Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 217 (2006) 277–294
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

Short note

A superlinearly convergent Mach-uniform finite volume method for the Euler equations on staggered unstructured grids

D. Vidović *,[1], A. Segal, P. Wesseling

J.M. Burgers Center and Department of Applied Mathematical Analysis, Delft University of Technology, Mekelweg 4, 2628 CD Delft, The Netherlands

Received 4 August 2004; received in revised form 10 January 2006; accepted 17 January 2006
Available online 2 March 2006

Abstract

A Mach-uniform finite volume scheme for solving the unsteady Euler equations on staggered unstructured triangular grids that uses linear reconstruction is described. The scheme is applied to three benchmark problems and is found to be considerably more accurate than a similar scheme based on piecewise constant reconstruction.

PACS: 47.11.+j

MSC: 76M12

Keywords: Staggered unstructured grid; Finite volume method; Compressible flow; Euler equations; Ringleb; Mach-uniform

1. Introduction

Numerical schemes to solve the Euler or Navier–Stokes equations on staggered unstructured triangular grids can be found in [1–8]. In [8], we presented a scheme for solving the unsteady incompressible Navier–Stokes equations on staggered unstructured triangular grids. This method uses linear reconstruction to achieve second-order accuracy. In [7], a scheme was presented that is first-order accurate, and can handle both incompressible and compressible flow in a unified way. The aim of the present paper is to increase the accuracy of the scheme presented in [7] using the linear reconstruction of staggered vector fields as described in [8]. Linear reconstruction of scalar quantities is also necessary and will be presented. Our new method solves the Mach-uniform Euler equations with second-order accuracy.

We briefly review related work on unstructured staggered schemes and on Mach-uniform methods. The covolume method described in [1–4] uses Delaunay grids and a dual mesh obtained by connecting the

* Corresponding author. Tel.: +31 152781692; fax: +31 152787245.
E-mail address: D.Vidovic@ewi.tudelft.nl (D. Vidović).
[1] Supported by the Technology Foundation STW, applied science division of NWO and the technology program of the Ministry of Economic Affairs (Project DWI.5552).

0021-9991/$ - see front matter © 2006 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2006.01.031

circumcenters. Scalar variables are located in the circumcenters, and as in all other schemes mentioned here except in [6], velocity components normal to the faces of the primal grid are stored. A similar scheme is described in [5], where conservation of kinetic energy and momentum are emphasised.

Schemes that use circumcenters require grids to be of Delaunay type. In our method, scalar variables are stored in centroids, and the Delaunay property is not required.

The scheme presented in [6] uses centroids instead of circumcenters, and stores momentum vectors in faces or vertices. The scheme described in [7] also uses centroids, but it stores the normal velocity components in faces. This is also the approach taken in [8].

Except for [8], the schemes described are first-order accurate. Our scheme is second-order accurate.

Except for [2,7], these papers are restricted to incompressible flows. Staggered schemes have an advantage over collocated ones in the incompressible limit, because they do not require numerical stabilization to couple velocity and pressure. In general, there is no reason to use staggered schemes for fully compressible flow. However, in certain applications, as for example flow in internal combustion engines or flow around aircraft in take-off or landing conditions, compressible and incompressible regions occur simultaneously. Standard compressible schemes suffer from efficiency and accuracy loss when the Mach number becomes small (below 0.2). Such problems require Mach-uniform methods that can handle flows at all speeds. Staggering of the grid is attractive for Mach-uniform methods, because of superior properties in the incompressible case. Mach-uniform methods that use structured staggered grids were presented in [9,10], and more recently in [11,12]. The only Mach-uniform method that uses staggered unstructured grids that we know of is presented in [7]. This method we extend here to second-order accuracy.

In Section 2 a dimensionless form of the Euler equations is presented that is convenient for Mach-uniform schemes. Section 3 discusses a Mach-uniform pressure-correction method. Spatial discretization is presented in Section 4. Linear reconstruction is discussed in Section 5, and in Section 6 we consider flux limiting methods. Finally, numerical results are presented in Section 7.

2. Governing equations

The Euler equations are given by:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\vec{u}\rho) = 0, \tag{1}$$

$$\frac{\partial m_\alpha}{\partial t} + \nabla \cdot (\vec{u} m_\alpha) = -p_{,\alpha}, \tag{2}$$

$$\frac{\partial \rho E}{\partial t} + \nabla \cdot (\vec{u}\rho H) = 0, \tag{3}$$

where ρ is the density, p is the pressure, $\vec{m} = \rho\vec{u}$ is the momentum, $\vec{u}$ is the velocity, $E = e + \frac{1}{2}\vec{u}\cdot\vec{u}$ is the total energy and $H = h + \frac{1}{2}\vec{u}\cdot\vec{u}$ is the total enthalpy. Greek indices indicate coordinate directions; Latin indices will refer to locations in a computational grid. We have $h = \gamma e$ with γ the ratio of specific heats, and the equation of state for a perfect gas can be written as $p = (\gamma - 1)\rho e$. By using the equation of state the energy equation (3) can be written as

$$\frac{\partial}{\partial t}\left[p + \frac{1}{2}(\gamma - 1)\rho\vec{u}^2\right] + \nabla \cdot \left[\vec{u}\left(\gamma p + \frac{1}{2}(\gamma - 1)\rho\vec{u}^2\right)\right] = 0, \tag{4}$$

where $\vec{u}^2 = \vec{u}\cdot\vec{u}$. This is the form on which the Mach-uniform pressure-correction method presented in Section 3 is based.

Let $\hat{u}_r$, L, $\hat{\rho}_r$ and $\hat{T}_r$ be the reference velocity, length, density, and temperature. We calculate the reference pressure from the equation of state

$$\hat{p}_r = R\hat{\rho}_r\hat{T}_r, \tag{5}$$

and the reference Mach number is

$$M_r = \hat{u}_r \Big/ \sqrt{\gamma R \hat{T}_r}. \tag{6}$$

If we use the following dimensionless quantities: $x_\alpha = \hat{x}_\alpha/L$, $t = \hat{t}\hat{u}_\mathrm{r}/L$, $T = \hat{T}/\hat{T}_\mathrm{r}$, $\rho = \hat{\rho}/\hat{\rho}_\mathrm{r}$, $u_\alpha = \hat{u}_\alpha/\hat{u}_\mathrm{r}$, and $p = \hat{p}/\hat{p}_\mathrm{r}$, where $\hat{t}$, $\hat{T}$, $\hat{\rho}$, $\hat{u}_\alpha$, and $\hat{p}$ are dimensional quantities, then the dimensionless momentum equation is found to be

$$\frac{\partial m_\alpha}{\partial t} + \nabla\cdot(\vec{u}m_\alpha) = -\frac{1}{\gamma M_\mathrm{r}^2}p_{,\alpha}. \tag{7}$$

This equation is singular as $M_\mathrm{r} \to 0$. The resulting difficulties are discussed in [13,7]. In order to obtain Mach-uniform dimensionless equations, instead of $p = \hat{p}/\hat{p}_\mathrm{r}$ we shall use the following dimensionless pressure, as in [7,11,12]:

$$p = \frac{\hat{p} - \hat{p}_\mathrm{r}}{\hat{\rho}_\mathrm{r}\hat{u}_\mathrm{r}^2}. \tag{8}$$

The dimensionless momentum and mass conservation equations have the same form as their dimensional counterparts (1) and (2), whereas the dimensionless version of the pressure-based energy equation (4) is

$$M_\mathrm{r}^2\left\{\frac{\partial}{\partial t}\left[p + \frac{1}{2}(\gamma-1)\rho\vec{u}^2\right] + \nabla\cdot\left[\vec{u}\left(\gamma p + \frac{1}{2}(\gamma-1)\rho\vec{u}^2\right)\right]\right\} + \nabla\cdot\vec{u} = 0, \tag{9}$$

and the dimensionless equation of state is

$$h = \frac{1}{\rho}(1 + \gamma M_\mathrm{r}^2 p). \tag{10}$$

When $M_\mathrm{r} = 0$, Eq. (9) reduces to the incompressible continuity equation.

3. Pressure-correction method

The Mach-uniform pressure-correction sequential update procedure to be used has been presented in [7,11]. In this section we discuss only the time discretization.

Initial conditions determine the starting solution vectors m^0, ρ^0, and p^0. First the new density is computed from the discretized mass conservation equation

$$\frac{\rho^{n+1} - \rho^n}{\Delta t} + \nabla\cdot(\vec{u}^n\rho^{n+1}) = 0, \tag{11}$$

where the superscripts indicate the time level, and Δt is the time step.

