VIII.2. Defect Correction and Nonlinear Multigrid for Steady Euler Equations

P.W. Hemker and B. Koren

1. INTRODUCTION

In this chapter we give a description of a multigrid method developed for the steady Euler and Navier-Stokes equations. The discretization method is based on cell-centered finite volumes. The solution method, that does not use time stepping, is based on nonlinear multigrid iteration (FAS). The method gets many of its good properties by the use of a sequence of first-order discretizations, based on a variant of Osher's approximate Riemann solver. Higher-order accuracy is obtained by defect correction iteration.
In this chapter, the method is described for the Euler equations only. However, it can be used for the Navier-Stokes equations equally well [16,17]. Recent research has shown that the method can be applied for flows ranging from subsonic to hypersonic speeds, with some slight modifications only for the latter regime [18,20].
In order to establish the notations to be used in this chapter, first we give the equations considered. On a two-dimensional domain $\Omega^* \subset \mathbb{R}^2$, the *Euler equations,* describing the physical laws of conservation of mass, momentum and energy, can be written as

$$\frac{\partial q}{\partial t} + \frac{\partial f(q)}{\partial x} + \frac{\partial g(q)}{\partial y} = 0, \tag{1}$$

where

$$q = (\rho, \rho u, \rho v, \rho e)^T, \tag{2a}$$

$$f = (\rho u, \rho u^2 + p, \rho uv, \rho uh)^T, \tag{2b}$$

$$g = (\rho v, \rho vu, \rho v^2 + p, \rho vh)^T. \tag{2c}$$

Here ρ, u, v, e and p denote density, velocity in x- and y-direction, specific energy and pressure, respectively, whereas $h = e + p/\rho$ is the specific enthalpy. For a perfect gas

$$p = (\gamma - 1)\rho(e - \tfrac{1}{2}(u^2 + v^2)), \tag{3}$$

where γ is the ratio of specific heats. The unknown vector $q(t,x,y)$ describes the state of the gas as a function of time and space, and f and g are the convective fluxes in the x- and y-direction, respectively.
Written in the quasi-linear form, the time-dependent Euler equations form a hyperbolic system;

$$\frac{\partial q}{\partial t}+\frac{df}{dq}\frac{\partial q}{\partial x}+\frac{dg}{dq}\frac{\partial q}{\partial y}=0, \tag{4}$$

i.e. the matrix

$$k_1A+k_2B=k_1\frac{df}{dq}+k_2\frac{dg}{dq} \tag{5}$$

has real eigenvalues for all directions (k_1,k_2). These eigenvalues are: k_1u+k_2v (a double eigenvalue) and $k_1u+k_2v\pm c$, where $c=\sqrt{\gamma p/\rho}$ is the local speed of sound. The sign of the eigenvalues determines the direction in which the information about the solution is carried along the line with direction (k_1,k_2), as time develops.
Because of the nonlinearity, solutions of the Euler equations may develop discontinuities, even if the initial flow $(t=t_0)$ is smooth. To allow discontinuous solutions, following Lax [21], eq. (4) is rewritten in its integral form

$$\frac{\partial}{\partial t}\int_{\Omega} q\,dx\,dy+\oint_{\partial\Omega}(fn_x+gn_y)\,ds=0, \qquad \forall\Omega\subset\Omega^*, \tag{6}$$

where $\partial\Omega$ is the boundary of Ω and where (n_x,n_y) is the outward unit normal at $\partial\Omega$. The form (6) of eq. (4) clearly shows the character of the system of conservation laws: the increase of q in Ω can be caused only by the inflow of q over $\partial\Omega$. In symbolic form, (6) is rewritten as

$$\frac{\partial}{\partial t}\int_{\Omega} q\,dx\,dy+N(q)=0. \tag{7}$$

The solution of the Euler equations in the weak form (7), is known to be non-unique. A unique and physically realistic solution (which is the limit of a solution with vanishing viscosity) is obtained by imposing the entropy condition.
Because we are mainly interested in *steady* flow computations, we can concentrate on a solution method for the steady Euler equations:

$$N(q)=0. \tag{8}$$

Notice that N can be seen as a nonlinear mapping between two Banach spaces, $N:X\to Y$.

