INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 16, 1029–1050 (1993)

A COLLOCATED FINITE VOLUME METHOD FOR PREDICTING FLOWS AT ALL SPEEDS

I. DEMIRDŽIĆ

Mašinski fakultet, Omladinsko šetalište bb, 71000 Sarajevo, Bosnia-Hercegowina

AND

Ž. LILEK AND M. PERIĆ*

Lehrstuhl für Strömungsmechanik, Cauerstrasse 4, D-8520 Erlangen, F. R. Germany

SUMMARY

An existing two-dimensional method for the prediction of steady-state incompressible flows in complex geometry is extended to treat also compressible flows at all speeds. The primary variables are the Cartesian velocity components, pressure and temperature. Density is linked to pressure via an equation of state. The influence of pressure on density in the case of compressible flows is implicitly incorporated into the extended SIMPLE algorithm, which in the limit of incompressible flow reduces to its well-known form. Special attention is paid to the numerical treatment of boundary conditions. The method is verified on a number of test cases (inviscid and viscous flows), and both the results and convergence properties compare favourably with other numerical results available in the literature.

KEY WORDS Finite volume Compressible flow Subsonic flow Transonic flow Supersonic flow

1. INTRODUCTION

Most of the traditional numerical methods developed for compressible flow simulation use an unsteady form of the Navier–Stokes or Euler equations.[1–6] These methods use density as the primary variable and pressure is determined via an equation of state. Their application in cases of incompressible or low Mach number flows is questionable, since in that situation the density changes are very small and the pressure–density coupling becomes very weak. Some possible ways to surmount this difficulty lie in a fictitious equation of state or artificial compressibility.[7]

On the other hand, methods for incompressible flows are mostly of the pressure correction type and use pressure as the primary variable.[8] These methods are well established, with many variations being possible depending upon the choice of the dependent variables and their arrangement, numerical grid, pressure correction algorithm, differencing schemes, etc. (cf. References 8–12).

Numerical methods which are applicable to flows at all speeds are less numerous. An important step in this direction was reported by Hirt *et al.*,[13] who used Cartesian velocity components stored at control volume corners and scalar variables stored at cell centres. Their

* Author to whom correspondence should be addressed.

0271–2091/93/121029–22$16.00

Received March 1992
Revised January 1993

method suffered from oscillations due to pressure–velocity decoupling. More recently, Karki and Patankar[14] and Demirdžić *et al.*[15] presented solution methods based on modified pressure–velocity coupling algorithms of the SIMPLE type which include the compressibility effects and are applicable to flows at all speeds. Karki and Patankar[14] use locally fixed base vectors, while Demirdžić *et al.*[15] use variable base vectors and contravariant vector and tensor components. Both methods suffer from sensitivity to grid smoothness, due to the presence (directly or indirectly) of curvature terms in the equations. This sensitivity is manifested by increased total pressure loss and kinks on the isolines wherever the grid lines change their directions.

The present work is an extension of the finite volume method which was developed for predicting incompressible flows in complex two- and three-dimensional geometries.[10,12] The aim was to obtain a method which, with good accuracy, stability and convergence properties, can be used to predict flows at all speeds.

The primary variables are the Cartesian velocity components, pressure and temperature; the density is linked to pressure via an equation of state. Pressure–velocity coupling is achieved through a modified SIMPLE algorithm[17] which now takes into account the density variation in a manner similar to that employed in References 14–16. The discretization is carried out on control volumes defined by a boundary-fitted, non-orthogonal grid. Since Cartesian base vectors are employed, the method is not sensitive to grid smoothness. Diffusion fluxes are approximated by central differences. In order to increase the stability of the computational procedure while preserving the numerical accuracy, central differences (CDs) are blended with upwind differences (UDs) when discretizing the convection terms in all equations. The contribution of CDs is kept as high as possible, typically above 90%. This practice leads to a good resolution of shocks while providing faster convergence than the first-order UDs.

The method is verified by applying it to some one-dimensional problems and to the well-known GAMM test cases: inviscid flow in a channel with circular arc bump and viscous flow in a double-throat nozzle. Finally, as an illustration of the capabilities of the method, it is used to predict the flow in one typical high expansion ratio rocket nozzle.

2. GOVERNING EQUATIONS

The integral, co-ordinate-free forms of the steady-state conservation equations for mass, momentum and enthalpy, and the equation of state, read[18]

$$\int_S \rho \mathbf{V} \cdot \mathrm{d}\mathbf{S} = 0, \tag{1}$$

$$\int_S (\rho \mathbf{V}\mathbf{V} - \mathbf{T}) \cdot \mathrm{d}\mathbf{S} = \int_\Omega \rho \mathbf{b}\, \mathrm{d}\Omega, \tag{2}$$

$$\int_S (\rho \mathbf{V} h - \mathbf{f}) \cdot \mathrm{d}\mathbf{S} = \int_\Omega [\mathbf{V} \cdot \operatorname{grad} p + (\mathbf{T} + p\mathbf{I}) : \operatorname{grad} \mathbf{V}]\, \mathrm{d}\Omega, \tag{3}$$

$$p = \rho R T, \tag{4}$$

where ρ represents the density, p the static pressure, $\mathbf{I}$ the unit tensor, $\mathbf{V}$ the fluid velocity, h the enthalpy per unit mass, T the absolute temperature, $\mathbf{b}$ the resultant of body forces per unit mass and R the universal gas constant. For a Newtonian fluid, the stress tensor $\mathbf{T}$ can be expressed as follows:

$$\mathbf{T} = -(p + \tfrac{2}{3}\mu \operatorname{div} \mathbf{V})\mathbf{I} + 2\mu \mathbf{D}, \tag{5}$$

where μ is the dynamic viscosity and $\mathbf{D}$ stands for the rate of strain (deformation) tensor, defined as

$$\mathbf{D} = \tfrac{1}{2}[\operatorname{grad}\mathbf{V} + (\operatorname{grad}\mathbf{V})^{\mathrm{T}}]. \tag{6}$$

From the momentum conservation equation (2), equations for the Cartesian velocity components U_i are obtained by taking a dot product with the corresponding base vector $\mathbf{i}_i$:

$$\int_S (\rho \mathbf{V} U_i - \mathbf{t}_i)\cdot \mathrm{d}\mathbf{S} = -\int_S p\mathbf{i}_i \cdot \mathrm{d}\mathbf{S} + \int_\Omega \rho \mathbf{b}\cdot\mathbf{i}_i \mathrm{d}\Omega, \tag{7}$$

where [see equations (5) and (6)]

$$\mathbf{T}\cdot\mathbf{i}_i \Rightarrow \mathbf{t}_i = \mu \operatorname{grad} U_i + \mu(\operatorname{grad}\mathbf{V})^{\mathrm{T}}\cdot\mathbf{i}_i - (\tfrac{2}{3}\mu \operatorname{div}\mathbf{V})\mathbf{i}_i, \tag{8}$$

and the pressure force term is separated from the rest of the stress tensor for reasons which will become clear in the next section.

Following Fourier's law, the heat flux vector $\mathbf{f}$ is defined as

$$\mathbf{f} = k \operatorname{grad} T, \tag{9}$$

where k is the heat conduction coefficient. An ideal gas with constant specific heats C_p and C_v is assumed, their ratio being γ. Under these circumstances the enthalpy may be expressed as $h = C_p T$, so equation (3) becomes an equation for temperature T.

Due to the fact that all the vectors and tensors are expressed through the Cartesian components, there are no curvature terms in the transport equations. This fact has important implications on the numerical solution method, as will be outlined later.

3. NUMERICAL APPROACH

In this section the finite volume discretization of the transport equations is briefly outlined; a more detailed description is available in References 10 and 12. To this end, equations (3) and (7) are seen as special cases of the general transport equation

$$\int_S \underbrace{(\rho \mathbf{V}\phi}_{\text{Convection}} \underbrace{- \Gamma_\phi \operatorname{grad}\phi)}_{\text{Diffusion}} \cdot \mathrm{d}\mathbf{S} = \int_\Omega \underbrace{q_\phi \mathrm{d}\Omega}_{\text{Source}}, \tag{10}$$

where ϕ stands for U_i or T and the diffusion coefficient Γ_ϕ represents μ or k, respectively. The continuity equation (1) has no diffusion and source terms; it will be used to derive an equation for the pressure correction. In other equations, all terms not appearing on the left-hand side of equation (10) are lumped into the source term.

For the sake of simplicity, the discretization will be carried out in a two-dimensional space; the extension to three-dimensional problems is straightforward (cf. Reference 10). The Cartesian co-ordinates and velocity components will, for convenience, be represented by x, y and U, V, respectively.

3.1. Discretization method

The solution domain is subdivided into a finite number of contiguous quadrilateral control volumes (CV). The CVs are defined by co-ordinates of their vertices, which are assumed to be connected by straight lines. This simple form is possible due to the fact that the equations contain no curvature terms and only the projections of the CV faces onto Cartesian co-ordinate surfaces are required in the course of discretization, as will be demonstrated below. All the dependent variables solved for and all fluid properties are stored in the CV center (collocated or cell-centred

arrangement). A suitable spatial distribution of dependent variables is assumed and the conservation equations (1), (7) and (3) are applied to each CV, leading to systems of non-linear algebraic equations. The main steps of the discretization procedure to calculate convection and diffusion fluxes and source terms are outlined below (more details can be found in Reference 10).

