Solution of the Steady Euler Equations by a Multigrid Method

Barry Koren, Stefan Spekreijse
Centre for Mathematics and Computer Science
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

An efficient multigrid method to obtain second-order accurate solutions of the 2D steady Euler equations is described and results are shown. The method is based on a nonlinear multigrid (FAS-) iteration method and on a defect correction principle. Both first- and second-order accurate finite volume upwind discretizations are considered. In the second-order discretization a limiter is used.
The method does not require any tuning of parameters. Flow solutions are presented for a channel and an airfoil. The solutions show good resolution of all flow phenomena and are obtained at low computational cost.

1980 Mathematics Subject Classification: 35L65, 35L67, 65N30, 76G15, 76H05.
Key Words and Phrases: steady Euler equations, multigrid methods, defect correction.
Note: This work was supported in part by the Netherlands Technology Foundation (STW).

1. Introduction

The Euler equations describe compressible inviscid gas flows with rotation. They are derived by considering the laws of conservation of mass, momentum and energy for an inviscid gas. The result is a nonlinear hyperbolic system of conservation laws.

To obtain numerical solutions of the steady Euler equations, the equations are discretized by a finite-volume upwind discretization [9]. Both first- and second-order discretizations are obtained by the projection-evolution approach [13]. In the projection-stage of this approach the discrete values, located in the volume centers, are interpolated to yield continuous distributions in each volume. First-order accuracy is obtained by piecewise constant interpolation, second-order accuracy by piecewise linear interpolation. In case of flows with discontinuities (shock waves or slip lines), the occurrence of spurious non-monotonicity (wiggles) when using a second-order interpolation, is suppressed by the use of a limiter in the interpolation formulae [22]. In this paper we use the Van Albada limiter [1,20]. In the evolution-stage, a Riemann problem is considered for the computation of the flux at each volume wall. To approximately solve each Riemann problem we use the Osher scheme [16].

To obtain solutions of the system of first-order discretized equations, the nonlinear multigrid (FAS-) iteration method is a very efficient solution method [9,10]. To improve the order of accuracy, we make use of a Defect Correction (DeC-) iteration process [2,5]. In each iteration of this process, the second-order discretization is only used for the construction of an appropriate right-hand side for a system of first-order discretized equations. The FAS-iteration method is used to solve this system.

Two test problems are considered. The first problem is a supersonic flow in a channel with a circular bump; $M_{inlet}=1.4$ (flow with shock generation, reflection, crossing and merging). The second problem is the transonic flow around the NACA0012-airfoil at $M_\infty=0.85$, $\alpha=1^0$ (flow with upper surface shock, lower surface shock and tail slip line).

In section 2 a description is given of the first- and second-order discretizations, and in section 3 the solution method is described. In section 4 we discuss the numerical results, and in section 5 some conclusions are listed.

2. Discretization

Consider on an open domain $\Omega\in\mathbb{R}^2$ the 2D steady Euler equations in conservation form and without source terms:

$$\frac{\partial f(q)}{\partial x}+\frac{\partial g(q)}{\partial y}=0\,, \tag{2.1}$$

where $q=(\rho,\rho u,\rho v,E)^T$ is the state vector of conservative variables, and where $f(q)=(\rho u,\rho u^2+p,\rho uv,(E+p)u)^T$ and $g(q)=(\rho v,\rho uv,\rho v^2+p,(E+p)v)^T$ are the flux vectors. The primitive variables of (2.1) are the density ρ, the velocity components u and v, and the pressure p. For a perfect gas, the total energy per unit of volume, E, is related to the primitive variables as $E=p/(\gamma-1)+\frac{1}{2}\rho(u^2+v^2)$ where γ is the ratio of specific heats.

To allow solutions with discontinuities we consider the Euler equations in their integral form. Then the 2D steady Euler equations read

$$\int_{\partial\Omega^*}\{\cos\phi f(q)+\sin\phi g(q)\}ds=0\,,\quad \forall\Omega^*\subset\Omega, \tag{2.2}$$

where $\Omega^*\subset\Omega$ is an arbitrary subregion of Ω, $\partial\Omega^*$ the boundary of Ω^*, and $(\cos\phi,\sin\phi)$ the outward unit normal on $\partial\Omega^*$. A straightforward and simple discretization of (2.2) is obtained by subdividing Ω into disjunct quadrilateral subregions $\Omega_{i,j}$ (the finite volumes) and by requiring that

$$\int_{\partial\Omega_{i,j}}(\cos\phi f(q)+\sin\phi g(q))ds=0 \tag{2.3}$$

for each volume $\Omega_{i,j}$ separately. We restrict ourselves to subdivisions such that only $\Omega_{i,j\pm1}$ and $\Omega_{i\pm1,j}$ are the neighbouring volumes of $\Omega_{i,j}$.

Using the rotational invariance of the Euler equations:

$$\cos\phi f(q)+\sin\phi g(q)=T^{-1}(\phi)f(T(\phi)q), \tag{2.4}$$

where $T(\phi)$ is the rotation matrix

$$T(\phi)=\begin{bmatrix}1&0&0&0\\0&\cos\phi&\sin\phi&0\\0&-\sin\phi&\cos\phi&0\\0&0&0&1\end{bmatrix}, \tag{2.5}$$

(2.3) becomes:

$$\int_{\partial\Omega_{i,j}}T^{-1}(\phi)f(T(\phi)q)ds=0\,. \tag{2.6}$$

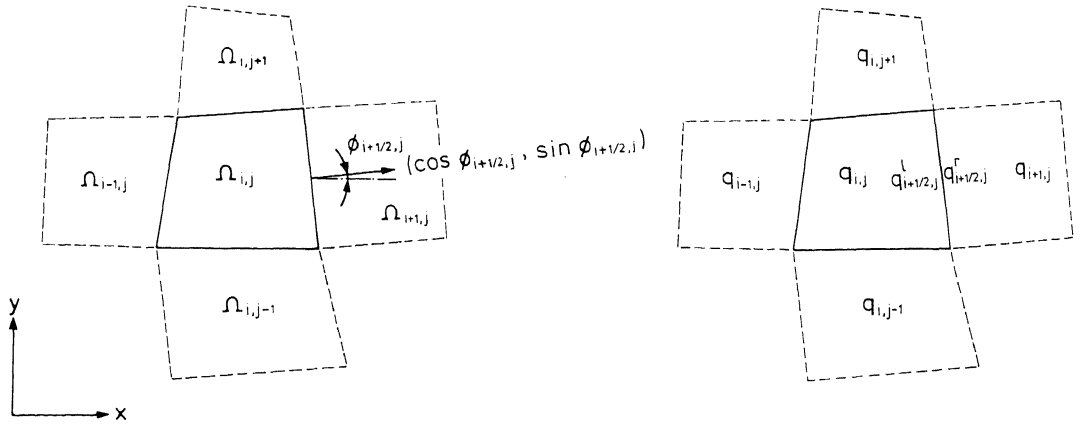
A numerical approximation of this formula is obtained by

