly amplified in these cases. In Ref. 7 the pressure differences occurring in the continuity equation were used to derive a pressure correction equation to respect the divergence-free constraint in the incompressible limit. In the present work, however, with regard to Eq. (7) attention is focused on the enthalpy flux of the energy equation. The discrete enthalpy flux density at a cell face S can readily be computed from the preceding equations as

$$F^{\rho H} = \tfrac{1}{2} \cdot \left[\left(\rho H q_n^L + \rho H q_n^R\right) - (H/c')(1 - |M_o|)\Delta p + ODT\right] \tag{10}$$

where q_n denotes the velocity normal to the cell interface defined as $q_n = \boldsymbol{q} \cdot \boldsymbol{n}$ with $\boldsymbol{n}$ as the outward facing normal of cell face S and ODT stands for other dissipative terms as listed in the preceding equations. The pressure difference Δp denotes the difference between values on the left side L and right side R at a cell interface, giving $\Delta p = p^R - p^L$, where the normal vector $\boldsymbol{n}$ at a cell interface is pointing from left to right when the cell boundary is traversed in a mathematically positive sense. The corresponding discrete enthalpy flux through the cell face is obtained by multiplying Eq. (10) by the cell face area S:

$$F^{\rho H} \cdot S = \tfrac{1}{2} \cdot \left[\left(\rho H q_n^L + \rho H q_n^R\right) - (H/c')(1 - |M_o|)\Delta p + ODT\right] \cdot S \tag{11}$$

Using the FDS discretization[15] to obtain the intermediate state at cell interfaces, the discrete divergence div_δ of $\rho H \boldsymbol{q}$ in a cell can be expressed as

$$\mathrm{div}_\delta(\rho H \boldsymbol{q}) = \frac{1}{vol} \cdot \sum_{i=1}^{i\,\mathrm{faces}} F_i^{\rho H} \cdot S_i = \frac{1}{vol} \cdot \sum_{i=1}^{i\,\mathrm{faces}} \times \frac{1}{2} \cdot \left[\left(\rho H q_n^L + \rho H q_n^R\right) - \frac{H}{c'}(1 - |M_o|)\Delta p + ODT\right]_i \cdot S_i \tag{12}$$

where i faces denotes the number of faces of the computational cell with i as the running index and vol is the corresponding cell volume.

Using definition (12), the discrete analog of Eq. (5) or (6) is expressed as

$$\delta p/\delta t + (\gamma - 1)[(\boldsymbol{q}^2/2)Q_\delta(\rho) - uQ_\delta(\rho u) - vQ_\delta(\rho v) + \mathrm{div}_\delta(\rho H \boldsymbol{q})] = 0 \tag{13}$$

where $\boldsymbol{Q}_\delta(\rho, \rho u, \rho v)$ denote the discrete flux balances of the continuity and momentum equations, respectively. Equation (13) can be rearranged with use of Eq. (12) to yield

$$\frac{\delta p}{\delta t} = -(\gamma - 1) \times \left\{ \frac{1}{vol} \cdot \sum_{i=1}^{ifaces} \times \frac{1}{2} \cdot \left[\left(\rho H q_n^L + \rho H q_n^R \right) - \frac{H}{c'}(1 - |M_o|)\Delta p + ODT \right]_i S_i - \frac{\boldsymbol{q}^2}{2} Q_\delta(\rho) - u\, Q_\delta(\rho u) - v\, Q_\delta(\rho v) \right\} \tag{14}$$

To compute incompressible flows, a pressure correction equation will be derived from Eq. (14), and the relevant pressure terms in this regime have to be identified. First, as already mentioned, for $M \to 0$ the pressure differences Δp of the dissipative terms scaled inversely with the preconditioned speed of sound c' dominate. The pressure differences of the momentum equations are also scaled linearly with the Cartesian velocity components. In the pressure equation (14) contributions from continuity and momentum equations are multiplied by $\boldsymbol{q}^2$, u, v, respectively. Thus, for $M \to 0$ in Eq. (14) only the pressure differences of the energy equation remain relevant. Introducing now an implicit formulation for pressure with respect to the relevant terms in the incompressible regime by using $p^{n+1} = p^n + \delta p$, it follows from Eq. (14):

$$\frac{\delta p}{\delta t} - (\gamma - 1)\frac{1}{\mathrm{vol}} \cdot \sum_{i=1}^{ifaces} \frac{1}{2} \cdot \left[\frac{H}{c'}(1 - |M_o|)\Delta(\delta p) \right]_i S_i = \mathrm{RHS} \tag{15}$$

where the right-hand side (RHS) is defined by the explicit flux balances as

$$\mathrm{RHS} = -(\gamma - 1)\left[(\boldsymbol{q}^2/2) Q_\delta(\rho) - u\, Q_\delta(\rho u) - v\, Q_\delta(\rho v) + \mathrm{div}_\delta(\rho H \boldsymbol{q}) \right] \tag{16}$$

Equation (15) reflects the role of pressure for compressible and incompressible flow in the discrete formulation: For compressible flow at Mach numbers above one, the second term on the left-hand side of Eq. (15) vanishes, and Eq. (15) becomes purely explicit. The system of Eqs. (2) can be solved simultaneously with any explicit time-stepping scheme, and the simultaneous solution of the system (2) reflects the tight coupling of all variables in compressible flow. However, when the Mach number tends to zero Eq. (15) requires an implicit solution for p. Pressure is not advanced in time in the same explicit manner as the other variables any more, but because of the implicit formulation pressure fluctuations influence the whole domain at every time step. This reflects the fact that for $M \to 0$ pressure fluctuations propagate infinitely fast compared to the advection speed $\boldsymbol{q}$.