All integrals are approximated using the midpoint rule, leading to simple second-order discretization. The mass flux through the cell face e (cf. Figure 1) is evaluated as

$$\dot{m}_e = \int_{S_e} \rho \mathbf{V} \cdot d\mathbf{S} \approx (\rho \mathbf{V})_e \cdot \mathbf{S}_{1e} = \rho_e (U S_1^x + V S_1^y)_e, \tag{11}$$

where $\mathbf{S}_{1e}$ is the surface vector representing the area of the e cell face ($\xi_1 = \text{const.}$) and S_{1e}^x and S_{1e}^y denote its Cartesian components. These are given in terms of the CV vertex co-ordinates as follows:

$$S_{1e}^x = (y_n - y_s)_e \quad \text{and} \quad S_{1e}^y = -(x_n - x_s)_e.$$

The mean cell face velocity components, U_e and V_e, are obtained by interpolating neighbour nodal values in a way which ensures the stability of the collocated grid scheme (cf. Reference 12). The convection flux of any variable ϕ can now be expressed as

$$F_e^C = \int_{S_e} \rho \phi \mathbf{V} \cdot d\mathbf{S} \approx (\rho \phi \mathbf{V})_e \cdot \mathbf{S}_{1e} = \dot{m}_e \phi_e. \tag{12}$$

The value of ϕ_e at the cell face centre, which represents the mean value over the whole face, is in this study expressed through the neighbouring nodal values ϕ_P and ϕ_E by a blend of the second-order central (CDS) and first-order upwind differencing scheme (UDS) using 'deferred correction' approach:[12]

$$\phi_e = \phi_e^{UDS} + \alpha_\phi (\phi_e^{CDS} - \phi_e^{UDS}), \tag{13}$$

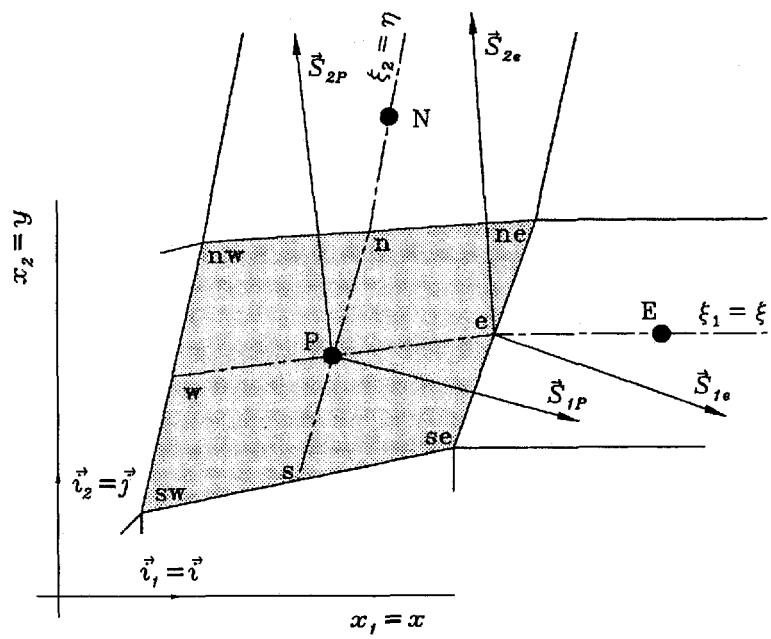


Figure 1. Computational grid and labelling scheme

where the underlined part is treated explicitly and α_ϕ represents a blending factor between 0 and 1 (0 = UDS, 1 = CDS). This factor is set to unity, except when oscillations at discontinuities require some damping and lower values (typically around 0·9) are employed; see Section 4. The same technique can be applied to any combination of higher-order and lower-order schemes. In present calculations a preset value of α_ϕ is used throughout the domain. However, the blending factors may in general be determined locally at each CV face and for each variable by requiring the solution to obey the appropriate bounds.[10]

The diffusion flux of ϕ may be calculated as

$$F_e^D = -\int_{S_e} \Gamma_\phi \operatorname{grad}\phi \cdot d\mathbf{S} \approx -(\Gamma_\phi \operatorname{grad}\phi)_e \cdot \mathbf{S}_{1e}. \tag{14}$$

By expressing the gradient of ϕ at the cell face centre e, which is here taken to represent the mean value over the whole cell face, through the derivaties in ξ_1 and ξ_2 directions (cf. Figure 1) and by discretizing these derivatives with CDS, the following expression results:

$$F_e^D \approx \frac{\Gamma_{\phi,e}}{\mathbf{S}_{1e} \cdot \mathbf{PE}} [(\phi_E - \phi_P)(\mathbf{S}_{1e} \cdot \mathbf{S}_{1e}) + \underline{(\phi_n - \phi_s)_e (\mathbf{S}_{1e} \cdot \mathbf{S}_{2e})}]. \tag{15}$$

PE is the vector representing the distance from P to E, directed towards E. $\mathbf{S}_{2e}$ is the surface vector orthogonal to **PE** and directed towards positive ξ_2 co-ordinate (cf. Figure 1), representing the area in the surface $\xi_2 = 0$ bounded by P and E. Its x and y components are

$$S_{2e}^x = -(y_E - y_P) \quad \text{and} \quad S_{2e}^y = x_E - x_P,$$

and the co-ordinates of the CV centre are equal to quarter of the sum of CV vertex co-ordinates. The underlined part of the diffusion flux (cross-derivative contribution) is treated explicitly and added to the source term. This part vanishes when the grid is orthogonal, since in that case $\mathbf{S}_1 \cdot \mathbf{S}_2 = 0$.

The volumetric source term is integrated by simply multiplying the specific source at the control volume centre P (which is assumed to represent the mean value over the whole cell) with the cell volume, i.e.

$$Q_\phi^q = \int_\Omega q_\phi \, d\Omega \approx (q_\phi)_P \, \Delta\Omega. \tag{16}$$

The pressure terms in the momentum equations are treated as body forces and may be regarded as pressure sources for the Cartesian velocity components. They are evaluated as

$$Q_{u_i}^p = -\int_S p\mathbf{i}_i \cdot d\mathbf{S} = -\int_\Omega (\operatorname{grad} p \cdot \mathbf{i}_i) \, d\Omega \approx -[(p_e - p_w)\mathbf{S}_{1P} + (p_n - p_s)\mathbf{S}_{2P}] \cdot \mathbf{i}_i, \tag{17}$$

where the surface vectors $\mathbf{S}_{1P}$ and $\mathbf{S}_{2P}$ represent the area of the CV cross section at $\xi_1 = 0$ and $\xi_2 = 0$, respectively. Since CVs are bounded by straight lines, these two vectors can be expressed through the CV surface vectors as, e.g., $\mathbf{S}_{1P} = \frac{1}{2}(\mathbf{S}_{1e} + \mathbf{S}_{1w})$ (note that surface vectors are taken positive when pointing in the direction of the corresponding positive ξ-co-ordinate). Terms in the momentum equations not featuring in equation (10) are discretized using the same approach and added to the source term.

After summing up all cell face fluxes and sources, the discretized transport equation reduces to the following algebraic equation:

$$A_P \phi_P + \sum_{nb} A_{nb} \phi_{nb} = Q_\phi, \tag{18}$$

where the coefficients A_{nb} contain the convection and diffusion flux contributions and Q_ϕ represents the source term, which includes all the terms calculated explicitly. For the solution domain as a whole, a matrix equation

$$[A]\{\phi\}=\{Q\}$$

results, where $[A]$ is the square coefficient matrix with non-zero elements only on five diagonals, $\{\phi\}$ is the column matrix of nodal variable values and $\{Q\}$ is the source term matrix. Matrix equations of the same form are obtained for all variables. The equations have been both linearized and decoupled in the course of discretization; these effects will be accounted for through the iteration solution scheme (cf. Section 3.4).

In case of compressible flows, the continuity equation represents a transport equation for density. However, in the present method it is used to obtain a pressure correction equation along the lines of the SIMPLE algorithm[17] for incompressible flows. This procedure is described in the next section.

3.2. *Pressure–velocity–density coupling*

Using the notation shown in Figure 1, equation (18) for the Cartesian velocity components may be written as [cf. equation (17)]

$$U^*_{i,P}=\frac{\sum A_{nb}U^*_{i,nb}+Q^*_{U_i}}{A_P}-\left(\frac{S^i_{1P}}{A_P}\right)(p^*_e-p^*_w)-\left(\frac{S^i_{2P}}{A_P}\right)(p^*_n-p^*_s), \tag{19}$$

where $Q^*_{U_i}$ represents sources without pressure terms. The mass fluxes based on the velocities calculated from equation (19) with existing density and pressure fields do not necessarily satisfy the continuity equation, i.e. there exists a mass imbalance Q^*_m:

$$\dot{m}^*_e-\dot{m}^*_w+\dot{m}^*_n-\dot{m}^*_s=Q^*_m. \tag{20}$$

The mass fluxes need be corrected to satisfy the continuity requirement. Employing the SIMPLE algorithm as a basis for pressure–velocity coupling, its compressible version is developed.