$$F_{i,j}:=f_{i+\frac{1}{2},j}+f_{i,j+\frac{1}{2}}-f_{i-\frac{1}{2},j}-f_{i,j-\frac{1}{2}}=0, \tag{2.7}$$

with

$$f_{i+\frac{1}{2},j}=l_{i+\frac{1}{2},j}T^{-1}(\phi_{i+\frac{1}{2},j})f_R(T(\phi_{i+\frac{1}{2},j})q^l_{i+\frac{1}{2},j},T(\phi_{i+\frac{1}{2},j})q^r_{i+\frac{1}{2},j}) \tag{2.8}$$

and similar relations for $f_{i-\frac{1}{2},j}$ and $f_{i,j\pm\frac{1}{2}}$. In (2.8), $l_{i+\frac{1}{2},j}$ is the length of the volume wall



a. Geometry. b. State vectors.

Fig. 2.1: Finite volume $\Omega_{i,j}$.

$\partial\Omega_{i+\frac{1}{2},j} = \Omega_{i,j} \cap \Omega_{i+1,j}$ and $(\cos\phi_{i+\frac{1}{2},j}, \sin\phi_{i+\frac{1}{2},j})$ is the outward unit normal on $\partial\Omega_{i+\frac{1}{2},j}$ (fig. 2.1a). Further, $f_R: \mathbb{R}^4 \times \mathbb{R}^4 \rightarrow \mathbb{R}^4$ is a so-called approximate Riemann-solver and $q^l_{i+\frac{1}{2},j}$ and $q^r_{i+\frac{1}{2},j}$ are state vectors located at the left and right side of volume wall $\partial\Omega_{i+\frac{1}{2},j}$ (fig. 2.1b). The flux vector $f_{i+\frac{1}{2},j}$ represents the transport of mass, momentum and energy per unit of time, across $\partial\Omega_{i+\frac{1}{2},j}$. For a more detailed discussion of (2.7) and (2.8) we refer to [9,19].

The application of an approximate Riemann-solver is the essential part of the evolution-stage, whereas the computation of the states $q^l_{i+\frac{1}{2},j}$ and $q^r_{i+\frac{1}{2},j}$ is the essential part of the projection-stage [13]. As the name suggests an approximate Riemann-solver is used to obtain an approximate solution of the Riemann problem [4,6]. Several approximate Riemann-solvers exist [12,16,18,21]. Here, we use Osher's Riemann-solver because of its consistent treatment of boundary conditions and its continuous differentiability [9,16,17]. For details about an efficient implementation of Osher's approximate Riemann-solver we refer to [9].

Depending on the way the states $q^l_{i+\frac{1}{2},j}$ and $q^r_{i+\frac{1}{2},j}$ are computed, the discretization (2.7) is first- or second-order accurate. First-order accuracy is obtained by taking

$$\begin{aligned} q^l_{i+\frac{1}{2},j} &= q_{i,j}, \text{ and} \\ q^r_{i+\frac{1}{2},j} &= q_{i+1,j}. \end{aligned} \tag{2.9}$$

Second-order accuracy can be obtained by for example the κ-schemes introduced by Van Leer [13]:

$$\begin{aligned} q^l_{i+\frac{1}{2},j} &= q_{i,j} + \frac{1+\kappa}{4}(q_{i+1,j} - q_{i,j}) + \frac{1-\kappa}{4}(q_{i,j} - q_{i-1,j}), \text{ and} \\ q^r_{i+\frac{1}{2},j} &= q_{i+1,j} + \frac{1+\kappa}{4}(q_{i,j} - q_{i+1,j}) + \frac{1-\kappa}{4}(q_{i+1,j} - q_{i+2,j}), \end{aligned} \tag{2.10}$$

with $\kappa \in [-1,1]$. For $\kappa = -1$, $\kappa = 0$, $\kappa = 1/3$ and $\kappa = 1$ we find respectively: the fully one-sided upwind scheme, the Fromm scheme, the upwind biased scheme (third-order accurate for 1D problems) and the central scheme. A disadvantage of these κ-schemes is that near discontinuities, spurious non-monotonicity (wiggles) appears [11]. A way to avoid this is by using a limiter. We modify the κ-schemes by introducing a limiter such that the schemes become monotone and remain second-order accurate. Let $q^{l\,(k)}_{i+\frac{1}{2},j}$ and $q^{r\,(k)}_{i+\frac{1}{2},j}$ be the kth component ($k = 1,2,3,4$) of $q^l_{i+\frac{1}{2},j}$ and $q^r_{i+\frac{1}{2},j}$. We rewrite (2.10) as

$$\begin{aligned} q^{l\,(k)}_{i+\frac{1}{2},j} &= q^{(k)}_{i,j} + \tfrac{1}{2}\psi_\kappa(R^{(k)}_{i,j})(q^{(k)}_{i,j} - q^{(k)}_{i-1,j}), \text{ and} \\ q^{r\,(k)}_{i+\frac{1}{2},j} &= q^{(k)}_{i+1,j} + \tfrac{1}{2}\psi_\kappa(1/R^{(k)}_{i+1,j})(q^{(k)}_{i+1,j} - q^{(k)}_{i+2,j}), \end{aligned} \tag{2.11}$$

where

$$R_{i,j}^{(k)} = \frac{q_{i+1,j}^{(k)} - q_{i,j}^{(k)}}{q_{i,j}^{(k)} - q_{i-1,j}^{(k)}}, \tag{2.12}$$

and where $\psi_\kappa : \mathbb{R} \to \mathbb{R}$ is defined by

$$\psi_\kappa(R) = \frac{1-\kappa}{2} + \frac{1+\kappa}{2} R. \tag{2.13}$$

If we replace $\psi_\kappa(R)$ in (2.11) by $\psi_\kappa^{\lim}(R)$, where $\psi_\kappa^{\lim}(R)$ is defined by

$$\psi_\kappa^{\lim}(R) = \frac{2R}{R^2+1} \psi_\kappa(R), \tag{2.14}$$

then (2.11) results in a monotone and yet second-order accurate scheme [20]. The function $\psi_\kappa^{\lim} : \mathbb{R} \to \mathbb{R}$ is called the limiter. The choice $\kappa = 0$ corresponds with the Van Albada limiter [1,20]. An advantage of the Van Albada limiter is that in the neighbourhood of discontinuities the scheme resembles the fully one-sided upwind scheme, which is a natural scheme in such regions. For all flow solutions presented in this paper we used $\psi_0^{\lim}(R)$ although $\psi_{1/3}^{\lim}(R)$ seems a reasonable choice as well.