2. A MULTIGRID APPROACH FOR THE FIRST-ORDER DISCRETIZATION

2.1 The first-order finite volume discretization

To discretize eq. (8), the domain Ω^* is divided into disjunct quadrilateral cells $\Omega_{i,j}$, in a regular fashion such that

$$\bar{\Omega}^*=\bigcup_{i,j}\bar{\Omega}_{i,j}, \tag{9}$$

where $\bar{\Omega}_{i,j}$ is the closure of $\Omega_{i,j}$. We restrict ourselves to divisions where each cell has (at most) four neighbors, such that $\Omega_{i\pm1,j}$ and $\Omega_{i,j\pm1}$ are the neighboring cells of

$\Omega_{i,j}$. Further we denote the neighbors of $\Omega_{i,j}$ by $\Omega_{i,j,k}$ $(k=N,S,E,W)$, and a common wall by $\Gamma_{i,j,k}=\bar{\Omega}_{i,j}\cap\bar{\Omega}_{i,j,k}$. The boundary of $\Omega_{i,j}$ is given by $\partial\Omega_{i,j}=\bigcup_{k=N,S,E,W}\Gamma_{i,j,k}$. The restriction to this kind of regular geometry is not necessary for the discretization method but leads to a simple data structure when the method is implemented. Evaluating eq. (6) over $\Omega_{i,j}$, we obtain

$$A_{i,j}\frac{\partial q_{i,j}}{\partial t}+\sum_k\int_{\Gamma_{i,j,k}}(fn_x+gn_y)ds=0, \tag{10}$$

where $A_{i,j}$ is the area of cell $\Omega_{i,j}$ and where $q_{i,j}$ is the mean value of q over $\Omega_{i,j}$. Further we introduce the notation

$$\int_{\Gamma_{i,j,k}}(fn_x+gn_y)ds=f_{i,j,k}s_{i,j,k},\quad \forall i,j,k, \tag{11}$$

where $s_{i,j,k}$ is the length of $\Gamma_{i,j,k}$ and where $f_{i,j,k}$ is the mean flux across $\Gamma_{i,j,k}$, outward $\Omega_{i,j}$. If $\bar{\Omega}_{i,j}$ and $\bar{\Omega}_{i',j'}$ are neighbors with a common side $(\Gamma_{i,j,k}=\Gamma_{i',j',k'})$, then $f_{i,j,k}=-f_{i',j',k'}$. The space discretization of eq. (6) is done according to the Godunov principle: the state $q(t,x,y)$ is approximated by $q_{i,j}(t)$ for all $\Omega_{i,j}$ and the mean fluxes $f_{i,j,k}$ are approximated from the states in the adjacent cells. For this purpose, a computed flux $f_{i,j,k}(q^k_{i,j},q^k_{i,j,k})$ is introduced to replace $f_{i,j,k}$. Here, $q^k_{i,j}$ and $q^k_{i,j,k}$ are approximations of q at both sides of $\Gamma_{i,j,k}$. Thus we obtain the following semi-discretization of eq. (6):

$$A_{i,j}\frac{\partial q_{i,j}}{\partial t}+\sum_k s_{i,j,k}f_{i,j,k}(q^k_{i,j},q^k_{i,j,k})=0,\quad \forall i,j. \tag{12}$$

For steady flows, this reduces to

$$\sum_k s_{i,j,k}f_{i,j,k}(q^k_{i,j},q^k_{i,j,k})=0,\quad \forall i,j, \tag{13}$$

which we abbreviate as

$$N_h(q_h)=0. \tag{14}$$

Notice that N_h can be seen as a mapping between two discrete Banach spaces, $N_h: X_h\to Y_h$.

If the cell $\Omega_{i,j}$ is adjacent to the boundary of Ω^*, i.e. $\Gamma_{i,j,k}\subset\partial\Omega^*$, then the state $q_{i,j,k}$ is not available in general. In that case $f_{i,j,k}$ is computed from $q_{i,j}$ and the boundary conditions at $\Gamma_{i,j,k}$.

The main difficulty in eq. (13) is the evaluation of $f_{i,j,k}(q^k_{i,j},q^k_{i,j,k})$ for a given $q^k_{i,j}$ and $q^k_{i,j,k}$. One possible approach is to consider the state $q(t,x,y)$ at $t=t_0$ as piecewise constant over each cell separately, to take $q^k_{i,j}=q_{i,j}$ and $q^k_{i,j,k}=q_{i,j,k}$, and to compute the fluxes over the walls as a quasi-one-dimensional problem during a small time interval $(t_0,t_0+\Delta t)$, by approximately solving the Riemann problem for gasdynamics. Approximate Riemann solvers have been proposed by Steger and Warming [34], Van Leer [22], Roe [29], Osher [28], and others. (Notice that by taking $q^k_{i,j}=q_{i,j}$ and $q^k_{i,j,k}=q_{i,j,k}$ the space discretization is first-order accurate.)