Next, a prediction of the momentum field m_α^* is computed from the momentum equation:

$$\frac{m_\alpha^* - m_\alpha^n}{\Delta t} + \nabla\cdot(\vec{u}^n m_\alpha^*) = -p_{,\alpha}^n. \tag{12}$$

Here Picard linearization is applied to the nonlinear convection term (i.e. $\vec{u}$ is taken at the old time level). The following pressure-correction is postulated:

$$\vec{m}^{n+1} = \vec{m}^* - \Delta t\nabla\delta p, \quad \delta p = p^{n+1} - p^n. \tag{13}$$

Discretizing (9) in time with the implicit Euler scheme, inserting (13) and substituting

$$\vec{u}^* = \vec{m}^*/\rho^{n+1} \tag{14}$$

in the dimensionless pressure-based energy equation (9) leads to

$$\begin{aligned} M_\mathrm{r}^2\Bigg\{&\frac{\delta p}{\Delta t} + \frac{1}{2}(\gamma-1)\frac{(\vec{m}^* - \Delta t\nabla\delta p)^2/\rho^{n+1} - (\vec{m}^n)^2/\rho^n}{\Delta t} \\ &+\nabla\cdot\left[\left(\vec{u}^* - \frac{\Delta t}{\rho^{n+1}}\nabla\delta p\right)\left(\gamma(p^n + \delta p) + \frac{1}{2}(\gamma-1)(\vec{m}^* - \Delta t\nabla\delta p)^2/\rho^{n+1}\right)\right]\Bigg\} \\ &+\nabla\cdot\left(\vec{u}^* - \frac{\Delta t}{\rho^{n+1}}\nabla\delta p\right) = 0. \end{aligned} \tag{15}$$

Linearization and some additional simplification gives

$$M_r^2\left\{\frac{\delta p}{\Delta t}+\frac{1}{2}(\gamma-1)\frac{\left[(\vec{m}^*)^2-2\Delta t\vec{m}^*\cdot\nabla\delta p\right]/\rho^{n+1}-(\vec{m}^n)^2/\rho^n}{\Delta t}\right\}$$
$$+\nabla\cdot\left[\vec{u}^*\left(1+\gamma M_r^2(p^n+\delta p)+\frac{1}{2}(\gamma-1)M_r^2(\vec{m}^*)^2/\rho^{n+1}\right)\right]$$
$$-\Delta t\nabla\cdot\left\{\left[\left(1+M_r^2p^n+\frac{1}{2}(\gamma-1)M_r^2(\vec{m}^*)^2\rho^{n+1}\right)\rho^{n+1}\right]\nabla\delta p\right\}=0. \tag{16}$$

For detailed derivation of this equation see [7,11]. It is used to compute δp. Finally the new pressure and momentum are computed from (13).

4. Discretization on unstructured triangular staggered grids

An unstructured triangular grid will be used. We choose the normal momentum components averaged over grid edges

$$m_e=\frac{1}{l_e}\int_{l_e}(\vec{m}\cdot\vec{N}_e)\,\mathrm{d}l \tag{17}$$

as primary momentum unknowns, where $\vec{N}_e$ is one of the two possible unit normal vectors in edge e, end l_e is the length of edge e. Scalar variables are associated with cells, as shown in Fig. 1.

4.1. Momentum equation

The semi-discretized momentum equation (12) is multiplied by $\vec{N}_i$ and integrated over a control volume Ω_i consisting of the two triangles sharing edge i, as illustrated in Fig. 2 on the right. Apart from the flux limiting, the momentum equation is discretized in the same way as in the incompressible case presented in [8] and therefore we shall not discuss the discretization of the time derivative and the pressure term. The divergence theorem is applied to the convective term:

$$\int_{\Omega_i}\nabla\cdot[\vec{u}(\vec{m}\cdot\vec{N}_i)]\,\mathrm{d}\Omega=\oint_{\partial\Omega_i}(\vec{u}\cdot\vec{n})(\vec{m}\cdot\vec{N}_i)\,\mathrm{d}\Gamma\approx\sum_e u_e(\vec{m}_e\cdot\vec{N}_i)\bar{l}_e, \tag{18}$$

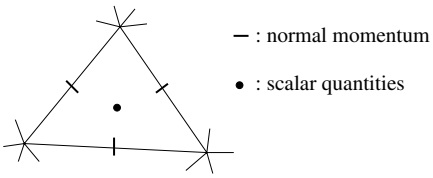

Fig. 1. Staggered positioning of the variables in an unstructured grid.

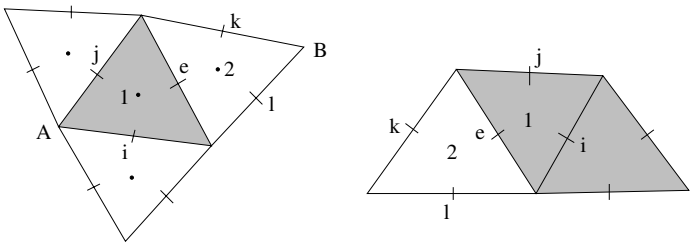

Fig. 2. Control volumes used for the scalar equations (left) and the momentum equation (right).

where $\overline{l}_e = (\vec{N}_e \cdot \vec{n}_e) l_e$, and $\vec{n}_e$ is the outward normal in face e. In [7], the term $\vec{m}_e \cdot \vec{N}_i$ is reconstructed from faces k and l if the flow is directed from cell 2 to cell 1, or it is assumed that $\vec{m}_e \cdot \vec{N}_i \approx m_i$ if the flow is directed from cell 1 to cell 2. In [8], a linear reconstruction with special treatment of the divergence is employed to approximate $\vec{m}_e \cdot \vec{N}_i$. In order to damp numerical wiggles, we use a linear combination of these two schemes. This will be explained in Section 6.

The reasons to use the divergence-free reconstruction polynomials even when the velocity field is not divergence free are the following: first, it needs twice less memory than the non-divergence-free reconstruction. Second, in some zones of the flow velocity may be divergence-free. Third, the divergence is computed exactly with the divergence-free reconstruction method even if it is not zero.

4.2. Density equation

The semi-discretized density equation (11) is integrated over each triangle of the mesh, and the divergence theorem is applied to the convective term. Taking the triangle Ω_1 depicted in the left part of Fig. 2 for example, one obtains

$$\int_{\Omega_1} \frac{\rho^{n+1} - \rho^n}{\Delta t} \, \mathrm{d}\Omega + \oint_{\partial\Omega_1} \rho^{n+1} (\vec{u}^n \cdot \vec{n}) \, \mathrm{d}\Gamma = 0. \tag{19}$$

We take the triangle averages of ρ as primary density variables. After approximating the line integral in the convective term, Eq. (19) reduces to

$$|\Omega_1| \frac{\rho_1^{n+1} - \rho_1^n}{\Delta t} + \sum_e \rho_e^{n+1} u_e^n \overline{l}_e = 0, \tag{20}$$

where summation takes place over the three faces.

In the first-order upwind scheme [7] the density in face e is taken to be

$$\rho_e^{n+1} = \begin{cases} \rho_1^{n+1} & \text{if } u_e^n \overline{l}_e \geqslant 0, \\ \rho_2^{n+1} & \text{if } u_e^n \overline{l}_e < 0, \end{cases} \tag{21}$$

if e is an internal face, and $\rho_e^{n+1} = \rho_1^{n+1}$ if e is a boundary face where ρ is not prescribed. Note that if $u_e^n \overline{l}_e > 0$ then the flow direction is out of cell 1, otherwise into.

In order to obtain a more accurate scheme, we use upwind-biased linear interpolation to approximate ρ_e^{n+1}. This will be discussed in Section 5.

The convecting velocity u_e^n is computed by dividing the convected momentum m_e^n by the density ρ_e^n, which needs to be computed. In the first-order scheme it is computed as an area-weighted average of the densities in the two neighboring cells:

$$\rho_e^n = \frac{\Omega_2}{\Omega_1 + \Omega_2} \rho_1^n + \frac{\Omega_1}{\Omega_1 + \Omega_2} \rho_2^n. \tag{22}$$

If e is a boundary face and cell 2 is missing then we take $\rho_e^n = \rho_1^n$.

In the new scheme we use face-centered linear reconstruction (see Section 5) to compute ρ_e^n.

4.3. Mach-uniform pressure-correction equation

The Mach-uniform pressure-correction equation (16) is discretized in [7] in a similar way as the continuity equation. It is integrated over the shaded control volume shown in the left part of Fig. 2. Before explaining how we modified this discretization in order to obtain higher-order accuracy, we shall explain how it is done in [7].

The momentum vector in the time derivative is reconstructed from the three faces of the control volume using least squares approximation. The gradient of the pressure correction is approximated by a least squares fit to the normal pressure gradients in the faces of the control volume, which are computed by the path integral method (see [14]). If the normal momentum component is prescribed in one of the control volume faces, then it follows from (13) that in this face $\nabla \delta p = 0$ because $m^* = m^{n+1}$.