For incompressible flow, by virtue of Eq. (7), Eq. (15) becomes a Poisson equation for pressure similar to the pressure correction equation derived in Ref. 7. The difference between Eq. (15) in the incompressible limit and the pressure correction equation of Ref. 7 is that in Eq. (15) the enthalpy flux $\rho H \boldsymbol{q}$ is regarded and not the mass flux $\rho \boldsymbol{q}$ because in Ref. 7 the Poisson equation was derived from the continuity equation. For inviscid flow the divergence-free constraint for the incompressible velocity field, $\mathrm{div}(\boldsymbol{q}) = 0$, derives from the energy equation, but not from the continuity equation.[14] In fact, the continuity equation becomes an advection equation for density fluctuations as $M \to 0$ (Refs. 1, 16, and 17). For $M \to 0$ the right-hand side of Eq. (15) reduces to the divergence of enthalpy flux, that is, the flux balance of the energy equation. Because for low-Mach-number flows the energy equation expresses that $\mathrm{div}(\boldsymbol{q})$ is going to zero,[17] it can be expected that Eq. (15) will account for the divergence-free constraint similarly as the pressure correction equation derived from the continuity equation.

Implementation of the Current Approach

The approach just derived is implemented into the framework of explicit Runge–Kutta time integration. In conventional compressible codes with Runge–Kutta time-stepping schemes, the set of governing equations (1) is solved simultaneously to update the conservative variables $\boldsymbol{W}$:

$$\boldsymbol{W}_j^{(m)} = \boldsymbol{W}_j^{(0)} + \alpha^{(m)} \cdot \boldsymbol{R}_j\left(\boldsymbol{W}^{(m-1)}\right) \tag{17}$$

where the superscripts (m) denote the stage count with m running from one to the maximum number of stages; the subscripts j correspond to the location at mesh points in the flowfield; $\alpha^{(m)}$ is the stage coefficient of the (m) stage; and $\boldsymbol{R}_j(\boldsymbol{W}^{(m-1)})$ represent the conservative residuals of continuity, momentum, and energy equations evaluated using the flow variables from the preceding $(m-1)$ stage.

To solve for another set of variables $\boldsymbol{U} = (\rho, \rho u, \rho v, p)^T$, Eq. (17) becomes

$$\boldsymbol{U}_j^{(m)} = \boldsymbol{U}_j^{(0)} + \alpha^{(m)} \cdot \boldsymbol{R}_j\left(\boldsymbol{U}^{(m-1)}\right) \tag{18}$$

where the explicit residual for the pressure equation is obtained after evaluation of the conservative flux balances $\boldsymbol{Q}(\boldsymbol{W})$ and application of the chain rule following Eq. (4).

To implement the formulation of the pressure equation (15) into the Runge–Kutta scheme (18), in each stage m a preliminary pressure update δp_j is computed from Eq. (15). In the present work Eq. (15) is solved with a point Jacobi method, and analogously to Ref. 7, 10 iterations are used as default. Having computed δp_j, pressure is then updated in the Runge–Kutta framework as

$$p_j^{(m)} = p_j^0 + \alpha^{(m)} \cdot \delta p_j \tag{19}$$

For Mach numbers greater than unity, the second term on the left-hand side of Eq. (15) vanishes, and the pressure update from Eq. (19) becomes purely explicit and identical to the update by the basic Runge–Kutta scheme of Eq. (18). In such cases the effort for solving Eq. (15) is discarded, which leads to unnecessary computational overhead. To avoid this overhead, a combination with implicit residual smoothing techniques will be established in the following.

The convergence of explicit compressible codes toward steady state can be accelerated by implicit residual smoothing,[18] and in Ref. 19 an implicit smoothing technique applicable to unstructured meshes was derived on the basis of the MAPS discretization. The smoothed residual $\tilde{\boldsymbol{R}}$ is obtained by solving[19]

$$\tilde{\boldsymbol{R}}_j - \frac{\delta t_j}{Vol_j} \sum_{i=1}^{ifaces} \varepsilon_i \Delta \tilde{\boldsymbol{R}} = \boldsymbol{R}_j \tag{20}$$

and the smoothing coefficients ε_i on the cell faces are defined as

$$\varepsilon_i = \tfrac{1}{2}[M_{\mathrm{up}} + (1 - |M_0|)(c'/c)]_i c_i S_i \tag{21}$$

The upwind face Mach number M_{up} is given in Eq. (23).

Equation (20) is solved in each stage of the Runge–Kutta scheme by point Jacobi iteration, and usually 6–10 iterations are employed.[19] The application of implicit smoothing via Eq. (20) allows an increase of Courant–Friedrichs–Lewy (CFL) number of the basic explicit scheme by a factor of 2–3.