In case $\Omega_{i,j}$ is a boundary volume, so that for example $\partial\Omega_{i+\frac{1}{2},j}$ is part of the domain boundary, no limiter can be used to compute $q_{i+\frac{1}{2},j}^l$ and $q_{i-\frac{1}{2},j}^r$. In this case we use a simple linear interpolation, i.e.

$$\begin{aligned} q_{i+\frac{1}{2},j}^l &= q_{i,j} + \tfrac{1}{2}(q_{i,j} - q_{i-1,j}), \text{and} \\ q_{i-\frac{1}{2},j}^r &= q_{i,j} - \tfrac{1}{2}(q_{i,j} - q_{i-1,j}). \end{aligned} \tag{2.15}$$

The boundary conditions, together with the state $q_{i+1/2,j}^l$, are used to compute the state $q_{i+\frac{1}{2},j}^r$. This computation is done by considering the Riemann boundary value problem [9,17]. The flux $f_{i+\frac{1}{2},j}$ at $\partial\Omega_{i+\frac{1}{2},j}$ is computed by (2.8).

3. Solution method

The method to solve the system of nonlinear discretized equations is based on a multigrid technique. For readers unfamiliar with multigrid techniques we refer to [3,5].

Let

$$F_h^1(q_h) = r_h, \tag{3.1}$$

and

$$F_h^2(q_h) = r_h \tag{3.2}$$

be first- and second-order accurate finite-volume upwind discretizations of the 2D steady Euler equations with source term r. Hence, $(F_h^1(q_h))_{i,j} = F_{i,j}$ is defined by (2.7), (2.8) and (2.9), and $(F_h^2(q_h))_{i,j} = F_{i,j}$ is defined by (2.7), (2.8), (2.11) and (2.12) with $\psi_\kappa(R) = \psi_0^{\lim}(R)$ (the Van Albada limiter). Although in general $r = 0$, we prefer to describe the solution method for systems with an arbitrary right-hand side. The subscript h denotes the meshwidth. To apply multigrid we construct a nested set of grids, such that each volume in a grid is the union of 4 volumes in the next finer grid, in the obvious way. Let Ω_{h_i} with $h_1 > h_2 > \cdots > h_l = h$ be a sequence of such nested grids. So Ω_{h_1} and Ω_{h_l} are respectively the coarsest and the finest grid.

The solution method for (3.2) can be divided into three successive stages. The first stage is the Full Multigrid (FMG-) method, which is used to find a good initial approximation of (3.1). The second stage is a nonlinear multigrid (FAS-) iteration method, which is used to find a better approximate solution of (3.1). The first iterand is the solution obtained by the FMG-method. The FAS-iteration method is a very efficient solution method for (3.1) [9,10]. In general, for a single FAS-iteration, the reduction factor of the first-order residual lies in the range 0.1-0.5. Therefore, just a few FAS-iterations are sufficient to drive the first-order residual to machine-zero. The third and last stage is a Defect Correction (DeC-) iteration process, which is used to find an approximate solution of (3.2). The first iterand of this process is obtained from the second stage. We will now discuss these stages more fully.

Stage I : The Full Multigrid (FMG-) method.
Let

$$F^1_{h_i}(q_{h_i}) = r_{h_i} \tag{3.3}$$

be the first-order discretization on Ω_{h_i}, $i = 1,2,...,l$. The FMG-method (or nested iteration) starts with a crude initial estimate of q_{h_1}; the solution on the coarsest grid. To obtain an initial estimate on the finer grid $\Omega_{h_{i+1}}$, first the solution on the next coarser grid Ω_{h_i} is improved by a single FAS-iteration (stage II). Hereafter this improved approximation is interpolated to the finer grid $\Omega_{h_{i+1}}$. These steps are repeated until the highest level has been reached. The interpolation used to obtain the first guess on a next finer grid is a bilinear interpolation. For this purpose the grid Ω_{h_i} is subdivided into disjunct sets of 2×2 volumes. The four states corresponding with each set are interpolated in a bilinear way, and since each volume of Ω_{h_i} overlaps 2×2 finer grid volumes of $\Omega_{h_{i+1}}$, 4×4 new states are obtained on $\Omega_{h_{i+1}}$.

Stage II: The nonlinear multigrid (FAS-) iteration method.
To find a better approximation to (3.1) we apply the FAS-iteration method on the finest grid (Ω_{h_l}). One FAS-iteration on a general grid Ω_{h_i} is recursively defined by the following steps:

(0) Start with an approximate solution of q_{h_i}.
(1) Improve q_{h_i} by application of p (pre-) relaxation iterations to $F^1_{h_i}(q_{h_i}) = r_{h_i}$.
(2) Compute the defect $d_{h_i} := r_{h_i} - F^1_h(q_{h_i})$.
(3) Find an approximation of $q_{h_{i-1}}$ on the next coarser grid $\Omega_{h_{i-1}}$. Either use $q_{h_{i-1}} := \hat{I}^{h_{i-1}}_{h_i} q_{h_i}$, where $\hat{I}^{h_{i-1}}_{h_i}$ is a restriction operator, or use the last obtained approximation $q_{h_{i-1}}$.
(4) Compute $r_{h_{i-1}} := F^1_{h_{i-1}}(q_{h_{i-1}}) + I^{h_{i-1}}_{h_i} d_{h_i}$ where $I^{h_{i-1}}_{h_i}$ is another restriction operator.
(5) Approximate the solution of $F^1_{h_{i-1}}(q_{h_{i-1}}) = r_{h_{i-1}}$ by σ FAS- iterations on $\Omega_{h_{i-1}}$. The result is called $\tilde{q}_{h_{i-1}}$. ($\sigma = 1$ results in a V-cycle and $\sigma = 2$ in a W-cycle.)
(6) Correct the current solution by $q_{h_i} := q_{h_i} + I^{h_i}_{h_{i-1}}(\tilde{q}_{h_{i-1}} - q_{h_{i-1}})$, where $I^{h_i}_{h_{i-1}}$ is a prolongation operator.
(7) Improve q_{h_i} by application of q (post-) relaxation iterations to $F^1_{h_i}(q_{h_i}) = r_{h_i}$.

The steps (2) - (6) are called the coarse-grid correction. These steps are skipped on the coarsest grid.