The divergence theorem is applied to the convection and the second-order term (second and third row in the formula (16)). Convected quantities in the convection term are computed in an upwind manner: density, pressure and the pressure-correction in face e are evaluated in cell 1 if the flow is outward (see the left part of Fig. 2), or in cell 2 otherwise. The momentum vector is reconstructed from faces i and j or from faces k and l, depending on the direction of the flow. Central approximations of momentum, pressure and density in faces are used in the second-order term. For details see [7].

In order to obtain higher-order accuracy, we modified this scheme by introducing linear reconstruction. Momentum in the time derivative is computed as the average of momenta reconstructed in the vertices of the control volume by using the divergence-free linear reconstruction polynomials (see [8]) based in three vertices of the control volume. The divergence is reconstructed by applying the divergence formula to the control volume. Momentum in the convective term in face e is obtained from the reconstruction polynomial based in vertex A if the fluid flows into the control volume and the divergence is computed from cell 1, otherwise vertex B and cell 2 are used. Pressure, pressure-correction and density are reconstructed by using upwind-biased cell-based or vertex-based linear reconstruction, which will be presented in Section 5. The second-order term vanishes for the steady solution, and since our scheme is first-order accurate in time, there is no need to use linear reconstruction for this term.

5. Linear reconstruction of scalars

A convected scalar quantity ψ is reconstructed in face e (see the left part of Fig. 2) by using the formula

$$\psi_e = \begin{cases} \psi_1 + \nabla\psi_{1,e} \cdot (\vec{r}_e - \vec{r}_1) & \text{if } u_e^n \bar{l}_e \geqslant 0, \\ \psi_2 + \nabla\psi_{2,e} \cdot (\vec{r}_e - \vec{r}_2) & \text{if } u_e^n \bar{l}_e < 0, \end{cases} \tag{23}$$

where $\vec{r}$ is the position vector. The gradient $\nabla\psi$ is determined by least squares approximation either from the cells surrounding the upwind cell (1 or 2) or from the cells surrounding the upwind vertex (A or B), as illustrated in Fig. 3. These two methods will be called *cell-based* and *vertex-based reconstruction*, respectively. Cell-based reconstruction typically involves fewer cells (cf. Fig. 3), and gives a unique gradient per cell (i.e. $\nabla\psi_{2,e}$ does not depend on e), which does not hold for the vertex-based reconstruction. The vertex-based reconstruction uses quantities from the upwind side of the edge only; this may enhance monotonicity of numerical solutions. Both approaches were tested and results are presented in Section 7.

Face-based reconstruction of the density needed to compute the convecting velocity in face e (see Section 4.2) is obtained as an average of the cell-based reconstruction polynomials based in cells 1 and 2.

We considered two ways to determine the gradient using the vertex-based reconstruction. One possibility, assuming $u_e \bar{l}_e < 0$, is to find a and $\nabla\psi_B = [b, c]$ such that a linear polynomial

$$P_B(\vec{r}) = a + \nabla\psi_B \cdot (\vec{r} - \vec{r}_B) \tag{24}$$

matches the values of the scalar in the cells surrounding vertex B as accurately as possible in the least-squares sense, and to set $\nabla\psi_{2,e} \equiv \nabla\psi_B$ in (23). The other possibility is to postulate, similar to (23),

$$\psi(\vec{r}) = \psi_2 + \nabla\psi_{2,e} \cdot (\vec{r} - \vec{r}_2) \tag{25}$$

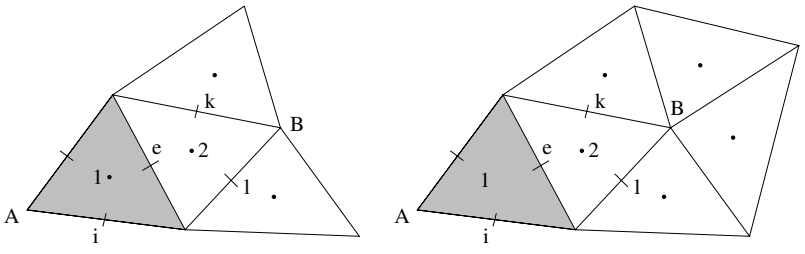

Fig. 3. Reconstruction stencil in the case when fluid flows into the control volume based in the upwind cell (left) and the upwind vertex (right). Cells that are used for reconstruction are marked by dots.

and to determine $\nabla\psi_{2,e}$ by least squares approximation. In the first case one linear polynomial is associated with each vertex. In the second case two linear polynomials are associated with each face, one for each possible flow direction. Since there are roughly three times more faces then vertices, the second method requires six times more memory, while it is only a bit more accurate.

We also tried not to ignore the constant part a of the linear polynomial (24) and to use it instead of ψ_2. However, the resulting scheme did not converge in time.

6. Monotonicity considerations

In the vicinity of steep gradients or discontinuities, spurious oscillations may occur. Not only are these oscillations non-physical, but if they become too large, the density may get close to zero. The convecting velocity is computed by dividing the momentum by the density, and therefore it may become very large, so the solution procedure breaks up.

These spurious wiggles are a well-known phenomenon, which was investigated by Godunov in [15]. His order barrier theorem shows that linear non-oscillatory schemes are at most first-order accurate. For this reason every higher-order non-oscillatory scheme must be nonlinear.

Several concepts exist that ensure that a scheme is non-oscillatory. One of them is *monotonicity preservation*, a requirement that new extrema cannot appear in the solution. Another one is that the scheme is *total variation diminishing* (TVD), which means that the total variation of the solution cannot grow in time. One which is easier to enforce is that a scheme is *local extremum diminishing* (LED), which means that local maxima cannot increase and local minima cannot decrease. All these requirements indeed hold for the exact solution (for scalar problems only).

Numerous authors have developed methods to enforce these requirements. For a survey, see Section 9.4 of [13]. One possible way to ensure that a scheme is LED is flux limiting. The approximation (23) is replaced by

$$\psi_e = \begin{cases} \psi_1 + \phi_1 \nabla\psi_{1,e} \cdot (\vec{r}_e - \vec{r}_1) & \text{if } u_e^n \bar{l}_e \geqslant 0, \\ \psi_2 + \phi_2 \nabla\psi_{2,e} \cdot (\vec{r}_e - \vec{r}_2) & \text{if } u_e^n \bar{l}_e < 0, \end{cases} \tag{26}$$

where ϕ is so-called *flux limiter*, determined such that

$$\min_j(\psi_j, \psi_i) \leqslant \psi_i + \phi_i \nabla\psi_{i,e} \cdot (\vec{r}_e - \vec{r}_i) \leqslant \max_j(\psi_j, \psi_i), \quad \forall e, \tag{27}$$

where the minimum and the maximum are taken over cell i and the surrounding cells. By *surrounding cells* we mean the cells that have a common edge with cell i. Quantities that are evaluated at boundaries are not limited. This is a simplified variant of the criterion that Barth uses in [16,17]. It is a necessary and, according to [18], close to sufficient condition for a scheme to be LED. We compute the limiter of Barth and Jespersen [16,19] in the following way:

$$\Theta_{i,e} = \nabla\psi_{i,e} \cdot (\vec{r}_e - \vec{r}_i), \tag{28}$$

$$\phi_{i,e} = \begin{cases} \min\left(1, \frac{\max_j(\psi_j,\psi_i) - \psi_i}{\Theta_{i,e}}\right) & \text{if } \Theta_{i,e} > 0, \\ \min\left(1, \frac{\min_j(\psi_j,\psi_i) - \psi_i}{\Theta_{i,e}}\right) & \text{if } \Theta_{i,e} < 0, \\ 1 & \text{if } \Theta_{i,e} = 0, \end{cases} \tag{29}$$

and a unique value of the limiter per cell is taken to be

$$\phi_i = \min_e(\phi_{i,e}). \tag{30}$$

In smooth regions where oscillations usually do not appear, the value of ϕ is close to one, while near discontinuities it is close to zero. In this way we switch to the first-order scheme when the second-order scheme is oscillatory. If only the density equation is solved, the first-order scheme is the same as the one presented in [16], and for this one it has been proven that it is non-oscillatory. It is essential that $\nabla\psi_{i,e}$ in the limiter (28) is computed in the same way as the gradient in the reconstruction (26). If we use the cell-based gradient

in the limiter and the vertex-based gradient in the reconstruction, the oscillations will not be successfully damped, as we have found.

One drawback of the limiter of Barth and Jespersen is that it prevents convergence to a truly steady state. Venkatakrishnan suggests in [20] that the reason for this is the non-differentiability of min and max functions, and proposes the following modification to the limiter of Barth and Jespersen:

$$\phi_{i,e} = \Psi(\Delta_i, \Theta_{i,e}) = \frac{\Delta_i^2 + 2\Delta_i\Theta_{i,e} + \varepsilon^2}{\Delta_i^2 + \Delta_i\Theta_{i,e} + 2\Theta_{i,e}^2 + \varepsilon^2}, \tag{31}$$

$$\Delta_i = \begin{cases} \max\limits_j(\psi_j, \psi_i) - \psi_i & \text{for } \Theta_{i,e} > 0, \\ \min\limits_j(\psi_j, \psi_i) - \psi_i & \text{for } \Theta_{i,e} < 0, \\ 1 & \text{for } \Theta_{i,e} = 0, \end{cases} \tag{32}$$

where

$$\varepsilon^2 = (Kh)^3 \tag{33}$$

is a relaxation factor aimed to prevent limiting of smooth extrema, h is a local mesh parameter, and K is a case-dependent constant. The original limiter of Barth and Jespersen can also be modified to include such a relaxation factor.