The smoothing coefficient defined in Eq. (21) can be integrated into Eq. (15) for the update of pressure:

$$\frac{\delta p}{\delta t} - (\gamma - 1)\frac{1}{vol} \cdot \sum_{i=1}^{ifaces} \frac{1}{2}\left(\left\{ \frac{H}{c'}(1 - |M_o|) + \left[M_{\mathrm{up}} + (1 - |M_0|)\frac{c'}{c} \right] c \right\} \Delta(\delta p) \right)_i S_i = \mathrm{RHS} \tag{22}$$

Using Eqs. (19) and (22) for the update of pressure and Eqs. (18) and (20) to update density and momentum, leads for compressible flow to the implicit smoothing proposed in Ref. 19, and for incompressible flows pressure is updated by an equation similar to the pressure correction equation derived in Ref. 7. Note that the numerical effort

to solve Eq. (22) is identical to the effort required in the original formulation of Ref. 19 for implicit smoothing of the energy equation. The only difference is the new smoothing coefficient defined in Eq. (22). By using 10 Jacobi iterations as a default in the present approach, the computational effort of one time step is comparable to a time step of the original method.[19] Any reduction in the number of time steps required to reach convergence therefore directly reduces total computation time. The conceptual difficulties encountered in Ref. 7 to efficiently cover both compressible and incompressible flow regimes are overcome with the present approach.

Throughout this work the MAPS+ approximate Riemann solver[13] is used for spatial discretization. The MAPS+ discretization is an extension of the MAPS (Mach-number-based advection pressure splitting) flux-splitting scheme,[20] which was designed employing elements of other hybrid flux-splitting formulations.[21,22] The MAPS formulation yielded a very simple and robust scheme for high-speed flows, and it does not violate the entropy condition, in contrast to other flux-splitting schemes.[20,23] MAPS was then extended to low-speed flows by adding cross-diffusion terms identified in the classical FDS scheme.[15] The resulting MAPS+ scheme was shown to be very similar to the FDS discretization[15]; however, no Roe-averaging or entropy correction is required. For completeness the dissipative terms of the MAPS+ discretization[13] are given here:

$$\Delta F_\rho = (1/c^{\max})(1-|M_0|)\Delta p + \rho\beta^M \Delta q_n + |q_n|\Delta\rho$$

$$\Delta F_{\rho u} = n_x\beta^p\Delta p + (1/c^{\max})u(1-|M_0|)\Delta p + n_x\rho^{\min}c^{\min} \times (1-|M_0|)\Delta q_n + \rho u\beta^M\Delta q_n + |q_n|\Delta\rho u$$

$$\Delta F_{\rho v} = n_y\beta^p\Delta p + (1/c^{\max})v(1-|M_0|)\Delta p + n_y\rho^{\min}c^{\min} \times (1-|M_0|)\Delta q_n + \rho v\beta^M\Delta q_n + |q_n|\Delta\rho v$$

$$\Delta F_{\rho H} = (1/c^{\max})H^{\min}(1-|M_0|)\Delta p + \rho H\beta^M\Delta q_n + |q_n|\Delta\rho H$$

$$|M_0| = \min[\max(|M^L|, |M^R|), 1]$$

$$c^{\max} = \max(c^L, c^R), \qquad c^{\min} = \min(c^L, c^R)$$

$$\rho^{\min} = \min(\rho^L, \rho^R), \qquad H^{\min} = \min(H^L, H^R)$$

$$u = 0.5\cdot(u^L + u^R), \qquad v = 0.5\cdot(v^L + v^R)$$

$$\rho = 0.5\cdot(\rho^L + \rho^R), \qquad \rho u = 0.5\cdot(\rho^L u^L + \rho^R u^R)$$

$$\rho v = 0.5\cdot(\rho^L v^L + \rho^R v^R), \qquad \rho H = 0.5\cdot(\rho^L H^L + \rho^R H^R)$$

$\beta^M \rightarrow$ see Eq. (23), $\beta^p \rightarrow$ see Eq. (23), $|q_n| \rightarrow$ see Eq. (23)

The functions β^M, β^P, and $|q_n|$ are defined by[13]

$$\begin{aligned}
\beta^M &= \min(0, 2\cdot M^{\max 1} - 1)\\
M^{\max 1} &= \min[\max(|M^L|, |M^R|), 1]\\
\beta^P &= \max(0, 2\cdot M^{\min 1} - 1)\\
M^{\min 1} &= \min[\min(|M^L|, |M^R|), 1]\\
|q_n| &= M_{\text{up}}\cdot c_{\text{av}}\\
M_{\text{up}} &= \max(|M^L|, |M^R|)\\
c^{\text{av}} &= 0.5(c^L + c^r)
\end{aligned} \tag{23}$$

For incompressible flows, the MAPS+ and the FDS scheme become identical. To obtain the correct scaling of dissipation in the incompressible limit,[13] the speed of sound c in the MAPS+ dissipation outlined has to be replaced by an artificial speed of sound c':

$$c' = \sqrt{\alpha^2\boldsymbol{q}^2 + M_r^2 c^2}, \qquad \alpha = \tfrac{1}{2}\left(1 - M_r^2\right) \tag{24}$$

where M_r is a reference Mach number defined as

$$M_r^2 = \min\left\{\max\left[\boldsymbol{q}^2/c^2, \quad k\left(\boldsymbol{q}_\infty^2/c_\infty^2\right)\right], 1\right\} \tag{25}$$

with k usually set to unity. The artificial speed of sound c' is also used to compute the allowable time step.