In order to complete the description of a FAS-iteration we have to discuss: (i) the choice of the transfer operators $I^{h_i}_{h_{i-1}}$, $I^{h_{i-1}}_{h_i}$ and $\hat{I}^{h_{i-1}}_{h_i}$, (ii) the relaxation method, and (iii) the FAS-strategy, i.e. the numbers p, q and σ.

(i) Choice of the operators:
The restriction operators $\hat{I}^{h_{i-1}}_{h_i}$ and $I^{h_{i-1}}_{h_i}$ are defined by

$$(q_{h_{i-1}})_{i,j} = (\hat{I}^{h_{i-1}}_{h_i} q_{h_i})_{i,j} := \tfrac{1}{4}\{(q_{h_i})_{2i,2j} + (q_{h_i})_{2i-1,2j} + (q_{h_i})_{2i,2j-1} + (q_{h_i})_{2i-1,2j-1}\}, \text{and} \tag{3.4}$$

$$(d_{h_{i-1}})_{i,j} = (I^{h_{i-1}}_{h_i} d_{h_i})_{i,j} := (d_{h_i})_{2i,2j} + (d_{h_i})_{2i-1,2j} + (d_{h_i})_{2i,2j-1} + (d_{h_i})_{2i-1,2j-1}. \tag{3.5}$$

The prolongation operator $I^{h_i}_{h_{i-1}}$ is defined by

$$(I^{h_i}_{h_{i-1}} q_{h_{i-1}})_{2i,2j} = (I^{h_i}_{h_{i-1}} q_{h_{i-1}})_{2i-1,2j} = (I^{h_i}_{h_{i-1}} q_{h_{i-1}})_{2i,2j-1} = (I^{h_i}_{h_{i-1}} q_{h_{i-1}})_{2i-1,2j-1} := (q_{h_{i-1}})_{i,j}. \tag{3.6}$$

Note that this prolongation is different from the bilinear interpolation used in FMG. By defining the transfer operators in this way, it can be verified that

$$F^1_{h_{i-1}} = I^{h_{i-1}}_{h_i} F^1_{h_i} I^{h_i}_{h_{i-1}}, \tag{3.7}$$

i.e. the first-order coarse grid discretizations of the steady Euler equations are Galerkin approximations of the fine grid discretizations. This is a very important property because it implies that the coarse grid correction efficiently reduces the smooth component in the residual.

(ii) The relaxation method:
We use Collective Symmetric Gauss-Seidel (CSGS-) relaxation. Collective means that the four vari-

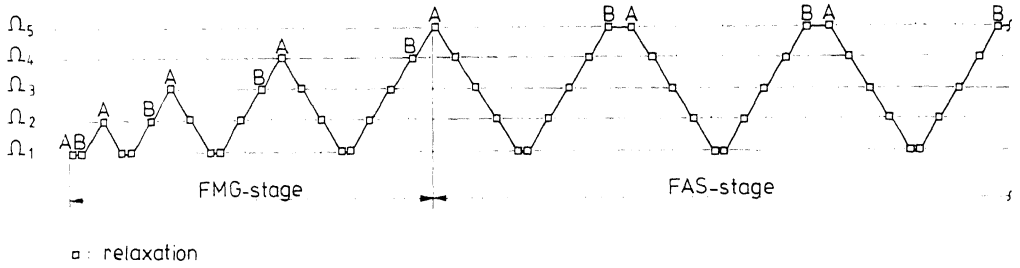


Fig. 3.1: Complete multigrid solution process for obtaining a first-order accurate solution (5 levels).

ables corresponding to a single volume are relaxed simultaneously. At each volume visited we solve the four nonlinear equations by Newton's method (local linearization). It appears that a single Newton iteration is sufficient. For details about the local linearization formulae we refer to [9].

(iii) The FAS-strategy:

We use a fixed strategy: $\sigma=1$ and $p=q=1$, i.e. we use V-cycles with one pre- and one post-relaxation.

When the exact solution of (3.1) is desired, more than one FAS-iteration has to be performed. In fig. 3.1 we give an illustration of the complete solution process for (3.1). It is supposed that there are 5 nested grids ($l=5$). Between two succeeding points A,B we have a single FAS-iteration (V-cycle). Between two succeeding points B,A in the FMG-stage, we have the bilinear prolongation.

Stage III : The Defect Correction (DeC-) iteration method.

For an introduction to the defect correction approach we refer to [2,5]. We approximate the solution of (3.2) with the DeC-iteration process:

$$F_h^1(q_h^{(n+1)}) = F_h^1(q_h^{(n)}) + (r_h - F_h^2(q_h^{(n)})) \, , \; n = 0,1,2,\ldots \, , \tag{3.8}$$

where $q_h^{(0)}$ is the solution obtained in stage II with only a single FAS-iteration. It is clear that the fixed point of this iteration process is the solution of (3.2). In fact it is not really necessary to iterate until convergence. For smooth solutions a single DeC-iteration is sufficient to obtain second-order accuracy [7]. For solutions with discontinuities experience shows that already a few DeC-iterations significantly improve the accuracy of the solution [11]. When more DeC-iterations are performed, the iterand $q_h^{(n)}$ does not always converge to the solution of (3.2) (See for example the channel flow problem in section 4.) But even in those cases, a significant improvement of the accuracy of the solution is observed.

For each DeC-iteration we have to solve a first-order system with an appropriate right-hand side. It appeared that it is inefficient to solve this system very accurately. Application of a single FAS-iteration to approximate $q_h^{(n+1)}$ in (3.8) usually is the most efficient strategy [7,11].

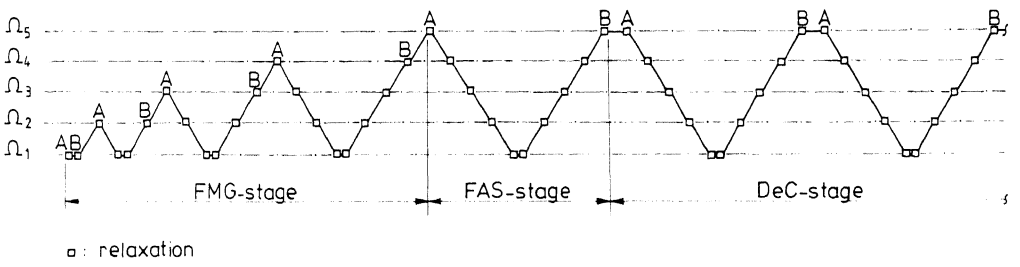


Fig. 3.2: Complete multigrid solution process for obtaining a second-order accurate solution (5 levels).