However, it is noted in [18] that Venkatakrishnan's limiter also may suffer from stagnation in convergence to steady state, and our experience confirms this. Therefore we use the so-called *historic modification* of the limiter proposed by Delanaye [21]: after a certain number of iterations, the limiter is computed as a minimum of the limiters in the previous and the current time step. This number of iterations is again case dependent, and should be taken large enough. The historic modification leads to the steady state in every case that we tried.

Although it is clear how to limit the convected density in the mass equation, it is not straightforward to decide how to handle the convective term of the energy equation (16). If we limit only δp and not p^n, the scheme does not work. The reason might be that it is actually the new pressure that we are computing here, i.e. $p^n + \delta p$. On the other hand, our experiments have shown that limiting the convected kinetic energy term $\frac{1}{2}(\gamma - 1)M_r^2\vec{m}^* \cdot \vec{m}^*/\rho^{n+1}$ does not make any significant difference.

However, the concepts of monotonicity preservation, TVD and LED are not easily extendible to staggered unstructured meshes. It is not clear what an extremum in the discrete momentum field should be, because this field consists of incomparable normal components. This is why it is hard to design an oscillation detector for the momentum. Furthermore, even the first-order upwind scheme [7] is not completely oscillation-free. Very small oscillations in the pressure and the density occur around discontinuities, though they are usually not visible. In the example presented in Section 7.2.2, the Mach number obtained by the first-order scheme has an oscillation of 0.6% of the jump in Mach.

Fortunately, oscillations typically appear in all variables simultaneously. Therefore we can determine the oscillation indicator ϕ from some scalar quantity. For this purpose we use the density. When strong shocks are present, it may be advantageous to use the inverse of the density, in order to switch to the first-order reconstruction when the density becomes small, otherwise the scheme may break down.

Another problem is that we cannot use a formula similar to (26) for the momentum, because we do not have the momentum vector in any point. Adding the gradient to a first-order reconstruction used in [7] cannot result in a second-order reconstruction, because there is no point where this first-order reconstruction is more than first-order accurate. If we remove the gradient part from the vertex-based divergence-free momentum reconstruction, this does not damp the wiggles.

A higher-order limited scheme for momentum is obtained as follows. We shall denote the piecewise constant approximation of $\vec{m}_e \cdot \vec{N}_i$ used in scheme [7] by $(\vec{m}_e \cdot \vec{N}_i)^1$ and the piecewise linear approximation used in [8] by $(\vec{m}_e \cdot \vec{N}_i)^2$. The approximation that we use here is a linear combination of these two, i.e.

$$\vec{m}_e \cdot \vec{N}_i \approx (1 - \phi)(\vec{m}_e \cdot \vec{N}_i)^1 + \phi(\vec{m}_e \cdot \vec{N}_i)^2. \tag{34}$$

Of course, we cannot expect that spurious wiggles will be totally eliminated, because the stencil that we use for the reconstruction of momentum is inevitably different from the one that is used to compute the limiter, among other reasons.

7. Numerical results

The purpose of the numerical experiments described here is to show improved accuracy compared to the first-order upwind method described in [7].

7.1. Ringleb flow

For this transonic flow in a curved duct the analytic solution of the potential flow equations and hence Euler equations is known. It was first published in [22], and a description may be found in [23]. This is a valuable test case to investigate the accuracy of numerical schemes for transonic flow. The flow is directed upwards in Fig. 4. There is a small supersonic flow region in the middle of the right wall, but there is no shock. The maximum Mach number is 1.09. The problem was solved on regular and distorted grids (see Figs. 4 and 7) using the first-order and the second-order method. The exact Mach isolines are circular arcs. The exact velocity and density were prescribed along the inlet (lower boundary), normal velocity was set to zero along the left and the right boundary, and the pressure was prescribed along the outflow (upper) boundary. The grids with 10–100 cells along the inlet were generated independently from each other, preserving the quality of the grids shown in Figs. 4 and 7.

We denote the local truncation error of the momentum, the density and the pressure equation by τ_m, τ_ρ and τ_p, respectively. The (global truncation) error of the momentum, the density, and the pressure will be denoted by ε_m, ε_ρ and ε_p, respectively. The accuracy of a scheme is usually estimated either in terms of the local truncation errors, or one refers to the maximal-order of a polynomial that can be exactly recovered by the reconstruction method being used. However, an example of a scheme which does not converge in the sense that the global truncation errors do not converge to zero, even though linear reconstruction is used and the local truncation error converges with second-order accuracy, can be found in [8]. Therefore we examine the global truncation error here.

The order of accuracy α is estimated in the following way. It is assumed that some norm of the error depends on h as $\|\varepsilon\| \approx ah^\alpha$, $h \downarrow 0$, where h is a mesh parameter (in our case, $h = 1/n$, n is the number of edges along the inlet), α is the order of accuracy, and a is some constant. This can be written as

$$\frac{\log a}{\log h} + \alpha = \frac{\log \|\varepsilon\|}{\log h}. \tag{35}$$

We estimate $\log a$ and α by least squares from a set of results obtained with meshes of the same quality and various h.

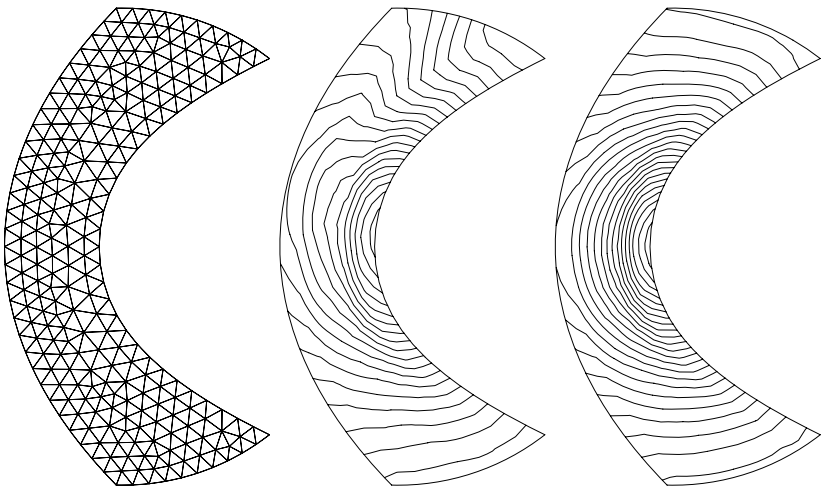

Fig. 4. Ringleb flow, regular mesh (left) and Mach isolines obtained with the first-order (middle) and the second-order unlimited (right) method.