Application of the Method

First, the present approach is used to compute one-dimensional nozzle flow. The computational meshes consisted of 160 cells along the nozzle axis. To clearly identify capabilities and limits, only a first-order implementation was investigated, and for time integration a three-stage Runge–Kutta scheme without any acceleration techniques except local time stepping was used. Thus, for compressible flow with Mach numbers greater than unity the original Runge–Kutta time stepping is recovered [see Eq. (15)]. Results obtained with the basic compressible code using the MAPS+ discretization[13] with preconditioning for incompressible flows served as reference. The first test case is the nearly incompressible flow in a converging/diverging nozzle with an inflow Mach number of $M_{\text{in}} = 0.00845$. Figure 1 shows the Mach-number distribution along the nozzle axis. Because the pressure equation is based on the conservative flux balances according to Eq. (5), the converged results of the reference code and the present approach are identical. The solution on a very fine mesh with 640 cells along the axis is given as reference, and it can be concluded that 160 cells are sufficient to resonably resolve the flow with the first-order method. In Fig. 2 the convergence histories for the L^2 norm of the density residual are displayed. In both cases a CFL number of 1.0 was used, and for the reference solution the preconditioning matrix of Ref. 7 was employed. In the second test case compressible flow in a Laval nozzle was computed, where the backpressure was adjusted such that the preshock Mach number of $M = 2.9$ is identical to that used in other investigations.[7,13,20] Figure 3 shows the Mach-number distribution in the Laval nozzle for both the reference code and the present approach. The identical convergence behavior shown in Fig. 4 confirms the statement that

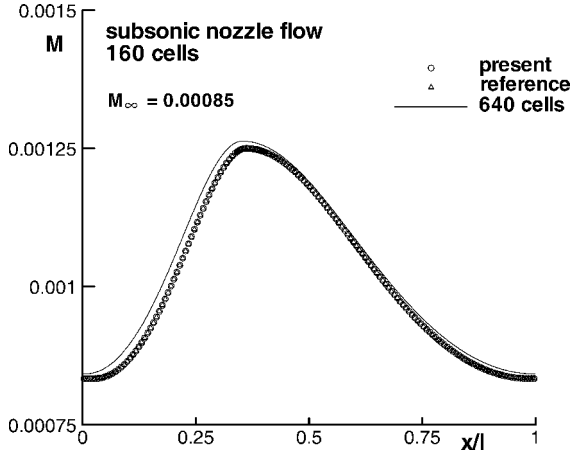

Fig. 1 Subsonic Mach-number distribution in converging/diverging nozzle.

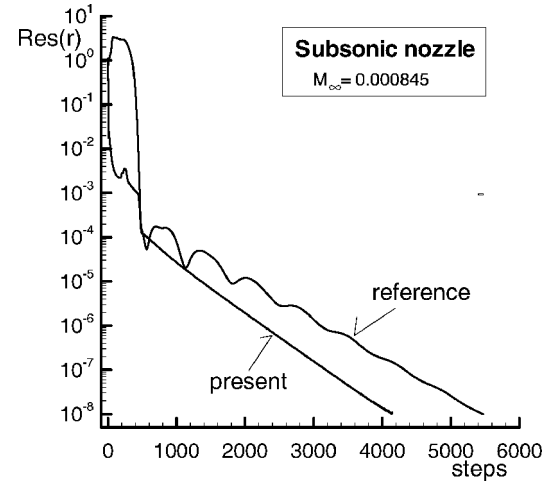

Fig. 2 Convergence histories for subsonic flow in converging/diverging nozzle.

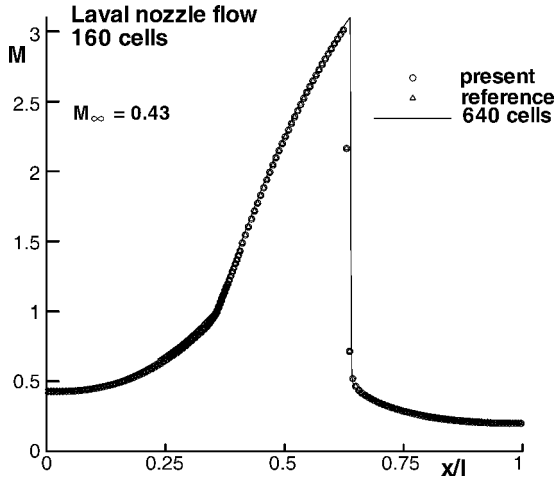

Fig. 3 Mach-number distribution for one-dimensional Laval nozzle flow.

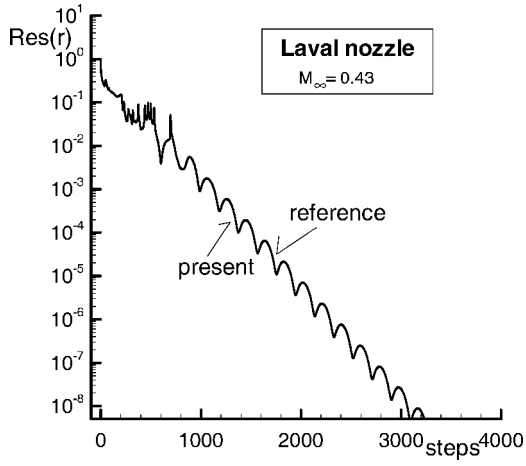

Fig. 4 Convergence histories for one-dimensional Laval nozzle flow.