In fig. 3.2. we give an illustration of the complete process for the approximate solution of (3.2). Suppose there are 5 nested grids ($l = 5$). Between two succeeding points A,B we have one FAS-iteration (V-cycle). Between two succeeding points B,A we have a bilinear interpolation in the FMG-stage, and an appropriate right-hand side computation in the DeC-stage.

4. Results

To show that the method is feasible for a good and efficient computation of typical Euler flows, we consider two standard Euler test cases: (i) a supersonic flow in a channel with a circular arc bump, and (ii) a transonic flow around the NACA0012-airfoil.

The channel:

The geometry of the channel and the grid (96×32) are shown in fig. 4.1. The bump has a thickness ratio of 4%. For the multigrid algorithm we use 4 coarser grids. At the inflow boundary ($x=-1$) the Mach number has been prescribed: $M_{inlet}=1.4$. For results obtained by others, we refer to [15,24].

For this problem we compare the first-order solution with a second-order solution. The first-order solution is obtained with the FAS-iteration process. Fig. 4.2a shows the convergence history of this process. The residual is computed as $\Sigma_{i,j}|F_h^1(q_h^{(n)})|_{i,j}$ (L_1-norm). A second-order accurate solution is obtained with the DeC-iteration process. The convergence history is shown in fig. 4.2b. Here, the residual is computed as $\Sigma_{i,j}|F_h^2(q_h^{(n)})|_{i,j}$. A slow convergence behaviour is observed, but, as mentioned before, the DeC-iteration process is not used to obtain the solution of $F_h^2(q_h)=0$, but to improve the accuracy of the first-order solution.

Fig. 4.3a,b show the iso-Mach lines of respectively the first- and second-order solution. In both solutions, the oblique shock generated at the leading and trailing edge of the bump is clearly visible. In the first-order solution the shocks are severely spread. The reflection of the leading edge shock at the upper wall is hardly visible. The second-order solution, on the contrary, shows very sharp shocks. The reflection of the leading edge shock at the upper wall, its crossing with the trailing edge shock, its further reflection at the lower wall and finally its merging with the trailing edge shock, are all clearly visible.

Fig. 4.4 shows the Mach number distributions along the lower surface of the channel. Downstream of the bump, the large qualitative difference between the first- and second-order solution is observed once more.

Finally, fig. 4.5 shows the entropy distribution $s/s_{inlet}-1$ with $s=p\rho^{-\gamma}$, for the first- and second-order solution, along the lower channel wall. The first-order solution shows a spurious entropy generation along the entire bump. The second-order solution has no such entropy generation, but shows some spurious non-monotonicity. The latter is caused by the fact that no limiter can be used near boundaries (see 2.15).

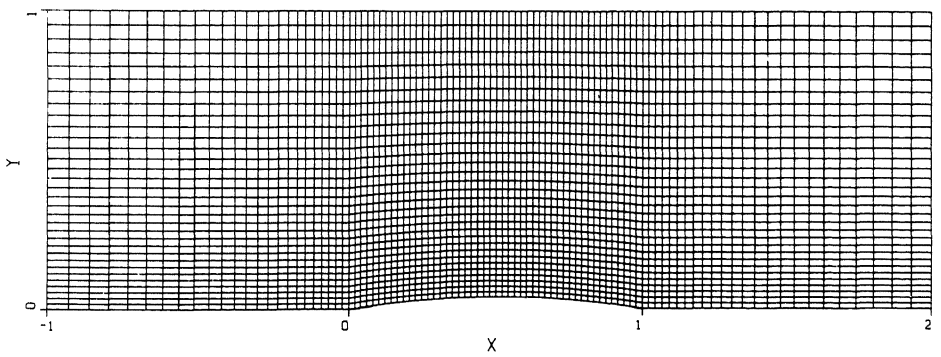
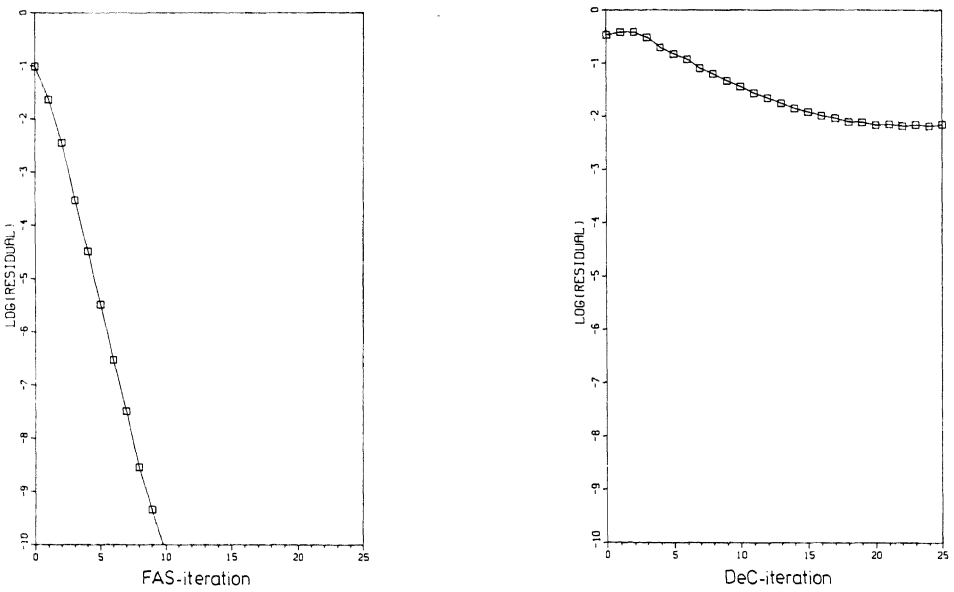
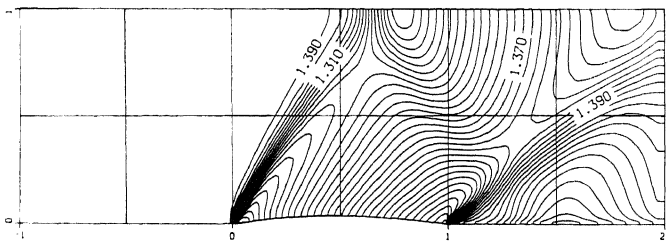


Fig. 4.1: 96×32-grid channel.

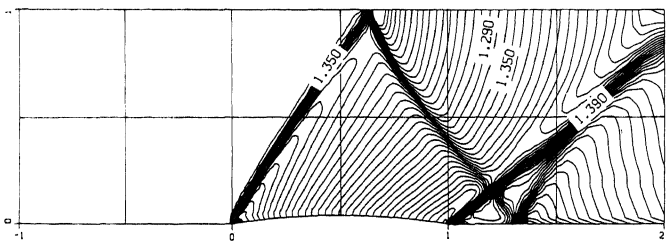


a. First-order. b. Second-order.