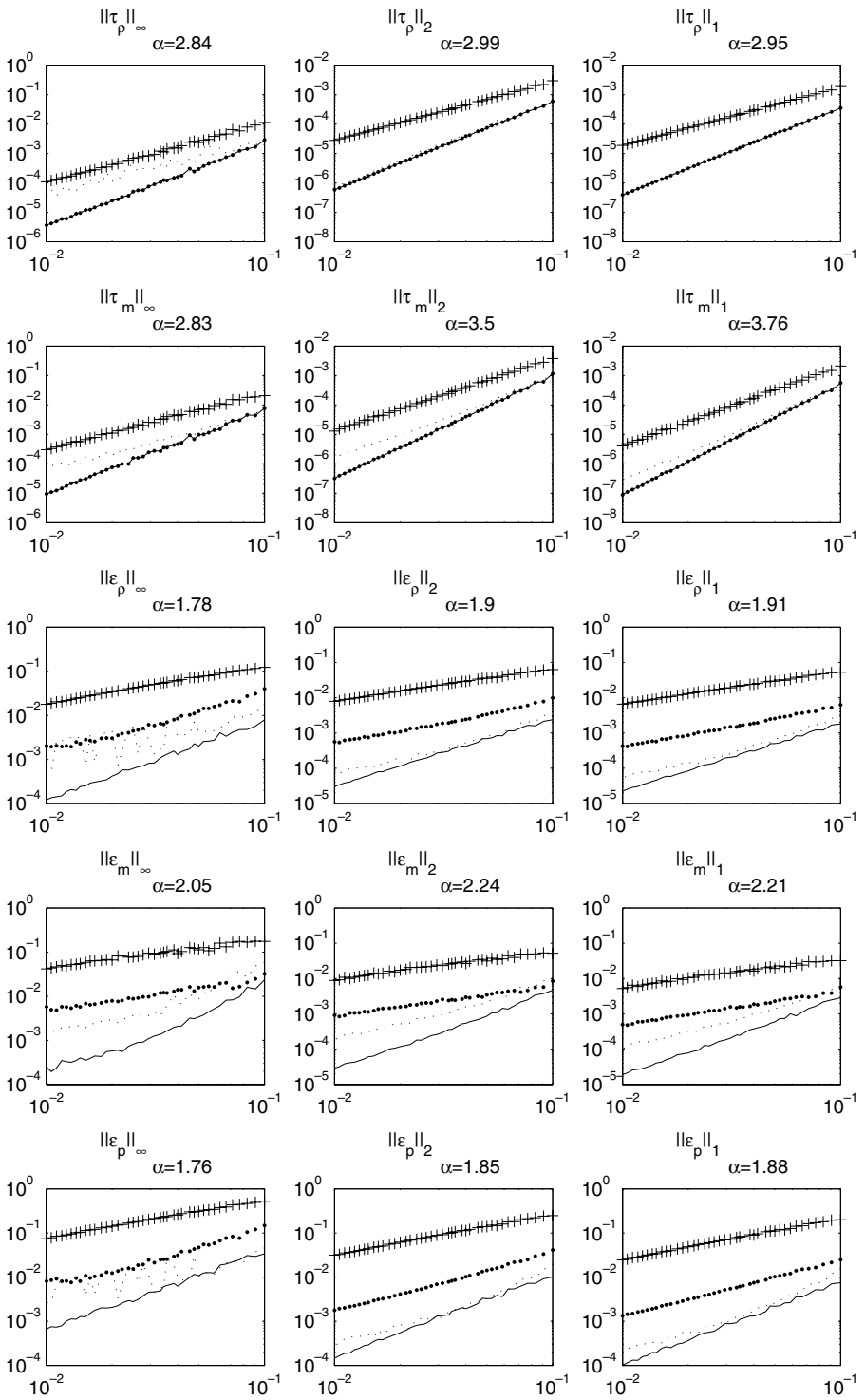

Fig. 5. Ringleb flow, regular grids; norms of local and global truncation errors versus the mesh parameter. "+": first-order scheme; ".": vertex-based reconstruction of all convected variables, no limiter; "–": vertex-based reconstruction of the density, cell-based reconstruction of the pressure, no limiter; "·": vertex-based reconstruction of the density, cell-based reconstruction of the pressure, unrelaxed Venkatakrishnan limiter. The estimate of the rate of convergence α refers to the results marked "–".

Fig. 5 shows the error of the solution obtained on the regular grid. The independent parameter along the horizontal axis is the mesh parameter, and the parameter along the vertical axis is a norm of an error.

Fig. 5 gives local (τ) and global (ε) truncation error norms of the first-order scheme and of three variants of the scheme with linear reconstruction. The best results were obtained with the vertex-based scalar reconstruction in the density equation, and the cell-based reconstruction in the pressure equation. If the vertex-based reconstruction is used in both equations, the results are less accurate, probably because more artificial dissipation is introduced, or because the cell-based reconstruction of the pressure allows the pressure prescribed at the outlet to propagate. If the cell-based reconstruction is used in the density equation, the scheme does not converge in time when strong shocks are present (see Section 7.2.2), therefore we do not consider this variant. We see that the limited scheme is a bit less accurate than the unlimited one, i.e. the limiter is not completely inactive. If a non-zero relaxation factor is used (see (6)), the difference in accuracy is smaller.

The amount of memory used by these schemes is proportional to the amount of data in the mesh. Schemes that employ linear reconstruction use about 50% more memory than the first-order scheme.

Fig. 6 shows the total time and the number of time steps needed to reach the stationary solution versus the mesh parameter. Since the maximal allowed time step is found to be proportional to the mesh parameter, we choose the time step as $\Delta t = 1.5 * h$. As expected, schemes with linear reconstruction usually need more time to converge to the steady solution than the first-order scheme, although the number of time steps needed is larger in the case of the first-order scheme.

A jump in the total time needed by the first-order scheme to reach the steady state is visible in Fig. 6 on the left, while there is no jump in the number of time steps. This jump is due to memory cashing. It also exists with the other schemes, although it is not so obvious.

On distorted grids (see Figs. 7 and 8) these schemes still work, but they are less accurate than on the regular ones. However, the linear reconstruction improves the accuracy considerably. If the grid is distorted more than shown in Fig. 7, the schemes with linear reconstruction may break up. The time and the number of steps needed to reach the stationary solution are shown in Fig. 9. The time step was chosen as $\Delta t = 1.5 * h$.

7.2. Channel with a bump

The flow in a channel with a circular arc bump is used to evaluate the presented method for computations of steady state solutions. We present the incompressible and the supersonic flow case. In the incompressible flow case the height of the bump is 10% of the channel height, while in the supersonic flow case it is 4%. This benchmark was proposed in [24].

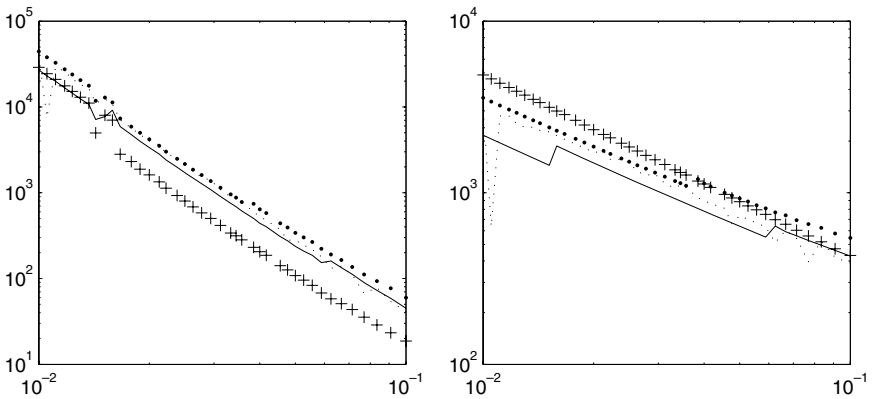

Fig. 6. Ringleb flow, regular grids; total computation time in seconds (left) and the number of time steps needed to reach the stationary solution (right) versus the mesh parameter. "+": first-order scheme; ".": vertex-based reconstruction of all convected variables, no limiter; "–": vertex-based reconstruction of the density, cell-based reconstruction of the pressure, no limiter; "·": vertex-based reconstruction of the density, cell-based reconstruction of the pressure, unrelaxed Venkatakrishnan limiter.

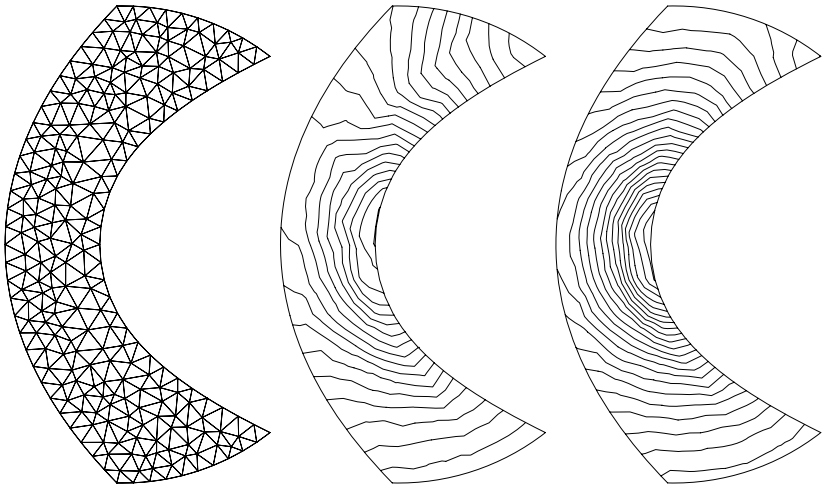

Fig. 7. Ringleb flow, distorted mesh (left) and Mach isolines obtained with the first-order (middle) and the second-order unlimited (right) method.

7.2.1. Incompressible flow case

For $M = 0$ we want to demonstrate that the solution obtained with the Mach-uniform method matches the solution obtained with the incompressible method presented in [8]. The mesh is shown in Fig. 10. The isobars and the pressure distribution along the lower wall obtained with the method for incompressible flow and with the Mach-uniform method are shown in Fig. 11. There is no visible difference between these two solutions. Actual differences are of order 10^{-5}, which we consider reasonable since we are comparing results of different codes.

7.2.2. Supersonic flow case

The mesh used in this case is shown in Fig. 12. The inflow Mach number is 1.65. The Mach number obtained by the first-order scheme has an oscillation before the first shock on the lower wall of about 0.6% of the jump in Mach (see Fig. 13).