The possible irregularity of the mesh is easily dealt with by making use of the invariance of the Euler equations, under rotation of the coordinate system. Let the normal of a skew wall $\Gamma_{i,j,k}$, directed from $\Omega_{i,j}$ to $\Omega_{i,j,k}$, be given by $(n_x, n_y) = (\cos\phi_{i,j,k}, \sin\phi_{i,j,k})$. Then the simple local rotation

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} n_x & n_y \\ -n_y & n_x \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{15}$$

reduces the computation of $f_{i,j,k}(q_{i,j}, q_{i,j,k})$ to

$$f_{i,j,k}(q_{i,j}, q_{i,j,k}) = T_{i,j,k}^{-1} f(T_{i,j,k} q_{i,j}, T_{i,j,k} q_{i,j,k}), \quad \text{or} \tag{16a}$$

$$f_{i,j,k}(q_{i,j}, q_{i,j,k}) = T_{i,j,k}^{-1} f(T_{i,j,k} q_{i,j,k}, T_{i,j,k} q_{i,j}), \tag{16b}$$

where the rotation matrix $T_{i,j,k}$ transforms the velocity components in q to the coordinate system that is associated with the normal to the cell interface. Notice that we have either (16a) or (16b), depending on whether $q_{i,j}$ is at the left or right side of $\Gamma_{i,j,k}$, respectively. The function $f(q^l, q^r)$ is called the *numerical flux function*. We see that the quantities $s_{i,j,k}$ and $\phi_{i,j,k}$ are the only geometrical data about the mesh which are needed to set up system (13). (Handling an irregular mesh by this rotation approach, the equations simply remain in the form (4).) It is clear that the resulting discrete system is conservative, also for the irregular mesh.

2.2 Osher's approximate Riemann solver

A convenient numerical flux function $f(q_0, q_1) = f(q^l, q^r)$ is Osher's approximate Riemann solver [28]. In this subsection we give a short description of this function. In fact, we may distinguish two strongly related variants: the O-(original) variant and the P-(physical) variant [10]. Here we restrict ourselves to the P-variant. It is our experience that it yields very reliable discretizations. Though being less complex than the O-variant, its main disadvantage still seems to be its supposed complexity when compared with other approximate Riemann solvers (such as e.g. those of Steger-Warming, Van Leer and Roe). An objective of our present exposition is to show that the scheme can be implemented in a simple and straightforward way. Further, we need this description to show (in section 2.4) how its linearization is obtained.

According to Osher, the numerical flux function is defined by

$$f(q_0, q_1) = \frac{1}{2}\left[f(q_0) + f(q_1) - \int_{q_0}^{q_1} \left| \frac{df(q)}{dq} \right| dq \right], \tag{17}$$

where

$$\left| \frac{df(q)}{dq} \right| = R|\Lambda|R^{-1}, \tag{18}$$

with $|\Lambda|$ the diagonal matrix of the absolute values of the eigenvalues λ of the Jacobian $df(q)/dq$. In eq. (17) the integration path is still to be defined, but we know

that the Jacobian has a complete set of eigenvalues λ_k, $k=1,2,3,4$: $\lambda_1=u-c$, $\lambda_2=\lambda_3=u$, $\lambda_4=u+c$, and a set of three corresponding eigenspaces $R_1, R_{2,3}$ and R_4. The integral in eq. (17) is computed along a path $q=q(s)$, $0\leqslant s\leqslant 1$, $q(0)=q_0$, $q(1)=q_1$. This path is divided into subpaths Γ_k, $k=1,2,3$, connecting the states $q_{(k-1)/3}$ and $q_{k/3}$. Each subpath Γ_k is constructed such that the direction of $dq(s)/ds$ is tangential to $R_{m(k)}$ [10], the corresponding eigenvector. In the P-variant, the choices for $R_{m(k)}$ are: $R_{m(1)}=R_1$, $R_{m(2)}=R_{2,3}$, $R_{m(3)}=R_4$. The states $q_{1/3}$ and $q_{2/3}$ are computed by means of the Riemann invariants $\psi_l^{m(k)}(q(s))$, $l\neq m$, $l=1,2,3,4$, which are constant along Γ_k [30].