Fig. 4.2: Convergence histories; channel flow.

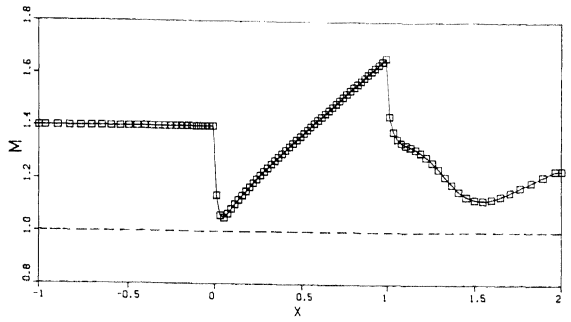


a. First-order.

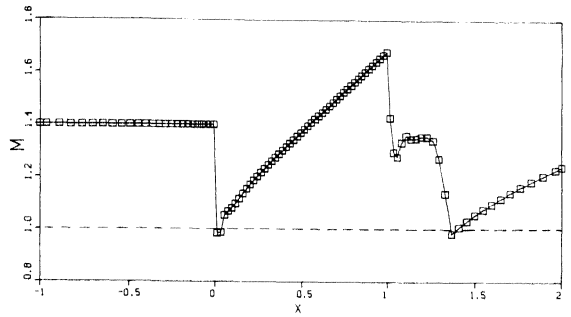


b. Second-order.

Fig. 4.3: Iso-Mach lines.



a. First-order.



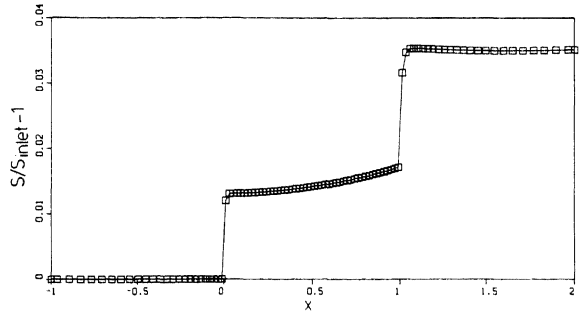
b. Second-order.

Fig. 4.4: Mach number distributions along lower channel wall.

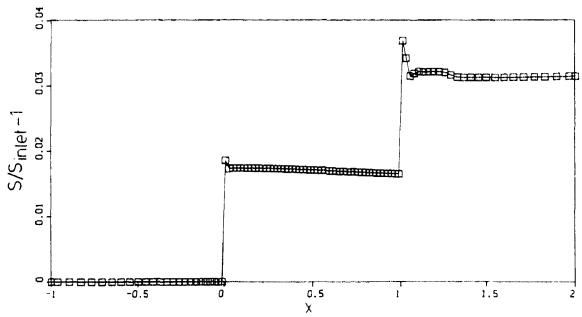
The NACA0012-airfoil:
As standard Euler test case for the NACA0012-airfoil we consider: $M_\infty = 0.85$, $\alpha = 1^0$. (M_∞ denotes the Mach number at infinity and α the airfoil's angle of attack.) We compare our results with results from [23]. We use a 128×32 O-type grid with the outer boundary at an approximate distance from the airfoil of 100 chord lengths (fig. 4.6). Following [8,11], we impose unperturbed flow conditions at the outer boundary, although we do not overimpose. For the subsonic outer boundary we impose 3 conditions at the inflow part of that boundary ($u = M_\infty \cos\alpha$, $v = M_\infty \sin\alpha$, $c = 1$), and 1 condition at the outflow part ($u = M_\infty \cos\alpha$). We perform 10 DeC-iterations and use a multigrid algorithm with (again) 4 coarser grids.

The results obtained are presented in fig. 4.7. In fig. 4.7a and 4.7b we present convergence histories. In fig. 4.7a the residual ratio $\Sigma_{i,j}|F_h^2(q_h^{(n)})|_{i,j} / \Sigma_{i,j}|F_h^2(q_h^{(0)})|_{i,j}$ (L_1-norm) is plotted versus n; the number of DeC-iterations. In fig. 4.7b we show the convergence history of the lift and drag force acting on the airfoil. (For a definition of lift, drag and their proper scaling we refer to e.g. [14].) Although the L_1-norm of the residual ratio is decreasing rather slowly, fig. 4.7b shows that a practical convergence of the lift and drag has been obtained after $\sim$7 DeC-iterations. This is typical for DeC-processes [11]. The shaded areas in fig. 4.7b represent the values of lift and drag as presented in [23] by 7 other investigators. As the best reference results from [23] we selected those obtained by Schmidt & Jameson. For the lift and drag they find: $c_l = 0.3472$, $c_d = 0.0557$, whereas we find (after the 10th DeC-iteration): $c_l = 0.3565$, $c_d = 0.0582$.

In fig. 4.7c we show a contour plot of the Mach number distribution and make a comparison with the distribution as obtained by Schmidt & Jameson. Both distributions show a good (i.e. a sharp and monotone) capturing of the two shock waves, and of the slip line leaving the airfoil's tail. Concerning the sharpness of the discontinuities, it should be noticed that Schmidt & Jameson used a 320×64 (!) O-type grid.



a. First-order.

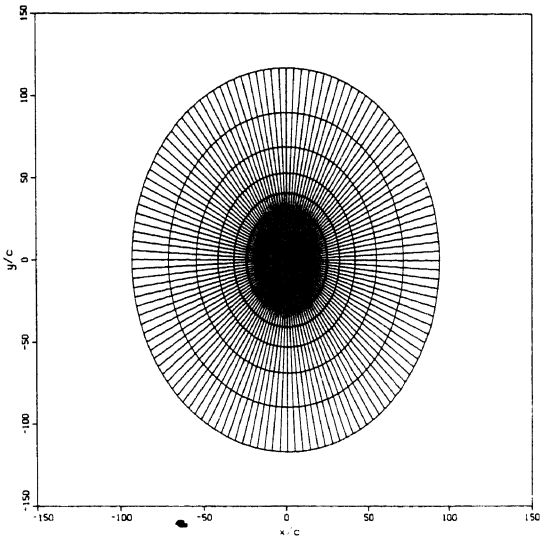


b. Second-order.