If the vertex-based reconstruction is used for the density equation and the cell-based reconstruction is used for the pressure equation, and limiting is applied, the oscillations are not much larger than in the first-order case, and the shock resolution is better. This especially holds for the reflected shock, which is completely blurred in the case of the first-order scheme. If the cell based reconstruction is used for the density, the scheme does not converge in time.

In this example we used time step $\Delta t = 0.001$. The scheme with linear reconstruction took twice more time steps and 4.4 times more computational time to finish, and it used 50% more memory. The difference between the time needed by the first-order scheme and by the second-order scheme to converge to the steady state is larger than in the case of Ringleb flow. The reason is that discontinuities are present.

The results obtained are in good agreement with [24]. The obtained solution is less accurate at the outflow boundary than in the interior (see the left side of Fig. 14), but this is also visible in [24].

7.3. NACA 0012 airfoil

We have computed the flow around the NACA 0012 airfoil for $M = 0.8$ and angle of attack $\alpha = 1.25°$. This transonic benchmark problem was presented in [25]. Mach isolines obtained with the first-order scheme and the superlinear scheme with limiting on the grid shown in Fig. 16 are shown in Fig. 15. Vertex-based scalar reconstruction was used in the density equation, and cell-based scalar reconstruction was used in the Mach-uniform pressure-correction equation. As expected, the shock on the upper surface of the airfoil is sharper in the case of the superlinear scheme.

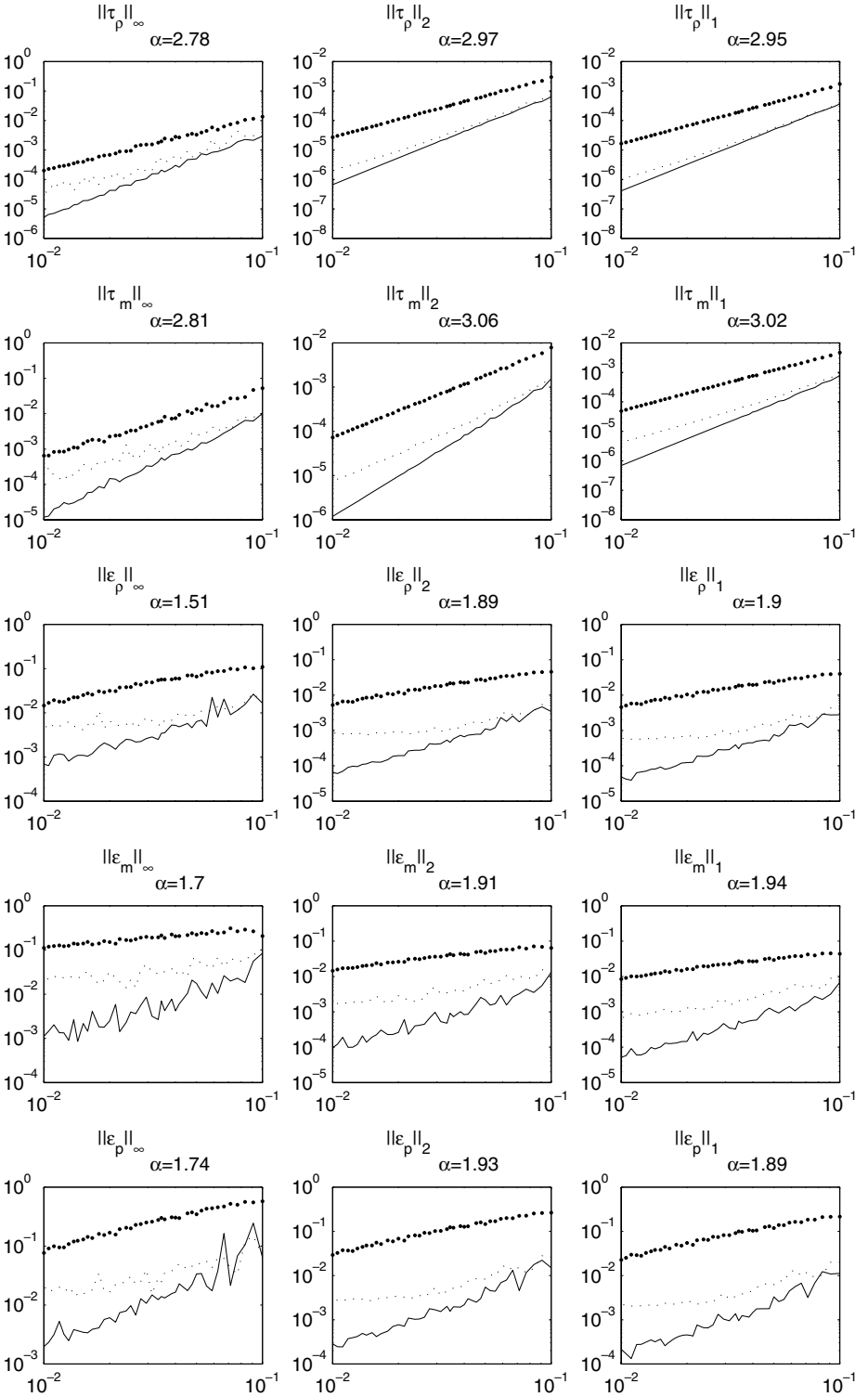

Fig. 8. Ringleb flow, distorted grids. The errors of solutions versus the mesh parameter. "•": first-order scheme; "–": vertex-based reconstruction of the density, cell-based reconstruction of the pressure, no limiter; "·": vertex-based reconstruction of the density, cell-based reconstruction of the pressure, unrelaxed Venkatakrishnan limiter. The estimate of the rate of convergence α refers to the results marked "–".

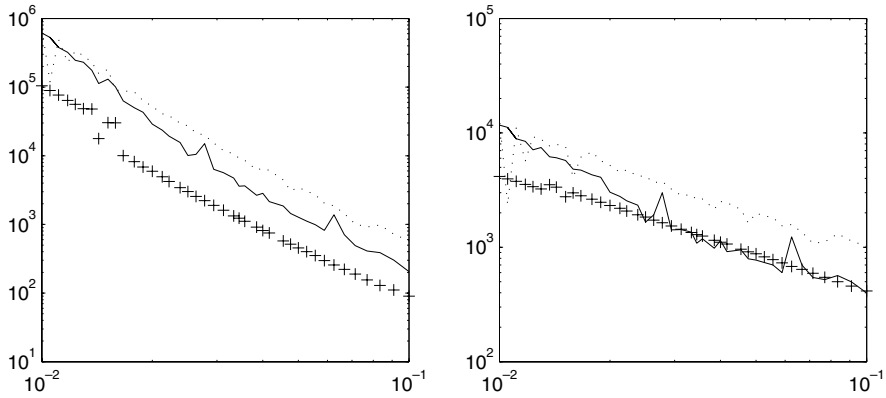

Fig. 9. Ringleb flow, distorted grids; total computation time in seconds (left) and the number of time steps needed to reach the stationary solution (right) versus the mesh parameter. "+": first-order scheme; "–": vertex-based reconstruction of the density, cell-based reconstruction of the pressure, no limiter; "·": vertex-based reconstruction of the density, cell-based reconstruction of the pressure, unrelaxed Venkatakrishnan limiter.

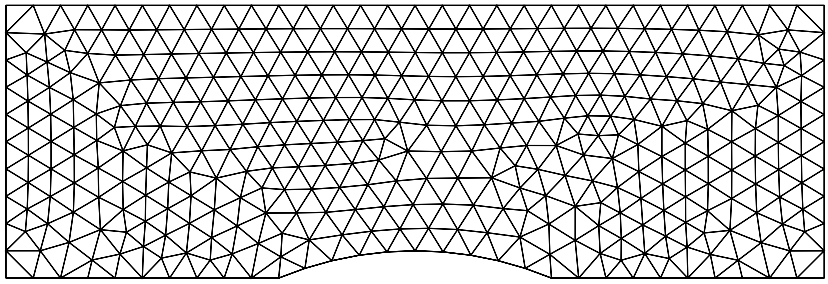

Fig. 10. Channel with a bump, mesh used in the incompressible flow case.