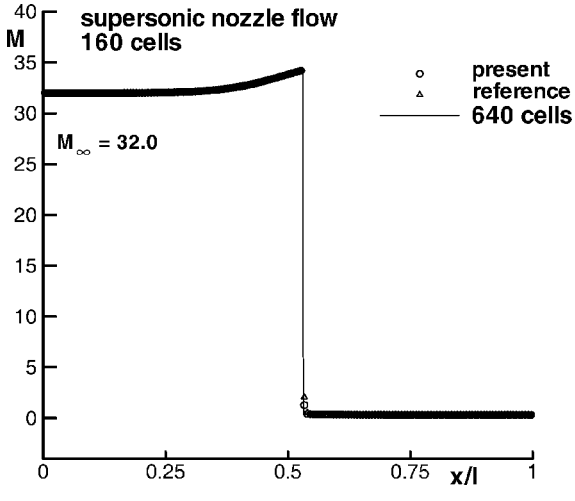

Fig. 5 Mach-number distribution for supersonic nozzle flow.

at high speed the present approach reverts to the explicit compressible formulation. Note that in Ref. 7 it was observed that the purely pressure-based approach failed for this case, and a special blending technique had to be introduced. To give further evidence that the present approach consistently blends to a compressible formulation suitable for reliable shock capturing, nozzle flow with a preshock Mach number of $M = 32$ was computed, similarly to prior investigations of compressible codes.[13,20] The axial Mach-number distribution and convergence behavior is shown in Figs. 5 and 6.

From the computation of these one-dimensional test cases, it can be concluded that the present approach is well suited for computations over the whole Mach-number range, without any artificial adaptation of the formulation to a specific flow regime.

Next, the method is applied to inviscid flow around the NACA0012 airfoil. For the two-dimensional problems the formulation with implicit smoothing according to Eqs. (18–22) was used. The implementation as described in the preceding section was established in the unstructured NOUGAT (Node Oriented Unstructured Generic Algorithm Testing) code.[7,19] This code employs a node-based scheme with dual-mesh metrics, the spatial discretization is based on the MAPS+ flux-splitting scheme, and least-squares gradients are used for second-order reconstruction. Time integration is achieved by a five-stage Runge–Kutta scheme, and convergence toward steady state is accelerated by local time stepping, agglomeration multigrid, and implicit residual smoothing as given in Eq. (20). Usually, CFL numbers of 6 are employed with this scheme.

For the computations a structured O mesh with 160 cells around the airfoil and 32 cells in normal direction was chosen. To check the behavior of the present scheme with varying Mach number, for an angle of attack of $\alpha = 0$ deg freestream Mach number was varied from $M_\infty = 0.001$ to 0.85. No multigrid acceleration was employed, and the allowable CFL number was 6, as in the reference code with preconditioning.[7,19] Figure 7 displays surface-pressure distributions obtained for selected Mach numbers, which illustrate the change in the flow characteristics. The convergence behavior of the present method and the reference code with preconditioning can be assessed

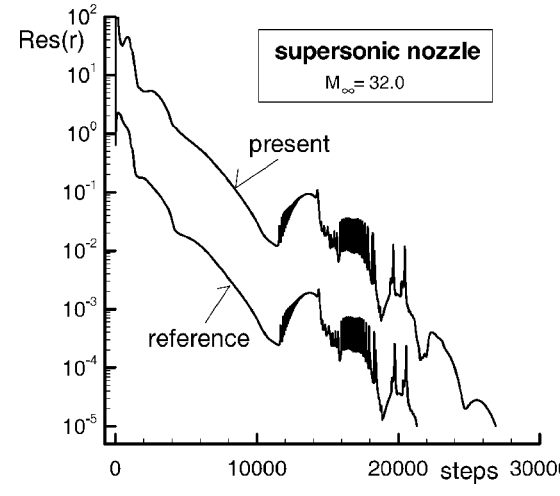

Fig. 6 Convergence histories for supersonic nozzle flow.

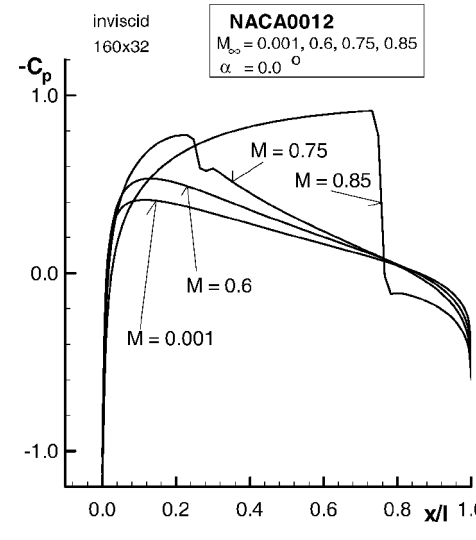

Fig. 7 Surface-pressure distribution for Mach-number variation of inviscid flow around NACA0012 airfoil at $\alpha = 0$ deg.