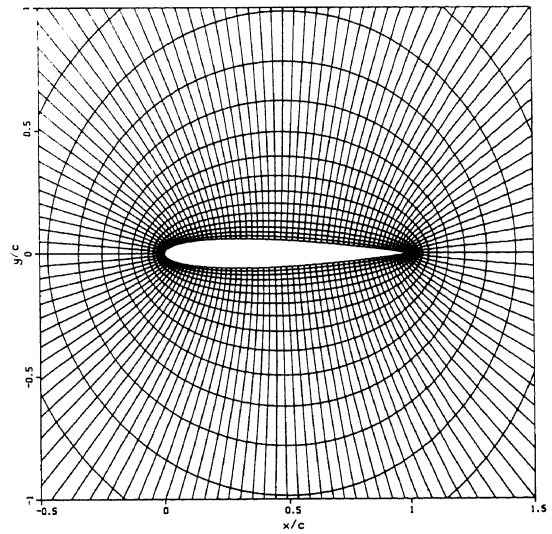
Fig. 4.5: Entropy distributions along lower channel wall.

In fig. 4.7d and 4.7e we show a contour plot of our pressure and entropy distribution. (No reference results are available.) The pressure distribution clearly shows the smoothness of the pressure across the slip line (up to the airfoil's tail). The Kutta-condition is satisfied automatically. The entropy distribution $s/s_\infty - 1$ has a convection of spurious entropy generated at the airfoil's nose of 0.003 only. Even more clear than the Mach number distribution, the entropy distribution shows the good capturing of all three discontinuities. The slight spreading of the slip line in downstream direction is only due to the grid enlargement in this direction.

In [11] it is shown for five different airfoil flows that we need 5 DeC-iterations on an average to drive the lift to within ½% of its final value. (The drag appeared to converge even faster in most cases.) On the single pipe Cyber 205 on which we performed our computation, for a 128×32-grid, 5 DeC-iterations take in scalar mode $\sim$100 sec (i.e. $\sim$5 msec per volume and per iteration). In vector mode it takes $\sim$50 sec. We did not extensively tune our code for use on vector computers since the method brings with it some severe inhibiters for vectorization. However, for large scale computations where all data cannot be kept in core, an advantage of the present method is the small number of iterations required. (For most Euler codes this number is significantly larger.) If all data cannot be kept in core, a small number of iterations results in a small data transport load. Since IO-times rather than CPU-times may be the bottleneck in large scale computations on vector computers, we consider this feature as an extra advantage of the present method.

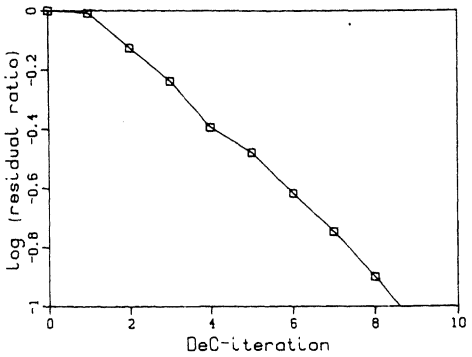


a. In full.

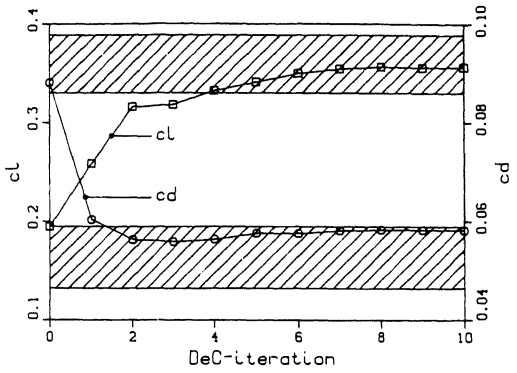


b. In detail.

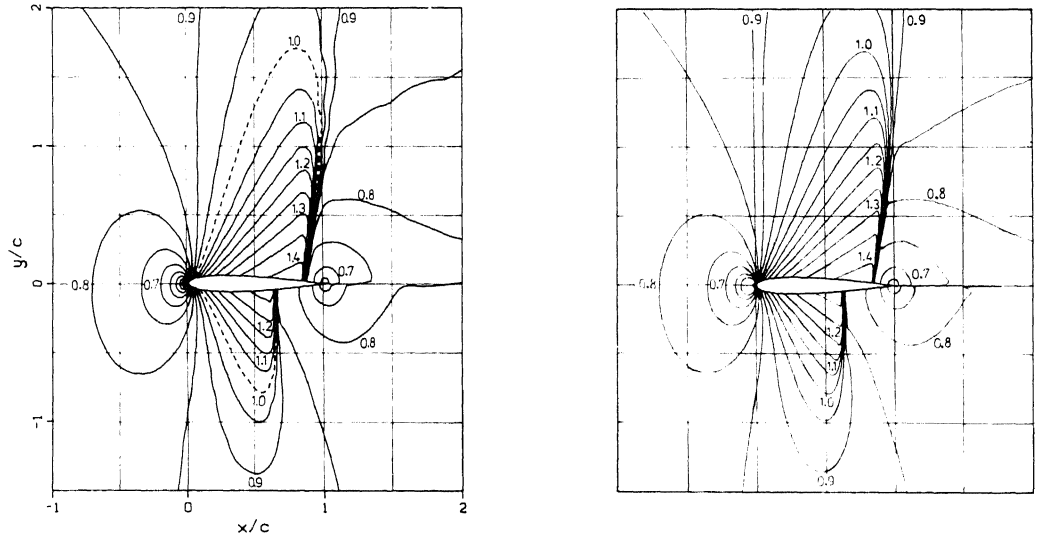
Fig. 4.6: 128×32-grid NACA0012-airfoil.



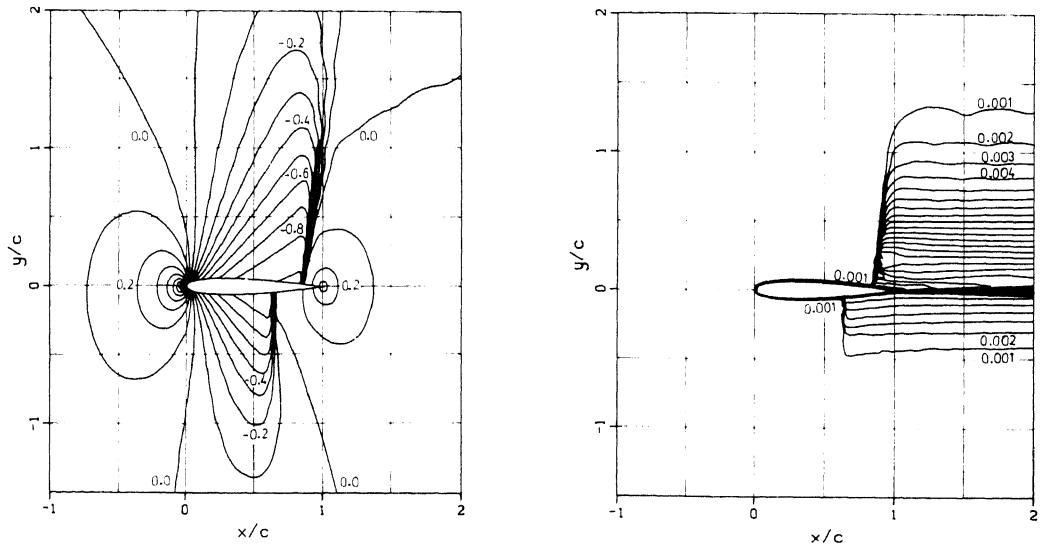
a. Convergence history residual ratio.



b. Convergence history lift and drag coefficient.



c. Mach number distributions; present result (left) and result Schmidt & Jameson (right).



d. Present pressure distribution (c_p).

e. Present entropy distribution ($s/s_\infty - 1$).