In the case of incompressible flows, the mass flux correction is obtained by correcting the cell face *velocity*. However, in the case of compressible flows, relative density changes become significant,[14–16] and they have to be taken into account when deriving the mass correction expression in terms of *velocity* and *density*. For the sake of simplicity, this will be explained for the mass flux through the e cell face and Cartesian grid ($S^y_{1e}=0$) as follows:

$$\dot{m}^{**}_e=[(\rho^*+\rho')(U^*+U')S^x_1]_e, \tag{21}$$

or, in terms of the mass flux correction,

$$\dot{m}'_e=(\rho^*U'+\rho'U^*+\underline{\rho'U'})_eS^x_{1e}. \tag{22}$$

The underlined term in equation (22) will be neglected hereafter since it vanishes faster than the other two terms.

Velocities at the CV faces are calculated by employing a suitable interpolation formula for the collocated variable arrangement used here (cf. Reference 12). For a Cartesian grid it follows that

$$U^*_e=\overline{\left(\frac{\sum A_{nb}U^*_{nb}+Q^*_U}{A_P}\right)_e}-\overline{\left(\frac{S^x_1}{A_P}\right)_e}(p^*_E-p^*_P), \tag{23}$$

where the overbar denotes interpolation from the neighbour nodal values. The velocity correction can be expressed in terms of the pressure corrections by using a truncated form of equation (23)

$$U'_e = -\overline{\left(\frac{S_1^x}{A_P}\right)_e}(p'_E - p'_P). \tag{24}$$

Density correction can be implicitly expressed through the pressure correction in the following manner:

$$\rho' = \left(\frac{\partial \rho}{\partial p}\right) p'. \tag{25}$$

The term $\partial\rho/\partial p$ can, for an isentropic flow, be written as follows:

$$\left(\frac{\partial \rho}{\partial p}\right) = \frac{1}{c^2} = \frac{1}{\gamma RT} = C_\rho, \tag{26}$$

where c represents the speed of sound. There are other ways to derive an expression for the term $\partial\rho/\partial p$, depending upon the kind of flow. However, the converged solution is independent of the manner in which the dependency of the velocity and density corrections on pressure correction is defined—only the rate of convergence is affected. Introducing equations (24) and (25) into equation (22), it follows that

$$\dot{m}'_e = -\rho_e^* S_{1e}^x \overline{\left(\frac{S_1^x}{A_P}\right)_e}(p'_E - p'_P) + U_e^* S_{1e}^x (C_\rho p')_e. \tag{27}$$

Some important conclusions can be derived from equation (27).

Firstly, it contains not only a term involving the pressure correction difference across the cell face (diffusion-like term), but also a term depending on p' at the cell face centre (convection-like term). This means that a function $p' + C$, where C is an arbitrary constant, will not satisfy equation (27). In the case of an incompressible flow, the mass flux correction is a function of the velocity correction only, which, on the other hand, is a function of the pressure correction gradient; this is where the diffusion-like term comes from. In the case of a compressible flow, the mass flux correction is a function of both velocity *and* density corrections, and from the latter the convection-like term results.

Secondly, the relative importance of the two parts of the mass flux correction depends on the kind of flow. If the right-hand side of equation (27) were normalized with $(U^* S_1^x C_\rho)_e$ to yield a weighting factor of 1 for p', then the first (diffusion-like) term will have a weighting factor proportional to $(c/U)^2 = M^{-2}$. Obviously, for a subsonic flow this factor is several orders of magnitude larger than 1, so the first term dominates and the second is negligible. For compressible flows with high Mach numbers, the contrary is true: the weighting factor of the first term will be an order of magnitude or more below 1, so the second (convection-like) term will dominate. As in the transport equations, the CDS is blended with the UDS when discretizing the convection-like term, i.e. the terms $(C_\rho p')_e$ and ρ_e^* are calculated in the following way:

$$\rho'_e = (C_\rho p')_e = (C_\rho p')_e^{\mathrm{UDS}} + \alpha_\rho [(C_\rho p')_e^{\mathrm{CDS}} - (C_\rho p')_e^{\mathrm{UDS}}], \tag{28}$$

$$\rho_e^* = \rho_e^{*\mathrm{UDS}} + \alpha_\rho (\rho_e^{*\mathrm{CDS}} - \rho_e^{*\mathrm{UDS}}). \tag{29}$$

By introducing the above expressions into equation (27), and requiring that the mass conservation is satisfied:

$$\dot{m}'_e - \dot{m}'_w + \dot{m}'_n - \dot{m}'_s + Q_m^* = 0, \tag{30}$$

an algebraic pressure correction equation of the following form results:

$$A_P p'_P + \sum_{nb} A_{nb} p'_{nb} = -Q^*_m. \tag{31}$$

Here A_{nb} has a similar form as in equation (18) and includes contributions from both convection-like and diffusion-like terms. However, A_P is not equal to the sum of the neighbour coefficients, as is the case in the momentum and scalar conservation equations. This would be an undesirable feature in these equations, since it would mean that, all the neighbour values being equal, the value at node P would turn out to be different. An exception would be when all the neighbour values are equal to zero and the source term at node P is also equal to zero. Fortunately, the pressure correction equation fulfils this condition at the converged state.

3.3. Boundary conditions

Several different types of boundaries may be encountered in flow calculations, such as inflow, outflow, impermeable wall and symmetry. In the case of viscous incompressible flows, the following boundary conditions usually apply:

(a) the velocities and temperature are prescribed at the inlet;
(b) zero normal gradient for the parallel velocity component and for all scalar quantities, and zero normal velocity component are specified at symmetry planes or axes;
(c) no-slip condition and prescribed temperature or heat flux are specified at the walls;
(d) zero (or constant non-zero) gradient of all variables is specified at the outlet.

In the case of compressible flows, some new boundary conditions may apply:

(i) prescribed total conditions (pressure, temperature) and flow direction at inflow;
(ii) prescribed static pressure* at outflow;
(iii) supersonic outflow.

Most of the above boundary conditions are either straightforward to implement or their implementation is described in earlier publications.[10, 12] In this section the last three [(i)–(iii)] boundary conditions and their influence on the pressure correction equation will be discused.

Prescribed total conditions at inlet. The implementation of this boundary condition is demonstrated at the west boundary (cf. Figure 2).

The total pressure is defined as[19]

$$p_t = p\left(1 + \frac{\gamma - 1}{2}\frac{U^2 + V^2}{\gamma RT}\right)^{\gamma/(\gamma-1)} \tag{32}$$

and the flow angle as

$$\tan\beta = \frac{V}{U} \quad \text{or} \quad V = U\tan\beta. \tag{33}$$

One possible way of implementing the prescribed total pressure condition at inlet would be to extrapolate the pressure from the interior to the boundary and calculate the velocity from (32),

* For incompressible flows, the pressure may be specified at inflow and outflow boundaries, instead of specifying velocities or velocity gradients.

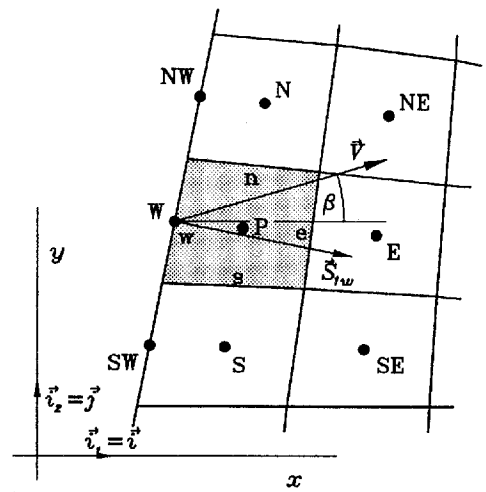


Figure 2. On the implementation of the prescribed total-pressure boundary condition for CVs along the west boundary

treating the velocity thus obtained as specified for one SIMPLE iteration. The temperature is assumed to be either directly specified or calculated from the prescribed total temperature:

$$T_t = T\left(1+\frac{\gamma-1}{2}\frac{U^2+V^2}{\gamma RT}\right). \tag{34}$$

However, this approach leads to poor convergence of the solution process. Better convergence properties are achieved by using the following practice, which incorporates implicitly the influence of pressure on the velocity.[14]

At the beginning of one outer iteration (see Section 3.4), the velocities at the inlet (west) boundary, U_w and V_w, are calculated from equations (32) and (33) using the available pressure and temperature values. These values are used in the momentum equations as specified boundary velocities, i.e. treated as known for the current iteration. However, the mass fluxes at the inlet boundary, which are used to approximate the convection fluxes, are not based on this velocity but are calculated such that the mass conservation is assured, as will be explained below. Upon solving the momentum equations for U^* and V^*, the *new* mass fluxes are calculated throughout the solution domain, using the newly calculated velocity components. The new inlet mass fluxes are now calculated using the velocities U_w and V_w defined above. In the SIMPLE procedure, the inlet mass fluxes are also to be corrected; the correction is invoked by correcting the velocity alone, assuming that the density is either prescribed or treated as such within one outer iteration. The velocity corrections are expressed via pressure correction as follows:

$$U'_w = \left(\frac{\partial U^*}{\partial p}\right)_w p', \quad V'_w = U'_w \tan\beta. \tag{35}$$

The coefficient $(\partial U^*/\partial p)_w$ is obtained from equation (32):

$$\left(\frac{\partial U^*}{\partial p}\right)_w = -\frac{\gamma RT^*}{U^*\gamma(1+\tan^2\beta)p_t\left[1+\dfrac{\gamma-1}{2}\dfrac{U^{*2}(1+\tan^2\beta)}{\gamma RT^*}\right]^{(1-2\gamma)/(\gamma-1)}}. \tag{36}$$

The boundary mass flux correction is expressed as a function of the boundary pressure correction as follows:

$$\dot{m}'_w = \left[\rho^* \left(\frac{\partial U^*}{\partial p} \right) (S_1^x + S_1^y \tan \beta) \right]_w \overline{p'_w}, \tag{37}$$

where $\overline{p'_w}$ is obtained by extrapolating the pressure correction from interior nodes, i.e. it is expressed as a linear function of p'_P and p'_E. Thus, the coefficients for the P and E nodes in the pressure correction equation are corrected along the inlet boundary according to the above expression.