Table 1 Drag coefficient for grid-refinement study of inviscid flow around NACA0012 at $\alpha = 0$ deg

	Mach number										
Grid	0.001	0.01	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.75	0.85
40×8	0.01875	0.01875	0.01882	0.01906	0.01950	0.02021	0.02135	0.02300	0.02549	0.02776	0.06780
80×16	0.00482	0.00482	0.00483	0.00487	0.00493	0.00502	0.00517	0.00541	0.00578	0.00615	0.05286
160×32	0.00096	0.00096	0.00096	0.00096	0.00096	0.00097	0.00097	0.00098	0.00099	0.00103	0.04748

Table 2 Influence of number of Jacobi iterations on convergence

	Case			
Method	$M = 0.001$ Iter = 10	$M = 0.001$ Iter = 20	$M = 0.85$ Iter = 10	$M = 0.85$ Iter = 20
Present	814	554	4442	3900
Reference	1437	1430	4225	4215

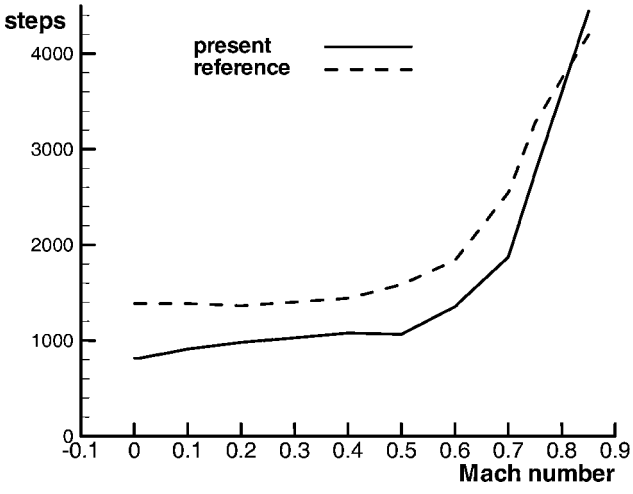

Fig. 8 Number of time steps as function of freestream Mach number for inviscid flow around NACA0012 airfoil.

from Fig. 8, where the time steps required to reduce the L^2 norm of the density residual by eight orders of magnitude are displayed as a function of freestream Mach number. At lower Mach numbers the present approach converged significantly faster than the reference code with preconditioning. This is a result of the implicit treatment of pressure through Eq. (15), which accounts at low Mach number for the divergence-free constraint. The numerical effort per time step is similar for both codes because the default number of 10 Jacobi iterations is used throughout and the reduction in the number of time steps leads to a corresponding reduction of total computation time. For higher Mach numbers the convergence rates of the different approaches become similar, as might be expected because the present approach becomes similar to the original compressible formulation.

Successive mesh refinement is used to demonstrate the accuracy of the method. Table 1 shows the convergence of the drag coefficient with mesh density. Drag converges with second order with mesh refinement, confirming second-order accuracy of the spatial discretization.[7]

To assess the influence of the number of Jacobi iterations on convergence, for $M_\infty = 0.001$ and $M_\infty = 0.85$ the number of Jacobi iterations was increased from 10 to 20. The results are summarized in Table 2. With the present approach, for the incompressible case the number of time steps is reduced by about 30% and for the compressible case only by about 10%. For the reference code the number of time steps is almost unaffected. This confirms the default of 10 Jacobi iterations as a good compromise to efficiently cover incompressible and compressible flows.

In the next case the robustness of the method was assessed. Following the route of investigations in Ref. 7, at $M_\infty = 0.001$ the angle of attack was increased from $\alpha = 12$ deg to 14 deg and 16 deg. In all cases computations were started from freestream conditions. The

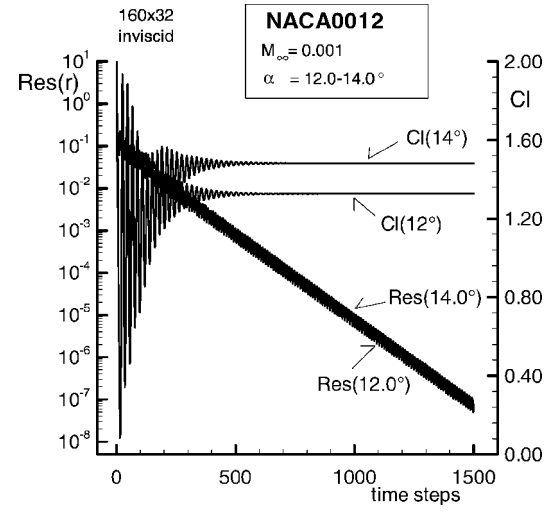

Fig. 9 Convergence for inviscid, incompressible flow around NACA0012 airfoil at $\alpha = 12$–14 deg with present approach.

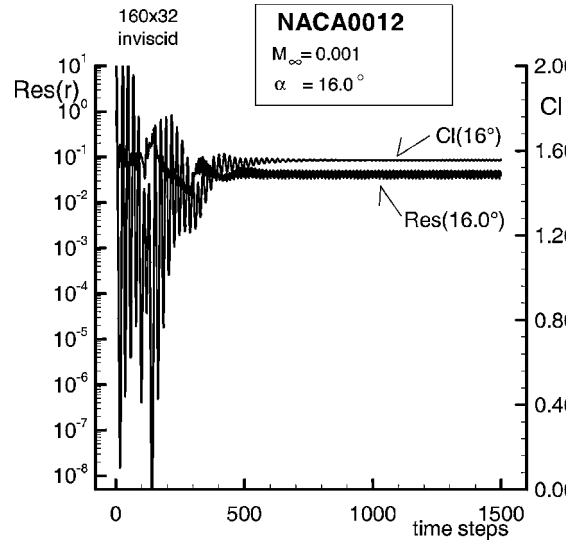

Fig. 10 Convergence for inviscid, incompressible flow around NACA0012 airfoil at $\alpha = 16$ deg with present approach.

corresponding convergence behavior of the L^2 norm of the density residual and lift are shown in Figs. 9 and 10. For $\alpha = 12$ and 14 deg convergence rates are similar. At $\alpha = 16$ deg the residual stalled, but total lift converged to a steady value. In Ref. 7 the same cases were computed using the reference code with preconditioning and using a pressure correction method. The pressure correction method achieved converged solutions up to $\alpha = 14$ deg, and a lift-converged solution for $\alpha = 16$ deg. The reference code with preconditioning did not yield a stable solution even for $\alpha = 14$ deg. In Ref. 7 the difficulties at higher angles of attack were traced back to onset of separation triggered by numerical dissipation, and it was concluded that the respect of the divergence-free constraint in the pressure-based method was the primary reason for increased robustness. The results obtained with the present method suggest that the divergence-free constraint is respected in the present formulation in the same manner as in the pressure-based method.