Fig. 4.7: Results for NACA0012-airfoil at $M_\infty = 0.85$, $\alpha = 1^0$.

5. Conclusions

For the computation of non-smooth flows with the steady Euler equations, defect correction and non-linear multigrid are found to be very efficient tools. A second-order accurate solution is obtained already after a few DeC-iterations. For each DeC-iteration, a first-order system with an appropriate right-hand side has to be solved approximately. This is done by a FAS-iteration method. It appears that a single FAS-iteration is already sufficient.

The scheme used is a second-order Osher upwind scheme supplied with the Van Albada limiter. The solutions obtained show a good resolution of all flow phenomena and are obtained at low computational cost.

An important property of the present method is that it is completely parameter-free; it needs no tuning of parameters.

Acknowledgements

The authors would like to thank P.W. Hemker and P. Wesseling for giving valuable suggestions, NLR for providing grids, and H. Viviand and AGARD for permitting the use of reference results.

References

1. G.D. van Albada, B. van Leer & W.W. Roberts (1982): *A Comparative Study of Computational Methods in Cosmic Gasdynamics.* Astron. Astrophys. 108, 76-84.
2. K. Böhmer, P.W. Hemker & H.J. Stetter (1984): *The Defect Correction Approach.* Computing, Suppl. 5, 1-32.
3. A. Brandt (1982): *Guide to Multigrid Development.* Lecture Notes in Mathematics 960, 220-312, Springer Verlag, Berlin.
4. S.K. Godunov (1959): *Finite Difference Method for Numerical Computation of Discontinuous Solutions of the Equations of Fluid Dynamics* (in Russian, also Cornell Aeronautical Lab. Transl.). Math. Sbornik 47, 272-306.
5. W. Hackbusch (1985): *Multi-Grid Methods and Applications.* Springer Verlag, Berlin.
6. A. Harten, P. Lax & B. van Leer (1983). *On Upstream Differencing and Godunov-type Schemes for Hyperbolic Conservation Laws.* SIAM Review 25, 35-61.
7. P.W. Hemker (1985): *Defect Correction and Higher-Order Schemes for the Multigrid Solution of the Steady Euler Equations.* Proceedings 2nd European Multigrid Conference, Cologne, 1985. Lecture Notes in Mathematics 1228, 149-165, Springer Verlag, Berlin.
8. P.W. Hemker & B. Koren (1986): *A Non-linear Multigrid Method for the Steady Euler Equations.* Report NM-R8621, Centre for Mathematics and Computer Science, Amsterdam. To appear in Proceedings GAMM-Workshop on The Numerical Simulation of Compressible Euler Flows, Rocquencourt, 1986. Vieweg Verlag Series Notes on Numerical Fluid Mechanics.
9. P.W. Hemker & S.P. Spekreijse (1986): *Multiple Grid and Osher's Scheme for the Efficient Solution of the Steady Euler Equations.* Appl. Num. Math. 2, 475-493.
10. P.W. Hemker & S.P. Spekreijse (1985): *Multigrid Solution of the Steady Euler Equations.* Notes on Numerical Fluid Mechanics, Vol 11, 33-44, Vieweg-Verlag, Braunschweig.
11. B. Koren (1987): *Defect Correction and Multigrid for an Efficient and Accurate Computation of Airfoil Flows.* To appear in J. Comput. Phys.
12. B. van Leer (1982): *Flux-Vector Splitting for the Euler Equations.* Proceedings 8th International Conference on Numerical Methods in Fluid Dynamics, Aachen, 1982. Lecture Notes in Physics 170, 507-512, Springer Verlag, Berlin.
13. B. van Leer (1985): *Upwind-Difference Methods for Aerodynamic Problems governed by the Euler Equations.* Lectures in Applied Mathematics 22-part 2, 327-336, AMS.

14. H.W. LIEPMANN & A. ROSHKO (1966): *Elements of Gasdynamics.* Wiley.
15. R.H. NI (1982): *A Multiple-Grid Scheme for Solving the Euler Equations.* AIAA-81-1025.
16. S. OSHER & F. SOLOMON (1982): *Upwind-Difference Schemes for Hyperbolic Systems of Conservation Laws.* Math. Comp. 38, 339-374.
17. S. OSHER & S. CHAKRAVARTHY (1983): *Upwind Schemes and Boundary Conditions with Applications to Euler Equations in General Geometries.* J. Comput. Phys. 50, 447-481.
18. P.L. ROE (1981): *Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes.* J. Comput. Phys. 43, 357-372.
19. S.P. SPEKREIJSE (1985): *Second-Order Accurate Upwind Solutions of the 2D Steady Euler Equations by the Use of a Defect Correction Method.* Proceedings 2nd European Multigrid Conference, Cologne, 1985. Lecture Notes in Mathematics 1228, 285-300, Springer Verlag, Berlin.
20. S.P. SPEKREIJSE (1987): *Multigrid Solution of Monotone Second-Order Discretizations of Hyperbolic Conservation Laws.* Math. Comp. 49, 135-155.
21. J.L. STEGER & R.F. WARMING (1981): *Flux-Vector Splitting of the Inviscid Gas Dynamic Equations with Applications to Finite-Difference Methods.* J. Comput. Phys. 40, 263-293.
22. P.K. SWEBY (1984): *High Resolution Schemes using Flux Limiters for Hyperbolic Conservation Laws.* SIAM J. Num. Anal. 21, 995-1011.
23. H. VIVIAND (1985): *Numerical Solutions of Two-Dimensional Reference Test Cases.* In: Test Cases for Inviscid Flow Field Methods. H. YOSHIHARA, et al. (eds.). AGARD Advisory Report 211.
24. J.Y. YANG (1985): *Numerical Solution of the Two-Dimensional Euler Equations by Second-Order Upwind Difference Schemes.* AIAA-85-0292.