The state q is suitably expressed in the dependent variables c,u,v and z, where $z=\ln(p\rho^{-\gamma})$ is an entropy function. We obtain directly: $z_{1/3}=z_0$, $z_{2/3}=z_1$, $v_{1/3}=v_0$, $v_{2/3}=v_1$ and $p_{1/3}=p_{2/3}$. Defining

$$\alpha=e^{\frac{z_1-z_0}{2\gamma}}, \tag{19a}$$

$$\Psi_0=u_0+\frac{2}{\gamma-1}c_0, \tag{19b}$$

$$\Psi_1=u_1-\frac{2}{\gamma-1}c_1, \tag{19c}$$

we also find (assuming that no cavitation occurs, $\Psi_0>\Psi_1$):

$$c_{1/3}=\frac{\gamma-1}{2}\frac{\Psi_0-\Psi_1}{1+\alpha}, \tag{20a}$$

$$c_{2/3}=\alpha c_{1/3}, \tag{20b}$$

$$u_{1/2}=u_{1/3}=u_{2/3}=\frac{\Psi_1+\alpha\Psi_0}{1+\alpha}. \tag{20c}$$

The eigenvalues at the points $q_{k/3}$, $k=0,1,2,3$, are:

$$\bar{\lambda}_0=\lambda_{m(1)}(q_0)=u_0-c_0, \tag{21a}$$

$$\bar{\lambda}_{1/3}=\lambda_{m(1)}(q_{1/3})=u_{1/3}-c_{1/3}, \tag{21b}$$

$$\bar{\lambda}_{1/2}=\lambda_{m(2)}(q_{1/3})=\lambda_{m(2)}(q_{2/3})=u_{1/3}=u_{2/3}, \tag{21c}$$

$$\bar{\lambda}_{2/3}=\lambda_{m(3)}(q_{2/3})=u_{2/3}+c_{2/3}, \tag{21d}$$

$$\bar{\lambda}_1=\lambda_{m(3)}(q_1)=u_1+c_1. \tag{21e}$$

Because λ_1 and λ_4 are genuinely nonlinear eigenvalues, $\lambda_{m(k)}(q(s))$ is monotonous along Γ_1 and Γ_3, and hence it changes sign at most once along Γ_1 and Γ_3. A sonic point q_{s_1} with $\lambda_{m(1)}(q(s_1))=0$ exists on Γ_1 if $\bar{\lambda}_0\bar{\lambda}_{1/3}\leqslant 0$. This sonic point $q_{s_1}=(c_{s_1},u_{s_1},v_{s_1},z_{s_1})$, is computed from the linear system

$$u_{s_1}-c_{s_1}=0, \tag{22a}$$

$$u_{s_1}+\frac{2}{\gamma-1}c_{s_1}=\Psi_0, \tag{22b}$$

$$v_{s_1} = v_0, \tag{22c}$$

$$z_{s_1} = z_0. \tag{22d}$$

Similarly, a sonic point q_{s_2} is found on Γ_3 if $\bar{\lambda}_{5/6}\bar{\lambda}_1 \leqslant 0$. Along the complete path $q(s)$, $0 \leqslant s \leqslant 1$, $\lambda_{m(k)}(q(s))$ can change sign only at the points $q_{1/6}$, $q_{3/6}$, q_{s_1} and q_{s_2}. We notice that $f(q_0, q_1)$ according to Osher, is a continuous function in all $\bar{\lambda}$'s and we see that $\bar{\lambda}_{1/6} \leqslant \bar{\lambda}_{2/6} < \bar{\lambda}_{3/6}$. Because of this continuity we may neglect the case of a zero eigenvalue $\bar{\lambda}$ and we compute the numerical flux by

$$\begin{aligned} f(q_0, q_1) &= H(\bar{\lambda}_0) f(q_0) \\ &+ H(-\bar{\lambda}_0\bar{\lambda}_{1/6}) \operatorname{sign}(\bar{\lambda}_{1/6}) f(q_{s_1}) \\ &+ H(-\bar{\lambda}_{1/6}\bar{\lambda}_{2/6}) f(q_{1/6}) \\ &+ H(-\bar{\lambda}_{2/6}\bar{\lambda}_{3/6}) f(q_{3/6}) \\ &+ H(-\bar{\lambda}_{3/6}\bar{\lambda}_1) \operatorname{sign}(\bar{\lambda}_1) f(q_{s_2}) \\ &+ H(-\bar{\lambda}_1) f(q_1), \end{aligned} \tag{23}$$