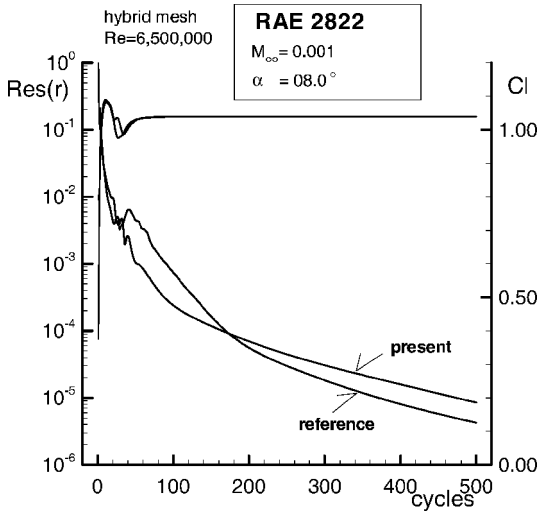

Fig. 11 Convergence for viscous, incompressible flow around RAE2822 airfoil computed with present and preconditioned reference method.

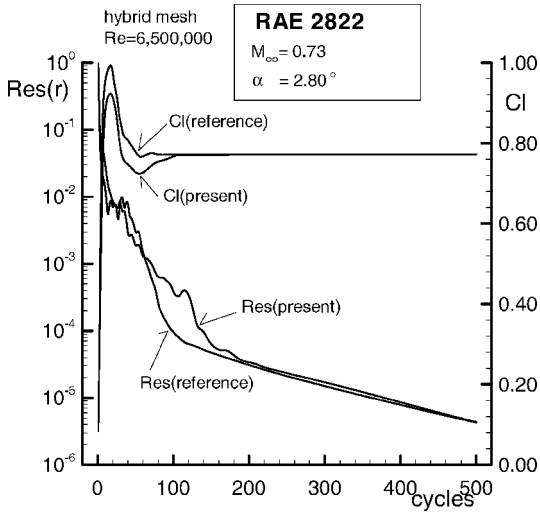

Fig. 12 Convergence for viscous, compressible flow around RAE2822 airfoil computed with present and preconditioned reference method.

In the next test cases turbulent viscous flow around the RAE2822 airfoil at $M_\infty = 0.001$, $\alpha = 8$ deg, and $Re = 6.5 \times 10^6$ was computed using a hybrid mesh. The hybrid mesh was identical to the mesh used in Refs. 7 and 19 and consisted of about 23,000 quadrilateral and triangular cells, where the 4800 quadrilateral cells were used in the vicinity of the airfoil to resolve the viscous region, with 192 of these cells located directly on the airfoil surface. A four-level agglomeration multigrid was used to accelerate convergence, and the influence of turbulence was modeled following Spalart and Allmaras.[24] Figure 11 shows that the convergence of the new method is comparable to that of the standard compressible code with preconditioning. Steady-state results are identical to those obtained with the compressible, preconditioned reference code,[7,19] which is also indicated by the convergence to the same final lift coefficient in Fig. 11.

Finally, transonic flow is computed for $M_\infty = 0.73$ and $\alpha = 2.8$ deg at the same Reynolds number. The convergence rate displayed in Fig. 12 is again similar to that of the reference code, and because of the use of conservative flux balances the steady-state solution is also identical to those of Refs. 7 and 19.

Conclusions

In the present work a computational approach was derived, which consistently covers incompressible and compressible flows. This is achieved by using pressure as a dependent variable while still strictly respecting conservation properties. For compressible flows the method is similar to standard explicit codes solving the compressible equations. For incompressible flows an implicit solution for pressure is established. The implicit formulation is derived from pressure terms arising in the enthalpy flux balance when the Mach number approaches zero with the pressure terms identified from the approximate solution of the Riemann problem. The resulting equation is similar to a pressure correction equation, which is however derived from the energy equation. This reflects the fact stated in the literature that the divergence-free constraint on the velocity field for incompressible flow derives from the energy equation and not from the continuity equation.

Computation of one-dimensional nozzle flows confirmed that the present approach is capable of covering the whole Mach-number range. The excellent shock-capturing capabilities and the robustness of the basic approximate Riemann solver were not compromised in the present formulation. Compared to a reference code with preconditioning and Runge–Kutta time integration, improved efficiency and robustness of the present approach were demonstrated for inviscid, incompressible flow around airfoils. Finally, the applicability to viscous flow was shown for incompressible and transonic flow around airfoils.

Acknowledgments

Parts of this study were conducted when the author was in residence at the former ICASE, and the author expresses his thanks to Manuel Salas, former ICASE director, for providing this opportunity.