Upon solving the pressure correction equation, the velocities, the pressure *and* the mass fluxes are corrected throughout the solution domain, including the inlet boundary. The corrected mass fluxes are used to calculate the convection fluxes in all equations in the following outer iteration. The velocities at the inlet boundary are, however, now calculated from equations (32) and (33) to serve as prescribed boundary velocities in the next iteration. At convergence, the mass flux correction will be zero so that the velocity thus calculated will also define the mass flux through the inlet boundary.

The above implementation of the total conditions at the inlet was used in all calculations of nozzle flows presented in the next section. It proved to be stable and of good convergence properties.

Prescribed static pressure. The east boundary is chosen here to demonstrate the implementation of this boundary condition (cf. Figure 3). It is assumed that the velocities—and in the case of an outflow boundary, other variables as well—are extrapolated to the boundary from the interior region. The pressure correction equation needs to be modified for the cells next to the boundary. The pressure correction at the boundary by definition will be equal to zero, but the velocity and mass flux corrections will be non-zero until the converged solution is obtained.

The velocity components at the boundary nodes are calculated in a similar way as in equation (23), namely,

$$\overline{U^*_{i,e}} = \overline{\left(\frac{\sum A_{nb} U^*_{i,nb} + Q^*_{U_i}}{A_P} \right)_e} - 2 \overline{\left(\frac{S_1^i}{A_P} \right)_e} (p^*_E - p^*_P) - \overline{\left(\frac{S_2^i}{A_P} \right)_e} (p^*_n - p^*_s)_e, \tag{38}$$

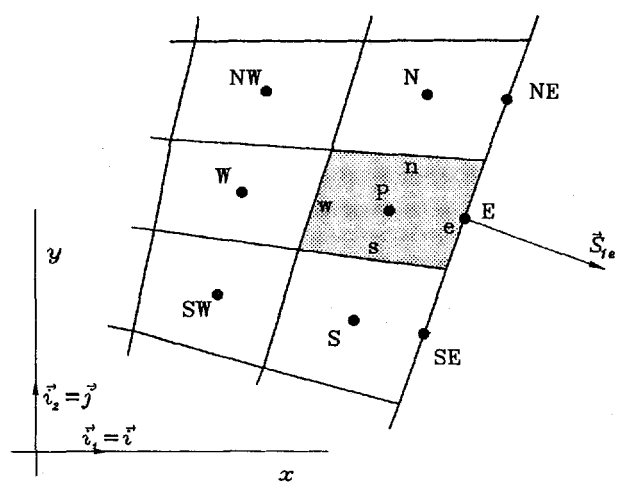


Figure 3. On the implementation of the prescribed static-pressure boundary condition for CVs along the east boundary

which results in the following expressions for the velocity corrections:

$$\overline{U'_{i,e}} = -2\overline{\left(\frac{S_1^i}{A_P}\right)_e}(p'_E - p'_P). \tag{39}$$

The factor 2 results from considering a CV centred around boundary location e (which is identical to E) and assuming that the pressure difference across it is twice the difference $p_E - p_P$. The mass flux correction is calculated as [cf. equations (11), (21) and (22)]

$$\dot{m}'_e = [\rho^*(\overline{U'}S_1^x + \overline{V'}S_1^y) + \rho'(\overline{U^*}S_1^x + \overline{V^*}S_1^y)]_e, \tag{40}$$

where the overbar denotes values which are obtained by extrapolation. The density and its correction are treated according to equations (28) and (29), where the CDS value is the one obtained for the boundary node E by extrapolation, and the UDS value is that of the cell centre P. The resulting modifications for the coefficients of nodes P, E and W in the pressure correction equation are easily derived from the above expressions.

While in the case of a compressible flow the condition of prescribed static pressure can be applied at exit boundaries only, in the case of incompressible flows it can be applied at both inlet and outlet boundaries. The corresponding expressions for the velocities and mass fluxes at boundaries can be derived from those given above by setting the density correction to zero. If the static pressure is prescribed at both inlet and outlet in an incompressible flow, the velocities cannot be prescribed at any of these boundaries as the mass flow rate is directly proportional to the pressure difference and cannot be separtely enforced. At low Reynolds numbers the above procedure leads to rapid convergence. As the Reynolds number increases, the rate of convergence becomes lower. Obviously, static pressures at inlet and outlet cannot be prescribed for an inviscid flow.

It should also be noted that this boundary condition allows both inflow and outflow to be present at the same boundary, as the flow direction is not prescribed. In case of a severely non-orthogonal grid at the boundary, it may be necessary to incorporate the effect of the pressure gradient along the boundary [cf. equation (38)] in the mass flux corrections.[20]

Supersonic outlet. If supersonic conditions prevail at the exit boundary, all variables there must be determined by extrapolation from interior of the solution domain.[21] The velocity and mass flux, as well as their corrections, are obtained in the same way as in the case of a specified pressure at the outlet [cf. equations (38)–(40)]. However, since the pressure is not specified but is also extrapolated, the pressure correction at the boundary node, p'_E, is not zero as in the former case. It is expressed here as a linear function of p'_P and p'_W, and the coefficients of the pressure correction equation have to be modified accordingly for the next-to-boundary cells. The implementation of this boundary condition, as well as those applied at the walls and symmetry planes or lines, is straightforward and will not be discussed further.

3.4. Solution method and its properties

The flow governing equations are solved iteratively using the segregated approach. For each transport equation, a system of linear algebraic equations is obtained. These are solved in turn by applying the Stone's[22] solver based on ILU decomposition. Keeping the coefficients in the algebraic equations fixed, one to ten iterations are performed (*inner iterations*). Typically, only one iteration is performed for the velocities and temperature, and in the pressure correction equation, iterations are stopped when the sum of the absolute residuals over the whole solution

domain has fallen about one order of magnitude, or the prescribed maximum number of inner iterations has been reached. The equation of state is used to update the density after the new solutions for temperature and pressure are obtained.

After one cycle of inner iterations has been performed for each variable, the coefficients of the algebraic equations are updated using the newest values of all variables (*outer iterations*). In this way the non-linearity and coupling of equations is accounted for. Convergence of the solution process is checked each time the new coefficients have been assembled, by calculating the residuals using the new coefficients and the old variable values. The iterations are stopped when the sum of the absolute residuals for each variable has fallen a prescribed number of magnitudes or when the normalized sums are smaller than a prescribed small number. In addition, variable values at a point located in a sensitive flow region are monitored; they should not be changing at the prescribed number of leading figures from iteration to iteration when the solution process is stopped.

It should be noted here that the pressure correction equation requires less inner iterations (1–3) in the case of a compressible flow than for incompressible flows (4–10). In the latter case, one is solving basically a Poisson equation (symmetric coefficient matrix) with Neumann boundary conditions on all boundaries, which converges slowly. In the former case, the convection-like contribution makes the coefficient matrix in the pressure correction equation highly asymmetric, and the boundary conditions are of the Dirichlet type where pressure is specified, which results in faster convergence.

The advantages of the present solution method over schemes which employ staggered grids and/or grid-oriented velocity components as dependent variables lie in its simplicity and robustness. The discretization is of second order and the method is not very sensitive to grid smoothness, discontinuities on boundaries and the Reynolds number (for the same geometry and computational grid, the rate of convergence does not differ much over the range from low Reynolds number incompressible flows to supersonic flows). The fact that only one set of CVs is used makes adaptation of grid to boundaries easy, as well as the implementation of multigrid methods, block-structured grids and parallel computing. The method is easily extended to three-dimensional and unsteady flows as well as to unstructured grids of arbitrary CV shape.

The simplicity of the method is largely lost if a consistent discretization of an order higher than second is sought, since the approximation of CV surface fluxes then becomes more complicated. One of the drawbacks is that the pressure at boundaries is always required in the integration process, as CVs of all velocity components are always boundary-fitted. Simple extrapolation of pressure from the interior has proven efficient and accurate enough in all cases studied so far.

It should also be noted that in the present cell-centred scheme all interpolations are carried out in physical space (no co-ordinate transformation is involved). The CDS approximation of convection fluxes is then of second order on both uniform and non-uniform grids, unlike in some other cell-centred schemes which interpolate on uniform, transformed grids, and therefore become less accurate when the grid is non-uniform.[23] All grids used in the calculations reported in the next section were non-uniform (finer near walls and in areas of expected large variation of gradients).

4. APPLICATION OF THE METHOD

Applications of the method to incompressible flows are documented in other publications.[10,12,24] Attention will therefore be focused here on compressible flows. Several representative test cases for inviscid and viscous flows are presented below.