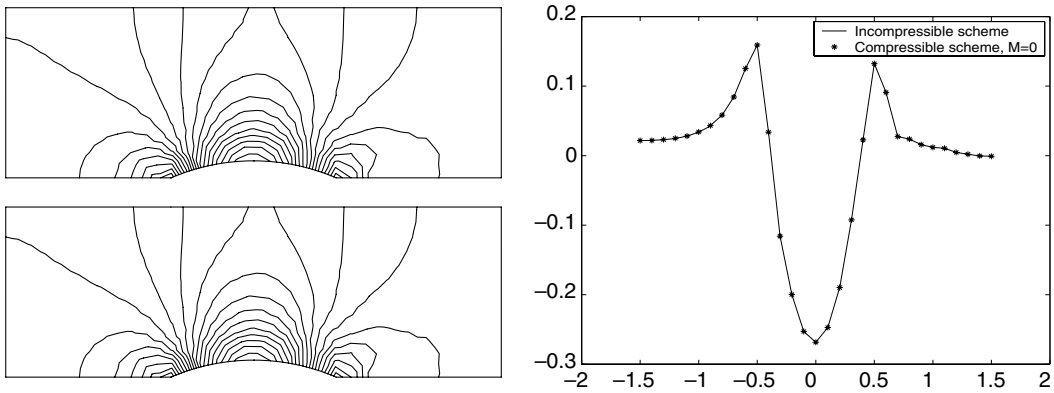

Fig. 11. Channel with a bump, incompressible flow case. Pressure isolines obtained with the incompressible flow (left upper) and the Mach-uniform (left lower) method, and the pressure along the lower wall.

The Mach distribution along the airfoil is shown in Fig. 16 on the right, in comparison with the AGARD benchmark results published in [25]. Parameter n is the number of grid points along the upper surface of the airfoil. The weak shock on the lower surface of the airfoil is particularly hard to capture with a highly dissipative scheme. The first-order scheme does not see it. But the second-order scheme gives an indication of this shock even for a twice less fine grid than the one shown in Fig. 16. In this example we used time step

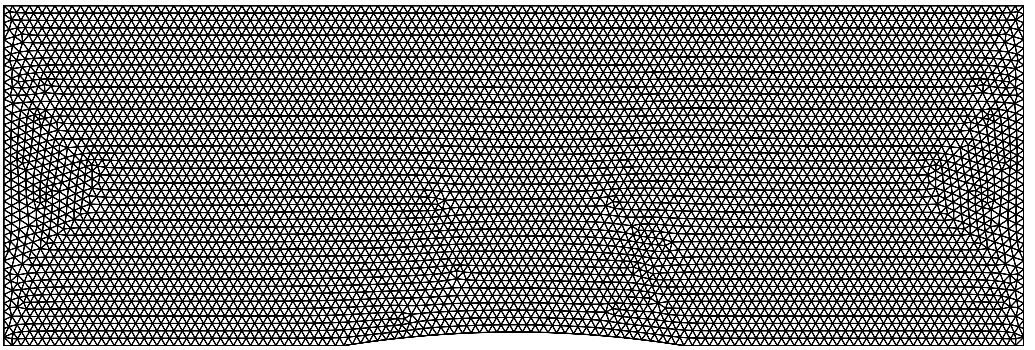

Fig. 12. Channel with a bump, mesh used in the supersonic flow case.

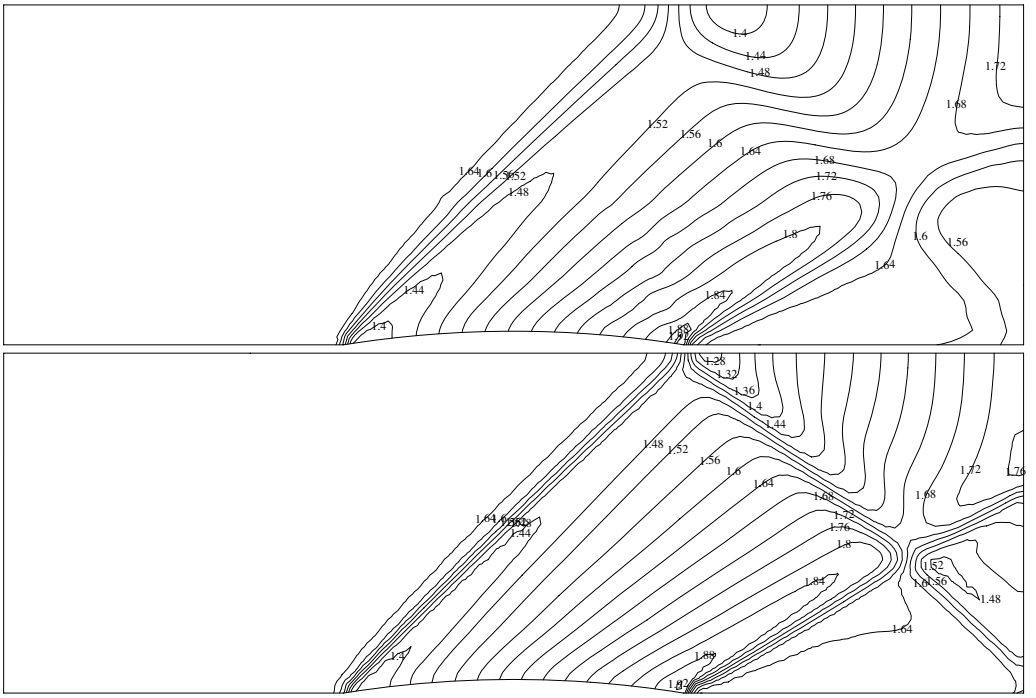

Fig. 13. Channel with a bump, supersonic flow case. Mach number isolines. First-order scheme (top) and the scheme with linear reconstruction (bottom). Vertex-based reconstruction was applied in the density and in the momentum equation, and the cell based reconstruction was used in the energy equation.

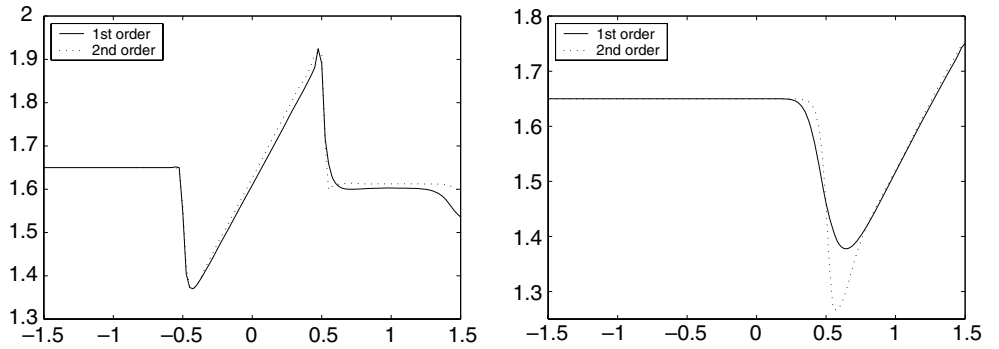

Fig. 14. Channel with a bump, supersonic flow case. Mach number distribution along the lower (left) and the upper wall (right).

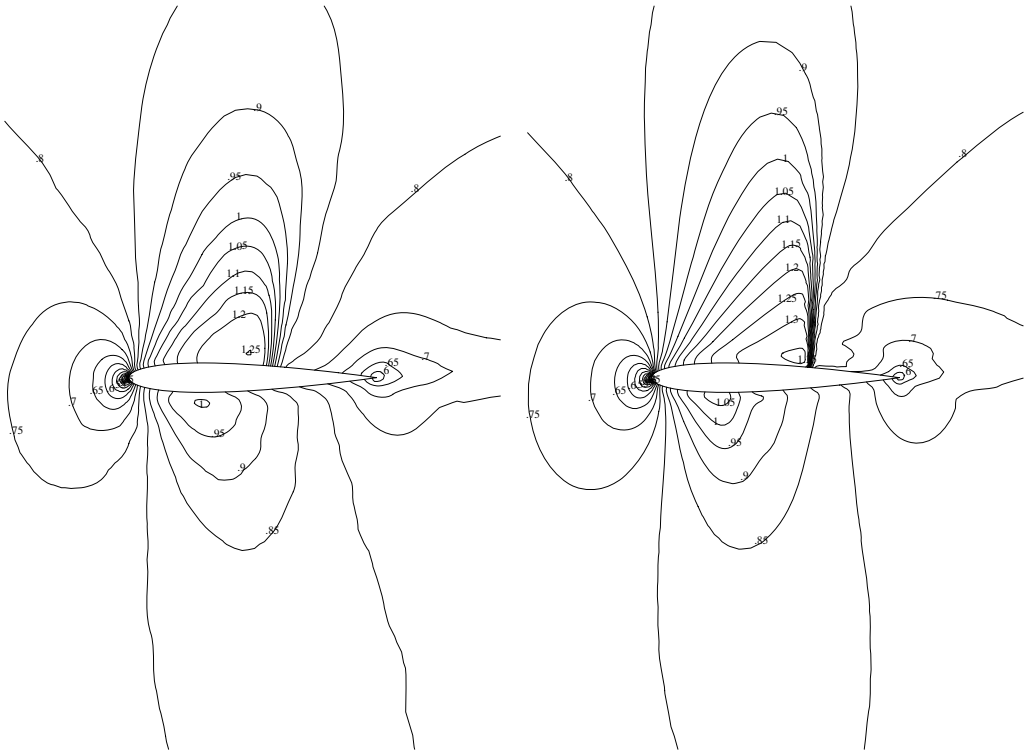

Fig. 15. NACA 0012 airfoil, the Mach isolines obtained with the first-order scheme (left) and with the superlinear scheme (right).