where $H(\lambda)$ is the Heaviside unit step function; $H(\lambda) = 0$ for $\lambda < 0$ and $H(\lambda) = 1$ for $\lambda > 0$. In most cases, many eigenvalues $\bar{\lambda}$ will have equal signs. If the ordered sequence $\bar{\lambda}_0, \bar{\lambda}_{1/6}, \bar{\lambda}_{2/6}, \bar{\lambda}_{3/6}, \bar{\lambda}_1$ can be split into two parts (possibly empty), the first one containing only negative eigenvalues and the second one only positive eigenvalues, then a $\hat{q}$ exists such that simply $f(q_0, q_1) = f(\hat{q})$. We identify this state $\hat{q}$ as the state of the gas *at* the cell wall. This situation occurs for instance for fully supersonic or fully subsonic cases. If we exclude the unlikely cases $u_{1/2} < 0$, $u_0 - c_0 > 0$ and $u_{1/2} > 0$, $u_1 + c_1 < 0$, the numerical flux near a shock is the only one for which $f(q_0, q_1)$ is found to be a sum of more (namely three) terms $f(q)$. For more details we refer to [10] and in particular to [32].

2.3 The numerical flux at the boundary

The flux $f_{i,j,k}$ at the boundary $\partial\Omega^*$ is partially determined by $q_{i,j}$, the state of the flow in the boundary cell, and partially by the boundary conditions [26]. To compute $f_{i,j,k}$ at $\partial\Omega^*$, first, from $q_{i,j}$ and the corresponding boundary conditions, we determine the state $q_B = q_{i,j,k}$ at the boundary $\partial\Omega^*$. Then the P-variant of Osher's approximate Riemann solver is used to compute the boundary flux. This is completely consistent with the discretization over internal cell walls as described in section 2.2.

To satisfy the boundary conditions in system (13), we determine q_B, the state at the boundary, such that it satisfies the boundary conditions, i.e. $B(q_B) = 0$, as well as the equality (assuming that the boundary is at the left):

$$f_{i,j,k} = f(q_B) = f(q_B, q_{i,j}). \tag{24}$$

In view of (17), eq. (24) implies

$$\int_{q_B}^{q_{i,j}} \frac{df(w)}{dq} dw = \int_{q_B}^{q_{i,j}} \left| \frac{df(w)}{dq} \right| dw, \tag{25}$$

i.e. q_B should satisfy the boundary conditions and should be connected with $q_{i,j}$ by a path $q(s)$ such that

$$\lambda_{m(k)}(q(s)) \geqslant 0. \tag{26}$$

Now only the eigenvectors corresponding to the positive eigenvalues can be used and the number of subpaths to be considered depends on the number of in-going characteristics.

2.4 The linearization of Osher's scheme

In the multigrid method, see section 2.5, we apply a point relaxation method. In this relaxation, locally, a system of four nonlinear equations has to be solved. An efficient way of doing this is by Newton iteration. For this we need convenient expressions for $dN_h(q_h)/dq_h$. From eqs. (13) and (14) we derive

$$\frac{\partial (N_h(q_h))_{i,j}}{\partial q_{l,m}} = \frac{\partial}{\partial q_{l,m}} \sum_k s_{i,j,k} f_{i,j,k}(q_{i,j}, q_{i,j,k})$$

$$= \sum_k s_{i,j,k} \frac{\partial}{\partial q_{i,j}} f_{i,j,k}(q_{i,j}, q_{i,j,k}), \quad \text{if } \Omega_{l,m} = \Omega_{i,j}, \tag{27a}$$

$$= s_{i,j,k} \frac{\partial}{\partial q_{i,j,k}} f_{i,j,k}(q_{i,j}, q_{i,j,k}), \quad \text{if } \Omega_{l,m} = \Omega_{i,j,k}, \tag{27b}$$

$$= 0 \quad \text{otherwise}. \tag{27c}$$

Now, in view of eqs. (16a) and (16b), the computation of $dN_h(q_h)/dq_h$ reduces to evaluations of