4.1. Inviscid test cases

One-dimensional flows. First calculations with the present solution method were performed for some standard one-dimensional (1D) compressible flow problems which are often found in the literature. In all cases, solutions of comparable accuracy to the reference literature solutions were obtained. Particular attention was paid to the prediction of compressible flow through a 1D nozzle. As is well known, there exists a 'forbidden' range of inlet Mach numbers which cannot be realized in practice, due to the fact that the nozzle will choke as long as the sonic line is present in the throat. As Bošnjaković[25] described it, should someone be trying to pull a nozzle through the air at a velocity which would correspond to an inlet Mach number within this range, a shock will appear immediately before the nozzle and reduce the inlet viscosity so that the limiting case is reached.

Here calculations were peformed for a converging–diverging 1D nozzle whose cross-sectional area varies as

$$S_{(x)} = S_{\text{th}} + (S_{\text{i}} - S_{\text{th}})\left(1 - \frac{x}{5}\right)^2$$

where $S_{\text{i}} = 2{\cdot}035$ and $S_{\text{th}} = 1$ are the inlet and throat areass, respectively, and $0 \leq x \leq 10$. A wide range of inlet Mach numbers was used, resulting in the pressure and Mach number distributions shown in Figure 4. The lower critical line is reached at an inlet Mach number of 0·3. For all inlet conditions leading to Mach numbers up to 2·2167, the same solution was obtained: a sudden drop of Mach number resulted immediately after the inlet. Specifying all variable values at the inlet so as to impose an 'invalid' Mach number implies to only setting a total pressure level: the solution obtained will be the one corresponding to that level and following the lower limit on Mach number. Only for inlet Mach numbers above 2·2167 is supersonic flow obtained in the whole

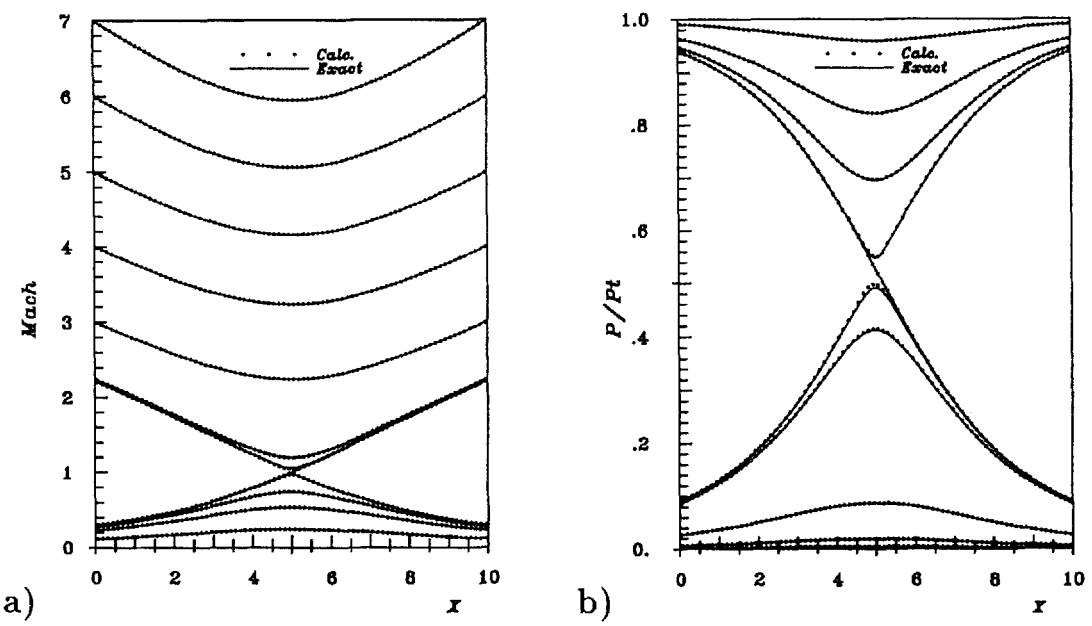


Figure 4. Comparison of distributions of Mach number (a) and pressure (b) for one-dimensional inviscid nozzle flow: solid line—analytic solution, symbols—calculation

nozzle, giving the pressure distributions shown in the lower portion of Figure 4(b). The solution method is thus behaving in a 'natural' way and not allowing unphysical solutions to be obtained.

Flows over circular arc bump. Three different types of flow (subsonic, transonic and supersonic) in a channel with a circular arc bump were chosen as further validation tests for inviscid flow calculations. The width of the channel is equal to the length of the bump, and the channel length is equal to three lengths of the bump. In all three cases, the grid is symmetrical left and right from the middle of the bump. For subsonic and transonic calculations, the thickness-to-chord ratio is 10% and for supersonic flow calculations it is 4%. These are the standard test cases proposed at the GAMM conference[26] and used by many investigators[4, 14, 15, 27, 28] to check the accuracy and stability of the numerical method used.

Due to the fact that the enthalpy equation for inviscid flows can simply be expressed as

$$h_t = C_p T + \frac{|\mathbf{V}|^2}{2}, \tag{41}$$

where h_t represents total enthalpy per unit mass, the enthalpy conservation equation (3) is not solved and the temperature is calculated directly from equation (41).

A *subsonic* flow regime is considered first. At the inlet, it is assumed that flow has uniform properties and the upstream far-field variable values (except for pressure) are specified. At the outlet, all variables are extrapolated except for pressure, which is prescribed. At the upper and

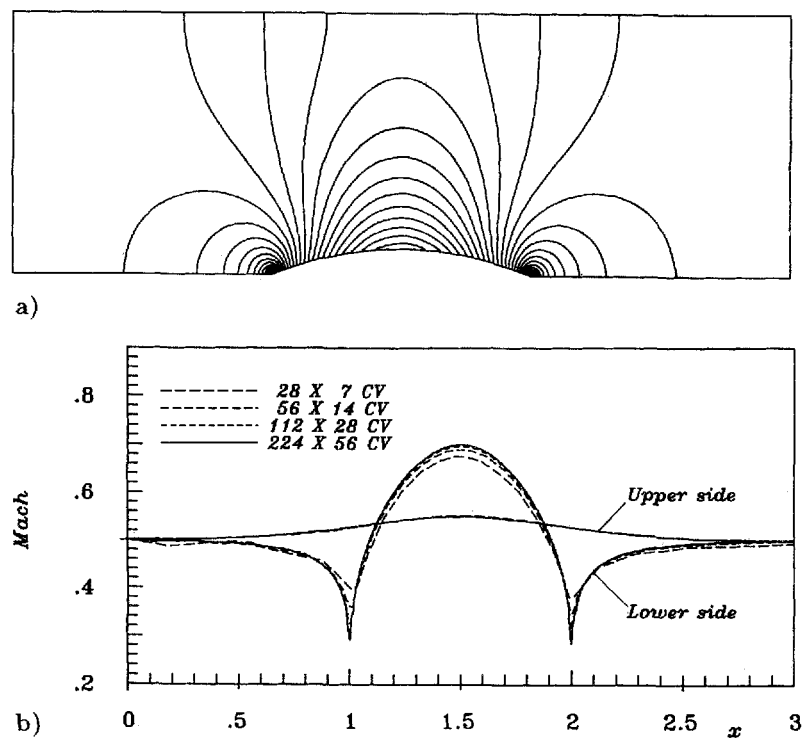


Figure 5. Predicted Mach number contours (a) and profiles along the walls (b) for subsonic inviscid flow through a channel with a circular arc bump in one wall, using a grid with 224 × 56 CV and 95% CDS. Contour levels: from 0·295 (at bump corners) to 0·685 (at bump top); step 0·015

lower wall, the flow tangency condition and zero mass flux through the boundary are prescribed. For a given Mach number at the inlet, $M_{in}=0{\cdot}5$, grid dependence tests were performed for four systematically refined non-uniform grids, from 28×7 to 224×56 CV. For all grids the CDS was blended with 5% UDS. In Figure 5(b), the Mach number distribution along the upper and lower wall calculated on all grids is presented. As there are no shocks in this case, a symmetrical solution must result. Even in the case of the coarsest grid with 28×7 CV, the total pressure error is relatively small. On the finest grid the error is below 0·25%. These results compare favourably with those found in many other publications, e.g. in References 14 and 29. In Figure 5(a), the isomach lines are shown as predicted on the finest grid. The fact that the grid lines suddenly change their direction at the beginning and end of the bump causes no disturbance to the isobars as is the case in solution methods employing grid-oriented velocity components.[14,15]

Next, *transonic* flow was considered. The grid arrangement and the treatment of boundary conditions are identical to those described for subsonic flow. For the given inlet Mach number $M_{in}=0{\cdot}675$, the Mach number distributions along the walls and isomach lines as predicted on the finest grid are shown in Figure 6. For all grids the CDS was blended with 10% UDS. The captured shock spreads over three grid points on all grids. The maximum Mach number immediately before the shock is 1·43. These results also correspond to reference solutions from the literature.[4,28] As Figure 6(b) suggests, the discretization error on the finest grid (224×56 CV) is fairly small; further grid refinement would only affect the resolution of shock and peaks at bump ends.