References

[1]Guillard, H., and Viozat, C., "On the Behaviour of Upwind Schemes in the Low Mach Number Limit," *Computers and Fluids*, Vol. 28, 1999, pp. 63–86.

[2]Turkel, E., "Preconditioning Techniques in Computational Fluid Dynamics," *Annual Review of Fluid Mechanics*, Vol. 31, 1999, pp. 385–416.

[3]Choi, Y.-H., and Merkle, C. L., "The Application of Preconditioning to Viscous Flows," *Journal of Computational Physics*, Vol. 105, 1993, pp. 207–223.

[4]Weiss, J. M., and Smith, W. A., "Preconditioning Applied to Variable and Constant Density Flows," *AIAA Journal*, Vol. 33, 1995, pp. 2050–2057.

[5]Lee, D., "Design Criteria for Local Euler Preconditioning," *Journal of Computational Physics*, Vol. 144, 1998, pp. 423–459.

[6]Lee, D., "The Design of Local Navier–Stokes Preconditioning for Compressible Flow," *Journal of Computational Physics*, Vol. 144, 1998, pp. 460–483.

[7]Rossow, C.-C., "A Blended Pressure/Density Based Method for the Computation of Incompressible and Compressible Flows," *Journal of Computational Physics*, Vol. 185, 2003, pp. 375–398.

[8]Darmofal, D. L., and Van Leer, B., "Local Preconditioning: Manipulating Mother Nature to Fool Father Time," *Computing the Future II: Advances and Prospects for Computational Aerodynamics*, edited by D. Caughey and M. Hafez, Wiley, New York, 1998, pp. 211–239.

[9]Karki, K. C., and Patankar, S. V., "Pressure Based Calculation Procedure for Viscous Flows at all Speed in Arbitrary Configurations," *AIAA Journal*, Vol. 27, 1989, pp. 1167–1174.

[10]Demirdzic, I., Lilek, Z., and Peric, M., "A Collocated Finite Volume Method for Predicting Flows at All Speeds," *International Journal of Numerical Methods in Fluids*, Vol. 16, 1993, pp. 1029–1050.

[11]Bijl, H., and Wesseling, P., "A Unified Method for Computing Incompressible and Compressible Flows in Boundary Fitted Coordinates," *Journal of Computational Physics*, Vol. 141, 1998, pp. 153–177.

[12]Mathur, S. R., and Murthy, J. Y., "All Speed Flows on Unstructured Meshes Using a Pressure Correction Approach," AIAA Paper 99-3365, July 1999.

[13]Rossow, C.-C., "A Flux Splitting Scheme for Compressible and Incompressible Flows," *Journal of Computational Physics*, Vol. 164, 2000, pp. 104–122.

[14]Klein, R., "Semi-Implicit Extension of a Godunov-Type Scheme Based on Low Mach Number Asymptotics I: One-Dimensional Flow," *Journal of Computational Physics*, Vol. 121, 1995, pp. 213–237.

[15]Roe, P. L., "Approximate Riemann Solvers, Parameter Vectors and Difference Schemes," *Journal of Computational Physics*, Vol. 43, 1981, pp. 357–372.

[16]Schneider, T., Botta, N., Geratz, K. J., and Klein, R., "Extension of Finite Volume Compressible Flow Solvers to Multi-Dimensional, Variable Density

Zero Mach Number Flows," *Journal of Computational Physics*, Vol. 155, 1999, pp. 248–286.

[17]Vierendeels, J., Riemslagh, K., and Dick, E., "A Multigrid Semi-Implicit Line-Method for Viscous Incompressible and Low-Mach-Number Flows on High Aspect Ration Grids," *Journal of Computational Physics*, Vol. 154, 1999, pp. 310–341.

[18]Jameson, 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, June 1981.

[19]Rossow, C.-C., "Convergence Acceleration on Unstructured Meshes," *Contributions to the 12th STAB/DGLR Symposium Stuttgart 2000, Notes on Numerical Fluid Mechanics*, edited by S. Wagner, U. Rist, J. Heinemann, and R. Hilbig, Vol. 77, Springer, Berlin, 2002, pp. 304–311.

[20]Rossow, C.-C., "A Simple Flux-Vector Splitting Scheme for Compressible Flows," *Contributions to the 11th STAB/DGLR Symposium Berlin 1998, Notes on Numerical Fluid Mechanics*, edited by W. Nitsche, J. Heinemann, and R. Hilbig, Vol. 72, Vieweg, Braunschweig, Germany, 1999, pp. 355–362.

[21]Edwards, J. R., "A Low-Diffusion Flux-Splitting Scheme for Navier–Stokes Calculations," *Computers and Fluids*, Vol. 26, 1997, pp. 653–659.

[22]Jameson, A., "Artificial Diffusion, Upwind Biasing, Limiters and Their Effect on Accuracy and Multigrid Convergence in Transonic and Hypersonic Flow," AIAA Paper 93-3559, July 1993.

[23]Moschetta, J.-M., and Gressier, J., "The Sonic Point Glitch Problem: A Numerical Solution," *Proceedings of the 16th International Conference on Numerical Methods in Fluid Dynamics*, Lecture Notes in Physics, Springer, Arcachon, France, 1998.

[24]Spalart, P. R., and Allmaras, S. R., "A One-Equation Turbulence Model for Aerodynamic Flows," *La Recherche Aerospatiale*, No. 1, 1994, pp. 5–21.

P. Givi
Associate Editor