$$f'_{(0)}(q_0, q_1) = \frac{\partial f(q_0, q_1)}{\partial q_0}, \tag{28a}$$

$$f'_{(1)}(q_0, q_1) = \frac{\partial f(q_0, q_1)}{\partial q_1}. \tag{28b}$$

If in (27a) $q_{i,j,k} = q_B$ is a boundary state, then a relation $q_{i,j,k} = q_B(q_{i,j})$ exists and the corresponding term in (27a) is (assuming again that the boundary is at the left):

$$s_{i,j,k} \frac{d}{dq_{i,j}} f_{i,j,k}(q_{i,j}, q_{i,j,k}) = s_{i,j,k} \frac{d}{dq_{i,j}} f_{i,j,k}(q_{i,j}, q_B(q_{i,j}))$$

$$= s_{i,j,k} \frac{d}{dq_{i,j}} \left[T^{-1} f(T q_B(q_{i,j}), T q_{i,j}) \right]$$

$$= s_{i,j,k} T^{-1} f'_{(0)}(T q_B, T q_{i,j}) T \frac{dq_B}{dq_{i,j}} + s_{i,j,k} T^{-1} f'_{(1)}(T q_B, T q_{i,j}) T, \tag{29}$$

where T denotes $T_{i,j,k}$ as in eqs. (16a) and (16b). The derivative matrix $dq_B/dq_{i,j}$ depends on the specific boundary conditions imposed and is derived from the relation $q_B(q_{i,j})$.

We already noticed that the integration paths are easily expressed in the dependent variables c,u,v and z. Similarly, the numerical flux and its partial derivatives are conveniently expressed in the same variables. The flux vector $f=(\rho u, \rho u^2+p, \rho uv, u(E+p))^T$ is expressed as a function of $q=(c,u,v,z)^T$ by using

$$\rho = \left[\frac{1}{\gamma} e^{-z} c^2\right]^{\frac{1}{\gamma-1}}, \tag{30a}$$

$$p = \frac{1}{\gamma}\rho c^2, \tag{30b}$$

$$E = \rho e = \tfrac{1}{2}\rho(u^2+v^2) + \frac{1}{\gamma(\gamma-1)}\rho c^2. \tag{30c}$$

For the variables c,u,v and z, the derivative matrix

$$f'(q) = \frac{df}{dq} = \frac{\partial(\rho u, \rho u^2+p, \rho uv, u(E+p))}{\partial(c,u,v,z)} \tag{31}$$

reads

$$f'(q) = \begin{bmatrix} \beta\rho u/c & \rho & 0 & -\tfrac{1}{2}\beta\rho u \\ \beta\rho(u^2+c^2)/c & 2\rho u & 0 & -\tfrac{1}{2}\beta(\rho u^2+p) \\ \beta\rho uv/c & \rho v & \rho u & -\tfrac{1}{2}\beta\rho uv \\ \beta u(E+p+\rho c^2)/c & \rho u^2+E+p & \rho uv & -\tfrac{1}{2}\beta u(E+p) \end{bmatrix}, \tag{32}$$

where $\beta = 2/(\gamma-1)$. In terms of this derivative matrix, from (23) it follows

$$\begin{aligned} \frac{\partial f(q_0,q_1)}{\partial q_0} &= H(\bar{\lambda}_0) f'(q_0) \\ &+ H(-\bar{\lambda}_0\bar{\lambda}_{1/3}) \operatorname{sign}(\bar{\lambda}_{1/3}) f'(q_{s_1}) \frac{\partial q_{s_1}}{\partial q_0} \\ &+ H(-\bar{\lambda}_{1/3}\bar{\lambda}_{1/2}) f'(q_{1/3}) \frac{\partial q_{1/3}}{\partial q_0} \\ &+ H(-\bar{\lambda}_{1/2}\bar{\lambda}_{2/3}) f'(q_{2/3}) \frac{\partial q_{2/3}}{\partial q_0}. \end{aligned} \tag{33}$$

The derivatives $\partial q/\partial q_0$, $q = q_{s_1}, q_{1/3}, q_{2/3}$, are derived from differentiable relations such as (19), (20) and (22). Explicit expressions are found in [10]. In this way the matrices $f'_{(0)}(q_0,q_1)$ and $f'_{(1)}(q_0,q_1)$ are readily computed. It appears that both matrices are continuous functions of q_0 and q_1 as long as $\bar{\lambda}_{1/2} = u_{1/3} = u_{2/3} \neq 0$. An efficient implementation is obtained by expressing the fluid state in the Riemann-like state variables c,u,v,z.