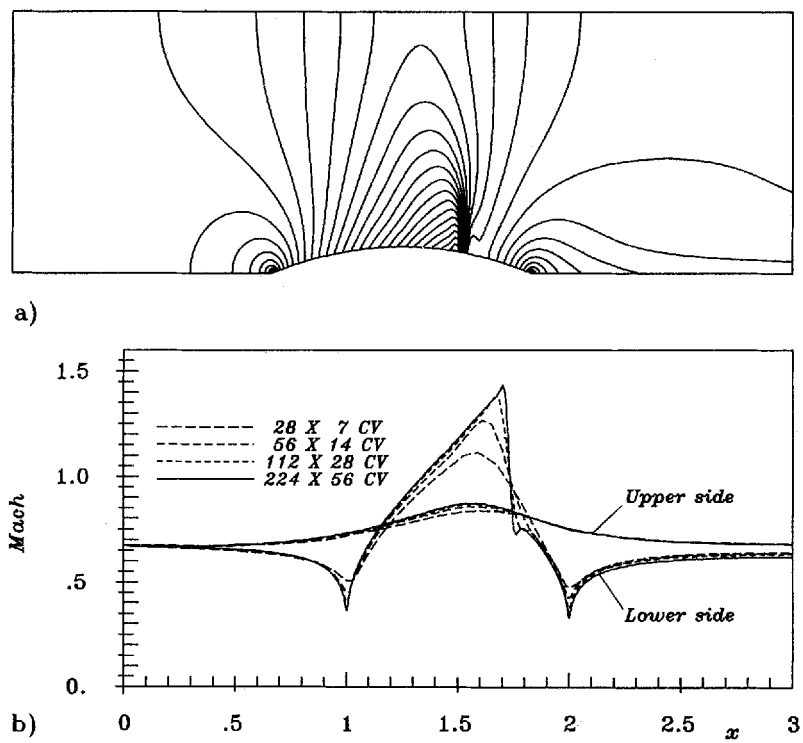


Figure 6. Predicted Mach number contours (a) and profiles along the walls (b) for transonic inviscid flow through a channel with a circular arc bump in one wall, using a grid with 224×56 CV and 90% CDS. Contour levels: from 0·34 (at bump corners) to 1·42 (at shock); step 0·04

Finally, *supersonic* flow was analysed. For the given uniform Mach number at the inlet, $M_{in} = 1{\cdot}65$, and for the used geometry, the flow is also supersonic at the outlet. For that reason all variables are prescribed at inlet, and at outlet all variables are extrapolated. In Figure 7 the Mach number distribution and isomach lines are shown, as predicted on a non-uniform 160×80 CV grid using a blend of 90% CDS and 10% UDS. Two oblique shocks are formed at both corners of the bump. The leading edge shock reflects from the top wall and intersects with the shock leaving the trailing edge. All shocks are resolved fairly well, although at this stage no effort was made to increase the shock resolution and the reflecting one is somewhat smeared. It is interesting to note that the shock position does not change with grid refinement [cf. Figure 7(b)]; only the steepness of the shock is improved, with the lines calculated on various grids intersecting at almost the same point. Figure 8 shows isomach lines for the flow under the same conditions, but with two bumps. As can be seen by comparing Figures 7 and 8, the second bump does not influence the flow upstream of it. This indicates that the solution method correctly reproduces the hyperbolic nature of the flow.

With 10% UDS blended with CDS, there are still some oscillations in the solution at locations where a sudden change in gradients occurs. This is a feature of all higher-order schemes. The oscillations are usually centred around the accurate solution and are reducing with grid refinement in both wavelength and amplitude. Figure 9 shows isomach lines for the above flow obtained using pure CDS for all convection terms. It indicates very good resolution of shocks, but

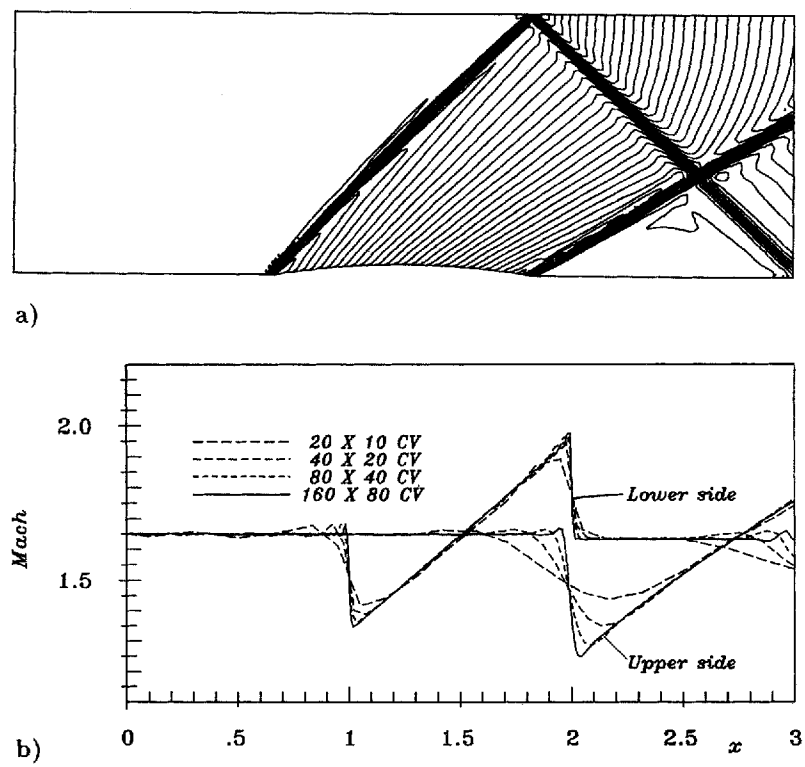


Figure 7. Predicted Mach number contours (a) and profiles along the walls (b) for supersonic inviscid flow through a channel with a circular arc bump in one wall, using a grid with 160×80 CV and 90% CDS. Contour levels: from 1·26 (after shock, upper wall) to 2·02 (before shock, lower wall); step 0·02

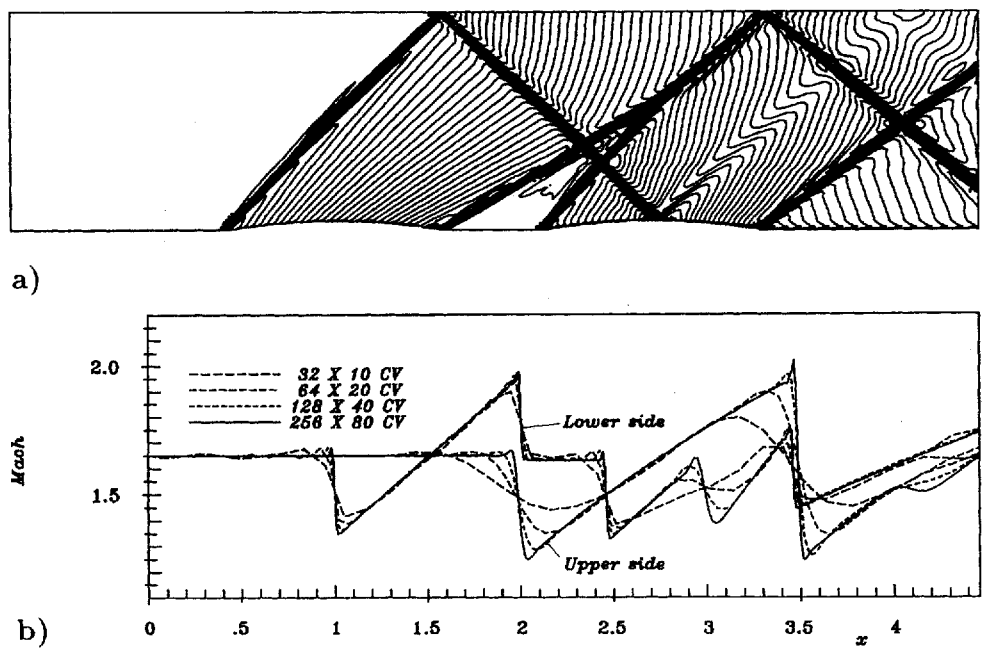


Figure 8. Predicted Mach number contours (a) and profiles along the walls (b) for supersonic inviscid flow through a channel with two circular arc bumps in one wall. Conditions and contour levels as in Figure 7

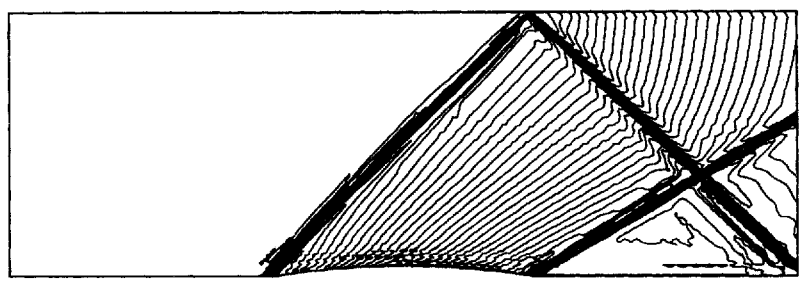


Figure 9. Predicted Mach number contours for supersonic inviscid flow through a channel with a circular arc bump in one wall, using a grid with 160×80 CV and pure CDS. Conditions and contour levels as in Figure 7

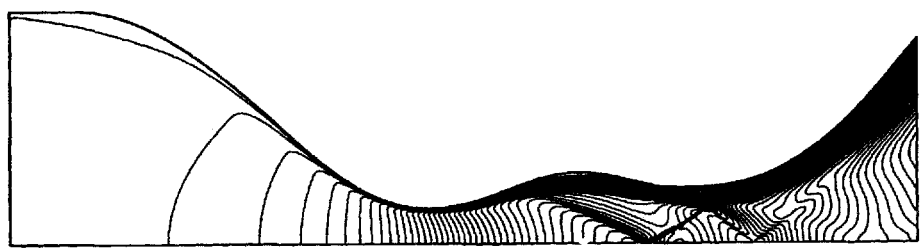
also pronounced oscillations. In order to avoid oscillations while retaining good resolution of shocks, blending of UDS with CDS should be done locally, as was done by Perić[10] for some simple scalar transport problems, instead of the global blending used here. This and other possibilities for improvement of shock resolution will be considered in future studies.