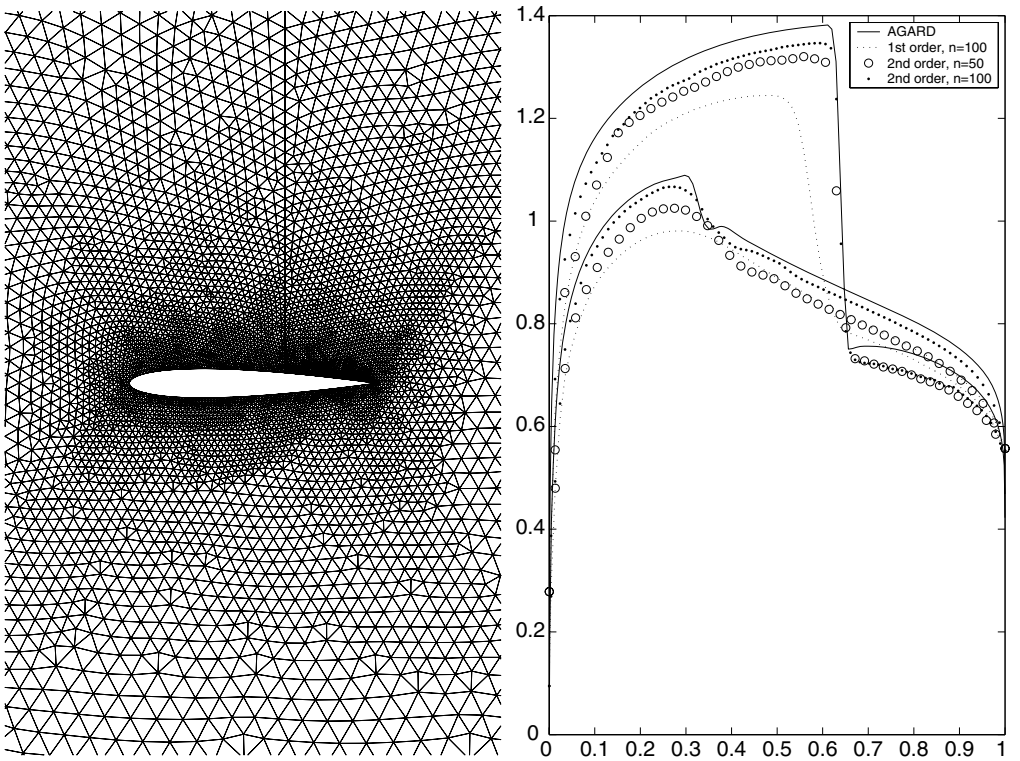

Fig. 16. NACA 0012 airfoil, the finest mesh used, and the Mach distribution along the airfoil.

$\Delta t = 0.0003$. The scheme with linear reconstruction took twice more time steps and 4.3 times more computational time to finish, and it used 50% more memory.

8. Conservation properties

As shown in [26], since the momentum field on an unstructured staggered grid consists of incomparable normal components, it cannot be explicitly shown that the discretization of the momentum equation is conservative. Hence, the Lax–Wendroff theorem does not apply. Nevertheless, as shown in by numerical experiments [26], numerical solutions satisfy the Rankine–Hugoniot conditions. Our scheme conserves mass locally, which is a direct consequence of our choice of the control volumes. The kinetic energy and the momentum are not conserved as they are in [5].

9. Conclusion

A novel Mach-uniform staggered unstructured scheme for solving Euler equations that uses linear reconstruction has been presented. It has been demonstrated that this scheme is second-order accurate for smooth solutions if the cell-based reconstruction is used to evaluate the convected quantities in the Mach-uniform pressure correction equation, and the node-based reconstructions are used in the density and the momentum equation. The estimate of the accuracy was based on the global truncation error and not on the local truncation error or on the order of the reconstruction employed.

The computational time needed to reach the steady state solution with the second-order scheme is comparable with the time needed to reach the steady state solution by the first-order scheme if the solution is smooth, and it is up to five times larger in the discontinuous case. The second-order scheme uses 50% more memory.

In the case of a discontinuous solution, the shock resolution is better and small oscillations that occur are comparable with those obtained with the first-order scheme.

References

[1] C. Hall, J. Cavendish, W. Frey, The dual variable method for solving fluid flow difference equations on Delaunay triangulations, Comput. Fluids 20 (1991) 145–164.
[2] C. Hall, T. Porsching, P. Hu, Covolume-dual variable method for thermally expandable flow on unstructured triangular grids, Int. J. Comput. Fluid Dyn. 2 (1994) 111–139.
[3] R. Nicolaides, T. Porsching, C. Hall, Covolume methods in computational fluid dynamics, in: M. Hafez, K. Oshima (Eds.), Computational Fluid Dynamics Review 1995, Wiley, Chichester, 1995, pp. 279–299.
[4] R. Nicolaides, The covolume approach to computing incompressible flows, in: M. Gunzburger, R. Nicolaides (Eds.), Incompressible Computational Fluid Dynamics, Cambridge University Press, Cambridge, UK, 1993, pp. 295–333.
[5] B. Perot, Conservation properties of unstructured staggered mesh schemes, J. Comput. Phys. 159 (2000) 58–89.
[6] S. Rida, F. McKenty, F. Meng, M. Reggio, A staggered control volume scheme for unstructured triangular grids, Int. J. Numer. Meth. Fluids 25 (1997) 697–717.
[7] I. Wenneker, A. Segal, P. Wesseling, A Mach-uniform unstructured staggered grid method, Int. J. Numer. Meth. Fluids 40 (2002) 1209–1235.
[8] D. Vidović, A. Segal, P. Wesseling, A superlinearly convergent finite volume method for the incompressible Navier–Stokes equations on staggered unstructured grids, J. Comput. Phys. 198 (2004) 159–177.
[9] F. Harlow, A. Amsden, Numerical calculation of almost incompressible flows, J. Comput. Phys. 3 (1968) 80–93.
[10] F. Harlow, A. Amsden, A numerical fluid dynamics calculation method for all flow speeds, J. Comput. Phys. 8 (1971) 197–213.
[11] D. van der Heul, C. Vuik, P. Wesseling, A conservative pressure correction method for compressible flow at all speeds, Int. J. Numer. Meth. Fluids 40 (2002) 521–529.
[12] H. Bijl, Computation of flow at all speeds with a staggered scheme, Ph.D. thesis, Delft University of Technology, The Netherlands, February 1999.
[13] P. Wesseling, Principles of Computational Fluid Dynamics, Springer, Heidelberg, 2001.
[14] P. van Beek, R. van Nooyen, P. Wesseling, Accurate discretization on non-uniform curvilinear staggered grids, J. Comput. Phys. 117 (1995) 364–367.
[15] S. Godunov, Finite difference method for numerical computation of discontinuous solutions of the equations of fluid dynamics, Mat. Sbornik 47 (1959) 271–306 (in Russian).
[16] T.J. Barth, Aspects of unstructured grids and finite-volume solvers for the Euler and Navier–Stokes equations, in: 25th Computational Fluid Dynamics Lecture Series, Von Karman Institute, 1994.
[17] T. Barth, D. Jespersen, The design and application of upwind schemes on unstructured meshes, AIAA Paper 89-0366, 1989.

[18] I. Lepot, A parallel high-order implicit finite volume method for three-dimensional inviscid compressible flows on deforming unstructured meshes, Ph.D. thesis, University of Liege, Belgium, January 2004.
[19] T. Barth, Recent developments in high order k-exact reconstruction on unstructured meshes, AIAA Paper 93-0668, 1993.
[20] V. Venkatakrishnan, Perspective on unstructured grid flow solvers, AIAA J. 34 (1996) 533–547.
[21] M. Delanaye, Polynomial reconstruction finite volume schemes for the compressible Euler and Navier–Stokes equations on unstructured adaptive grids, Ph.D. thesis, University of Liege, Belgium, September 1996.
[22] F. Ringleb, Exakte Lösungen der Differentialgleichungen einer adiabatischen Gasströmung, ZAMM 20 (1940) 185–198.
[23] G. Chiocchia, Exact solutions to transonic and supersonic flows, AGARDograph No. 211, AGARD, Neuilly-sur-Seine, France, 1985.
[24] S. Eidelman, P. Colella, R. Shreeve, Application of the Godunov method and its second-order extension to cascade flow modelling, AIAA J. 22 (1984) 1609–1615.
[25] H. Yoshihara, P. Sacher, Test cases for inviscid flow field methods, AGARDograph No. 211, AGARD, Neuilly-sur-Seine, France, 1985.
[26] I. Wenneker, A. Segal, P. Wesseling, Conservation properties of a new unstructured staggered scheme, Comput. Fluids 32 (2003) 139–147.