2.5 MULTIGRID ITERATION

In order to solve the discrete equations (14), first we slightly generalize them to

$$N_h(q_h)=r_h. \tag{34}$$

For the solution of (34) we apply then nonlinear multigrid iteration (in the FAS-variant [3]). For this we need a sequence of discretizations

$$N_{h_l}(q_{h_l}), \quad l=0,1,2,...,L, \quad h_0>h_1>h_2\cdots>h_L=h. \tag{35}$$

For a regular mesh with size h_{l-1}, we take $h_{l-1}=2h_l$. For an irregular mesh we delete each second line of mesh points to obtain the cells in the coarser grid. Further, we introduce grid transfer operators $R_{2h,h}: X_h \to X_{2h}$ and $\bar{R}_{2h,h}: Y_h \to Y_{2h}$ (restrictions that make a representation onto the level $2h$ of a grid function at the level h), and $P_{h,2h}: X_{2h} \to X_h$ (which interpolates a solution-function at level $2h$ to the level h). Now, one iteration cycle of the FAS-algorithm for the solution of eq. (34) consists of the following steps:

0. start with an approximate solution q_h,
1. improve q_h by application of p (pre-) relaxations to $N_h(q_h)=r_h$,
2. compute the residual $N_h(q_h)-r_h$,
3. find an approximation of q_h at the next coarser grid, say q_{2h}. (For this we use either a restricted solution $q_{2h}=R_{2h,h}q_h$, or a previously obtained approximation q_{2h}),
4. compute $r_{2h}=N_{2h}(q_{2h})+\bar{R}_{2h,h}(r_h-N_h(q_h))$,
5. approximate the solution of $N_{2h}(q_{2h})=r_{2h}$, by application of σ nonlinear multigrid cycles. The result is $\tilde{q}_{2h}$,
6. correct the current solution by $q_h=q_h+P_{h,2h}(\tilde{q}_{2h}-q_{2h})$,
7. improve q_h by application of q (post-) relaxations to $N_h(q_h)=r_h$.

The steps 2-6 in this process constitute the *coarse grid correction.* These steps are skipped at the coarsest grid h_0. For the solution of the nonlinear system (14), the FAS-algorithm is applied with $r_h=0$ at the finest grid. During the FAS-iteration, at the coarser grids, non-zero right-hand sides appear. In order to complete the description of the FAS-cycle we need to be explicit about: (i) the choice of the operators N_{2h}, $P_{h,2h}$, $\bar{R}_{2h,h}$ and possibly $R_{2h,h}$, (ii) the FAS-strategy, i.e. the numbers p, q, σ, (iii) the nonlinear relaxation method, and (iv) the computation of an initial guess for the FAS-iteration. These subjects will be treated in the following paragraphs.

2.5.1 A nested sequence of Galerkin discretizations. For the operators $P_{h,2h}$ and $\bar{R}_{2h,h}$ we make a choice that is consistent with the concept of our finite volume discretization. The discretization is essentially a weighted residual method, where the solution is approximated by a piecewise constant function (on cells $\Omega_{i,j}$) and where the residual is weighted by characteristic functions on $\Omega_{i,j}$. From this point of view, it is natural to use a piecewise constant interpolation for $P_{h,2h}$ and to use addition over subcells for $\bar{R}_{2h,h}$. Notice that $\bar{R}_{2h,h}$ is the adjoint of $P_{h,2h}$. With these choices it is clear that

$$N_{2h}^1(q_{2h}) = \bar{R}_{2h,h} N_h^1(P_{h,2h} q_{2h}), \tag{36}$$

i.e. the coarse grid finite volume discretization is a formal Galerkin approximation of the fine grid finite volume discretization. By the superscript 1 (starting from eq. (36)) we indicate explicitly that the discretization used is first-order accurate. Applying (36) on all different levels we obtain a nested sequence of discretizations.
The effect of the Galerkin approximation $N_{2h}^1 = \bar{R}_{2h,h} N_h^1 P_{h,2h}$ on the approximate solution $\tilde{q}_h$ obtained after a coarse grid correction is the following. If we take $q_{2h} = R_{2h,h} q_h$ in step 3 of the algorithm, with $R_{2h,h}$ such that $R_{2h,h} P_{h,2h} = I_{2h}$ is the identity operator on X_{2h}, and if $N_{2h}(q_{2h}) = r_{2h}$ is solved exactly, then