For the wall inclination angle of 9·14° at the beginning of the bump, a shock appears for inlet Mach numbers greater than approximately 1·37 (cf. Reference 19). When inlet Mach numbers are below this critical value, no shock is formed and the bump causes the flow to behave as in a nozzle (since the upper wall in an inviscid case can be replaced by a symmetry plane of a 2D nozzle). This means that bump will choke the flow, as discussed for the 1D example nozzle. Specifying inlet conditions such that Mach numbers lie in the 'forbidden' range causes a shock-like drop of Mach

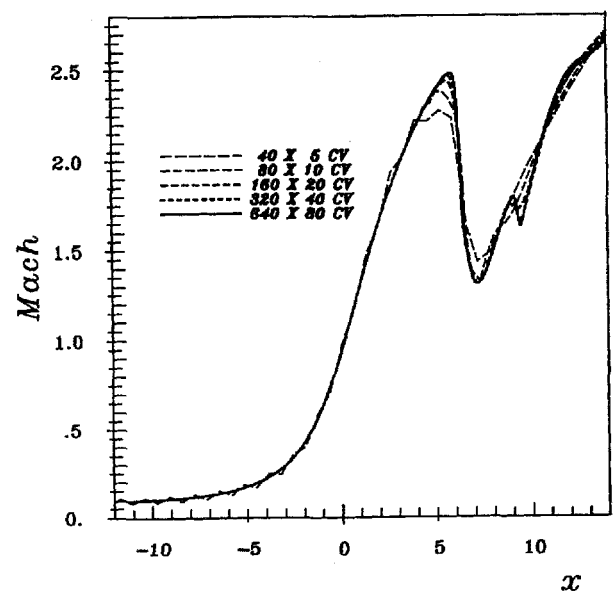
number immediately after the inlet, and the flow follows the lower critical line (for this geometry it occurs at an inlet Mach number of 0·8).

4.2. *Viscous test cases*

Flow in a double-throat nozzle. Flow in a double-throat nozzle was used as a validation test case for viscous flow calculations. Flow conditions and geometry are well described by Bristeau *et al.*[30] At the inlet, total pressure and total enthalpy are prescribed. At the outlet, all variables are extrapolated. At the wall, the temperature is set equal to the total temperature. For this test case the full CDS scheme is used for all variables. Figure 10 presents Mach number contours for the flow at Reynolds number $Re = 1600$, as predicted on a grid with 640×80 CV (a) and centreline profile of Mach number as a function of grid fineness (b). Solutions on the two finest grids differ appreciably only near the exit, where the effect of better resolution of the shock reflection can be observed. The position of shocks and the peak values of Mach number of the centreline are in good agreement with reference solutions presented in Reference 30.



a)



b)

Figure 10. Predicted Mach number contours for a supersonic laminar flow at $Re = 1600$ in a double-throat nozzle using a grid with 640×80 CV and pure CDS (a) and centreline profile of Mach number as a function of grid fineness (b). Contour levels: from 0·06 (inlet) to 2·88 (outlet); step 0·06

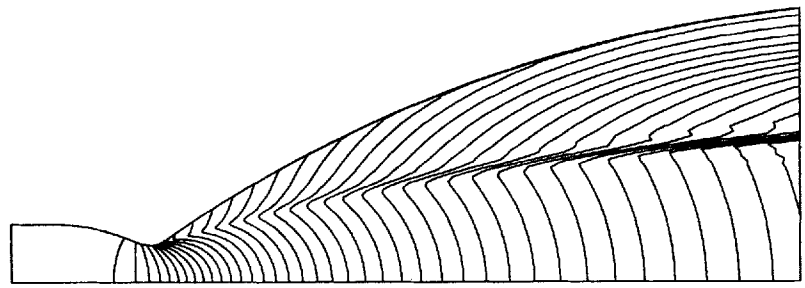
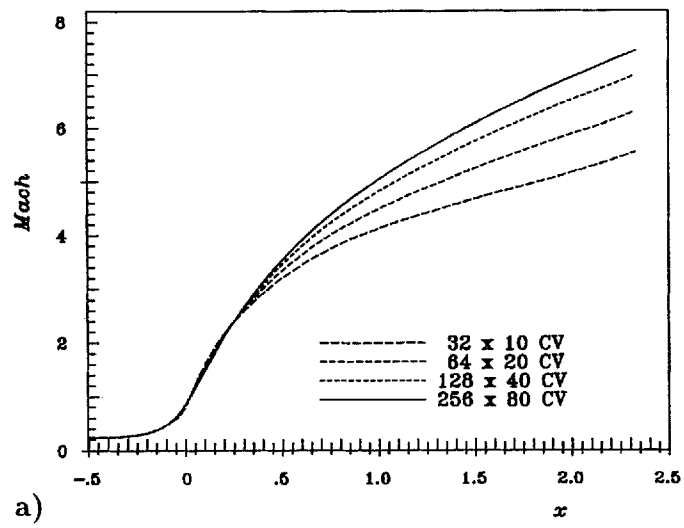


Figure 11. Predicted Mach number contours for supersonic laminar flow in high-expansion rocket nozzle using a grid with 256 × 80 CV and 95% CDS. Contour levels: from 0·4 (inlet) to 8·0 (outlet); step 0·2



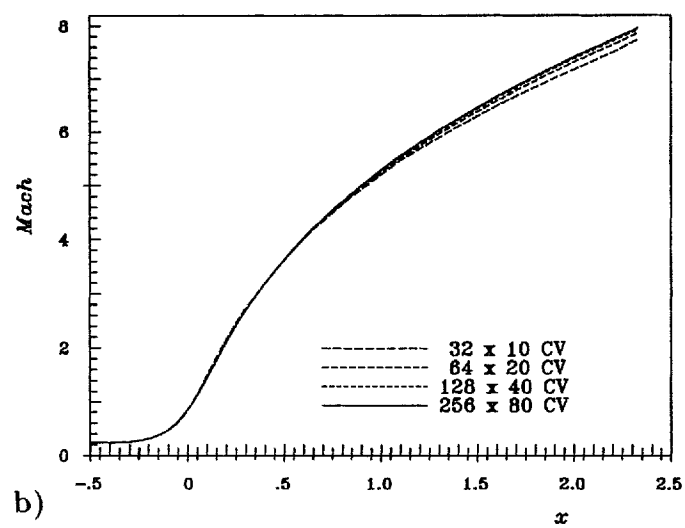


Figure 12. Mach number profiles along centreline for supersonic flow in a high-expansion rocket nozzle as a function of a grid fineness using pure UDS (a) and 95% CDS (b)

Flow in a high-expansion rocket nozzle. As a further illustrative example, the prediction of the air flow in a high-expansion-ratio rocket nozzle has been performed. At the inlet, the total pressure $p_t = 104$ bar, the total temperature $T_t = 3600$ K and flow direction ($\beta = 0$) are prescribed. At the outlet, all variables are extrapolated. In this example the CDS was blended with 5% UDS for convection fluxes on the finest grid (256 × 80 CV) for all variables. The flow is characterized by very high Mach numbers without shocks. The inlet pressure is 100 bar and at the outlet it is 0·0108 bar. The Mach number varies from 0·239 at the inlet to over 8 at the outlet. In Figure 11, the calculated isomachs are presented. A compression wave can be seen downstream of the throat: below it, the Mach number is almost constant in the cross section, and rises steadily in the axial direction; above it, the Mach number suddenly drops. This is a typical feature of nozzle flows and can be seen in many similar predictions.[31,32]

Figure 12 shows the Mach number profile along the centreline as predicted on various grids using a pure UDS (a) and using 95% CDS and 5% UDS (b). This result indicates the inadequacy of the first-order upwind scheme for the prediction of complex flows: the solution on the finest grid with pure UDS is less accurate than the solution on the coarsest grid when 95% CDS is used! The discretization error on the finest grid is around 7% for pure UDS and only around 0·3% for the CDS with 5% UDS damping.

Figure 13 shows the convergence properties of the solution method on a 256 × 80 CV grid, using 95% CDS and 5% UDS. The calculation was started using initial variable values obtained by extrapolation from the solution on the next coarser grid. The normalized sums of absolute residuals for the momentum, temperature and continuity equations are shown as a function of the

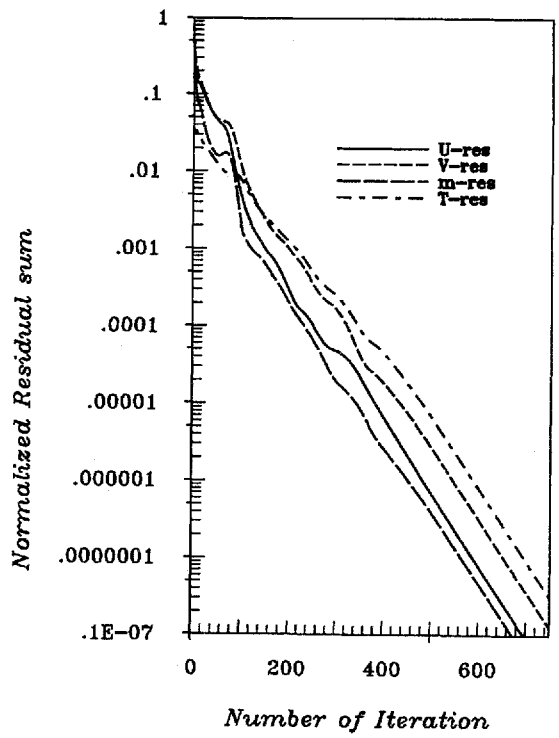


Figure 13. Convergence of the solution process for the supersonic laminar flow through a high-expansion rocket nozzle on a grid with 256 × 80 CV

number of iterations performed. All residuals are falling steadily at a constant rate. Each order of magnitude reduction corresponds approximately to an accuracy improvement of one significant figure in variable values. For practical applications, four-digit accuracy is sufficient, meaning that about 300 iterations would have been necessary in the above case. The rate of convergence is influenced strongly by the oscillations of the solution at the edge of the compression wave. By using 60% CDS and 40% UDS, the oscillations are dumped completely and the number of required iterations reduces by almost a factor of three.

5. CONCLUSIONS

The analysis presented in the previous sections allows the following conclusions to be drawn:

(a) The present method is based on the SIMPLE like pressure correction algorithm and can be used to calculate flows at all speeds. It is self-adaptive to the type of flow and in case of incompressible flows it reduces to its standard form, which is—in several variants—being used widely by many authors. Since the collocated variable arrangement and Cartesian velocity components are used, there are no curvature terms in the momentum equations and the solutions are not as sensitive to grid smoothness as they are when grid-oriented velocity components are used.

(b) In an extensive testing of the method it was found to be stable and have good convergence properties. In incompressible flows, zero fields are used usually to initialize the variables, as the rate of convergence and the total computing time are usually not very dependent on the choice of these values. For compressible flows, it is desirable to have a more reasonable guess of the pressure and velocity distributions. It was found advantageous to use a sequence of systematically refined grids even without the multigrid algorithm, for the possible difficulties with initial conditions are then limited to the coarsest grid only. Since the computing time on the coarsest grid is very short, one can afford to use heavier under relaxation and other stabilizing techniques. The solution on each grid is used to obtain (by extrapolation) the starting fields on the next finer grid, thus providing a very good initial solution. In addition, a series of solutions on systematically refined grids is useful for estimating the level of discretization errors, since these are proportional to the difference between the solutions on two consecutive grids.[33]

(c) The present method is found to capture shocks automatically at the right locations on all grids. The grid refinement improves the resolution of shocks, but their position was not changing during this process in all calculations performed so far.

(d) The use of the central differencing scheme for convection fluxes, blended with a small proportion of the first-order upwind scheme, was found to result in a small total pressure loss and acceptable resolution of shocks. Both features can be improved further by calculating the blending factor locally (e.g. in a manner described by Perić[10]).

The results presented here were all limited to two-dimensional flow problems. The solution method, however, can easily be implemented in any three-dimensional solution procedure using algorithms of the same kind. Computational efficiency can be improved further by implementing a multigrid solution algorithm of the kind presented in Reference 33.

ACKNOWLEDGEMENTS

The authors would like to thank Dr. R. I. Issa for the helpful discussions on the subject of this paper. The financial support from the Deutsche Forschungsgemeinschaft is greatly appreciated.

REFERENCES

1. A. Jameson, W. Schmidt and E. Turkel, 'Numerical solutions to the Euler equations by finite volume methods using Runge–Kutta time stepping', *AIAA Paper 81–1259*, 1981.
2. R. M. Beam and R. F. Warming, 'An implicit factored scheme for the compressible Navier–Stokes equations', *AIAA J.*, **16,** 393–402 (1978).
3. R. W. MacCormack and A. J. Paullay, 'Computational efficiency achieved by time splitting of finite difference operators', *AIAA Paper 72–154*, 1972.
4. R. H. Ni, 'A multiple grid scheme for solving the Euler equations', *AIAA J.*, **28,** 1565–1571 (1982).
5. A. Rizzi and L. E. Eriksson, 'Transfinite mesh generation and damped Euler equation algorithm for transonic flow around wing-body configuration', *AIAA Paper 81–0999*, 1981.
6. J. Sahu, C. J. Nietubitcz and J. L. Steger, 'Navier–Stokes computations of projectile base flow with and without base injection', *AIAA J.*, **23,** 1348–1355 (1985).
7. W. Soh and J. Goodrich, 'Unsteady solution of incompressible Navier–Stokes equations', *J. Comput. Phys.* **79,** 113–134 (1988).
8. S. V. Patankar, *Numerical Heat Transfer and Fluid Flow*, Hemisphere, New York, 1980.
9. C. M. Rhie and W. L. Chow, 'A numerical study of the turbulent flow past an isolated airfoil with trailing edge separation', *AIAA J.*, **21,** 1525–1532 (1983).
10. M. Perić, 'A finite volume method for the prediction of three-dimensional fluid flow in complex ducts', *Ph.D. Thesis*, University of London, London, 1985.
11. I. Demirdžić, A. D. Gosman, R. I. Issa and M. Perić, 'A calculation procedure for turbulent flow in complex geometries', *Comput. Fluids*, **15,** 251–273 (1987).
12. I. Demirdžić and M. Perić, 'Finite volume method for prediction of fluid flow in arbitrary shaped domains with moving boundary', *Int. j. numer. methods fluids*, **10,** 771–790 (1990).
13. C. W. Hirt, A. A. Amsden, J. L. Cook, 'An arbitrary Lagrangean–Eulerian computing method for all flow speeds', *J. Comput Phys.*, **14,** 226–253 (1974).
14. K. C. Karki and S. V. Patankar, 'Pressure based calculation procedure for viscous flows at all speeds in arbitrary configurations', *AIAA J.*, **27,** 1167–1174 (1989).
15. I. Demirdžić, R. I. Issa and, Ž. Lilek, 'Solution method for viscous flows at all speeds in complex domains', in P. Wesseling (ed.), *Notes on Numerical Fluid Mechanics*, Vol. 29, Vieweg, Braunschweig, 1990.
16. R. I. Issa and F. C. Lockwood, 'On the prediction of two-dimensional supersonic viscous interactions near walls', *AIAA J.*, **27,** 182–188 (1977).
17. S. V. Patankar and D. B. Spalding 'A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows', *Int. J. Heat Mass Transfer*, **15,** 1787–1806 (1972).
18. J. C. Slattery, *Momentum, Energy and Mass Transfer in Continua*, McGraw-Hill, New York, 1972.
19. A. H. Shapiro, *The Dynamics and Thermodynamics of Compressible Fluid Flow*, Wiley, New York, 1953.
20. M. Perić, 'Analysis of pressure–velocity coupling on non-orthogonal grids', *Numer. Heat Transfer*, Part B, **17,** 63–82 (1990).
21. C. Hirsch, *Numerical Computation of Internal and External Flows*, Wiley, New York, 1991.
22. H. L. Stone, 'Iterative solution of implicit approximations of multidimensional partial differential equations', *SIAM J. Numer. Anal.*, **5,** 530–558 (1968).
23. C. Rossow, 'Comparison of cell centred and cell vertex finite volume schemes', in M. Deville (ed.), *Notes on Numerical Fluid Mechanics*, Vol. 20, Vieweg, Braunschweig, 1988.
24. I. Demirdžić, Ž. Lilek and M. Perić, 'Fluid flow and heat transfer test problems for non-orthogonal grids: bench-mark solutions', *Int. j. numer. methods fluids*, **15,** 329–354 (1992).
25. F. Bošnjaković, *Technische Thermodynamik*, Steinkopff, Dresden, 1972.
26. A. Rizzi and H. Viviand (eds), 'Numerical methods for the computation of inviscid transonic flows with shock waves', in *Notes on Numerical Fluid Mechanics*, Vol. 3, Vieweg, Braunschweig, 1981.
27. S. Eidelman, A. D. Colella and R. P. Shreeve, 'Application of the Godunov method and its second-order extension to cascade flow modelling', *AIAA J.*, **22,** 1609–1615 (1984).
28. K. W. Morton and M. F. Paisley, 'A finite volume scheme with shock fitting for steady Euler equations', *J. Comput. Phys.*, **80,** 168–203 (1989).
29. W. T. Thompkins, Jr., S. S. Tong, R. H. Bush, W. J. Usab, Jr. and R. J. G. Norton, 'Solution procedures for accurate numerical simulations of flow in turbomachinery', *AIAA Paper 83–0257*, 1983.
30. M. O. Bristeau (ed.), 'Numercial simulation of compressible Navier–Stokes flows', in *Notes on Numerical Fluid Mechanics*, Vol. 18, Vieweg, Braunschweig, 1987.
31. C. Chang and C. Merkle, 'The relation between flux vector splitting and parabolized schemes', *J. Comput. Phys.*, **80,** 334–361 (1989).
32. C. Weiland, M. Pfitzner and G. Hartmann, 'Euler solvers for hypersonic aerothermodinamic problems', in M. Deville (ed.), *Notes on Numerical Fluid Mechanics*, Vol. 20, Vieweg, Braunschweig, 1988.
33. M. Hortmann, M. Perić and G. Scheuerer, 'Finite volume multigrid prediction of laminar natural convection: benchmark solutions', *Int. j. numer. methods fluids*, **11,** 189–207 (1990).