$$\bar{R}_{2h,h}\left[r_h - N_h^1(P_{h,2h} R_{2h,h} \tilde{q}_h)\right] = \bar{R}_{2h,h}\left[N_h^1(q_h) - N_h^1(P_{h,2h} R_{2h,h} q_h)\right], \tag{37}$$

or, for the restriction of the residual

$$\bar{R}_{2h,h}\left[r_h - N_h^1(\tilde{q}_h)\right] = \bar{R}_{2h,h}\left[\left[N_h^1(q_h) - N_h^1(P_{h,2h} R_{2h,h} q_h)\right] - \left[N_h^1(\tilde{q}_h) - N_h^1(P_{h,2h} R_{2h,h} \tilde{q}_h)\right]\right]. \tag{38}$$

In the neighborhood of a solution, the difference $q_h - \tilde{q}_h$ will be small and N_h^1 will approximately behave as a linear operator: the restriction of its residual will be very small; $\mathcal{O}(\| q_h - \tilde{q}_h \|^2)$. For a sufficiently differentiable operator N_h^1, this implies

$$\bar{R}_{2h,h}\left[r_h - N_h^1(\tilde{q}_h)\right] = \mathcal{O}(\| q_h - \tilde{q}_h \|^2). \tag{39}$$

Because $\bar{R}_{2h,h}$ is an addition over four neighboring cells, this means that the residual mainly contains high-frequency components. A small restriction of the residual implies that large residuals cancel over neighboring cells. Because the residual is varying rapidly, local relaxation methods should be able to eliminate such residuals efficiently.

2.5.2 Multigrid strategy. Experience with multigrid algorithms in other contexts shows that $p = q = \sigma = 1$ (i.e. a multigrid V-cycle with a single pre- and post-relaxation sweep) may be a good choice for a successful strategy. It is the standard choice in our computations. Other choices, with small values for p, q and σ, can be made. What is best depends much on the relaxation used, and research can be made for seeking the most efficient combination. However, the result may depend on the particular problem solved. Up to now, it appears that different (p, q, σ)-strategies are not much different in efficiency. A smaller convergence factor is usually compensated by a corresponding amount of additional work.

2.5.3 Relaxation. The important feature for a relaxation method in a multiple grid context (both for linear and nonlinear problems) is its capacity to damp the high-frequency components in the error. Therefore, the difference scheme should be sufficiently dissipative. The first-order upwind schemes usually are. An advantage of these schemes over central difference schemes is that their numerical dissipation is

well-defined and independent of any parameter, this as opposed to central difference schemes. For the relaxation method several alternatives are available. For nonlinear multigrid methods most experience exists for methods of the collective Gauss-Seidel type. Here, all the cells in the computational domain are scanned in a well-defined order, and when a cell is visited, the four state variables (c,u,v,z) are updated simultaneously. For the solution of the corresponding system of four nonlinear equations, one or more steps of a Newton iteration are used until the local residual is reduced below a specified amount. In almost all cases it appears to be most efficient to take this tolerance so crude that usually no more than a single iteration step per cell is performed. Possible relaxations are: (i) Gauss-Seidel-relaxation with lexicographical ordering (GS), (ii) symmetric Gauss-Seidel-relaxation from north-west to south-east and vice versa (SGS1), (iii) the same but going from north-east to south-west and vice versa (SGS2), (iv) checkerboard- (or red-black-) relaxation (RB). In almost all cases the same relaxation can be used in both the pre- and post-relaxation. Another good choice is SGS3: to use SGS1 for the pre- and SGS2 for the post-relaxation. In [9], some of these relaxations are compared in combination with a uniform grid. There, also the effect of other (p,q,σ)-strategies is considered. For a standard model problem (transonic flow in a channel with circular-arc bump) on a non-uniform grid and with SGS3, the following convergence rates per multigrid cycle are obtained: 0.38 for $p=q=1,\sigma=1$; 0.23 for $p=q=1,\sigma=2$; and 0.17 for $p=q=2,\sigma=1$. The smoothing behavior of the different possible relaxation methods can be analyzed by local mode analysis, but we should notice that the *smoothing factor* as used for common elliptic problems, has no significant meaning for the Euler equations because we have to take into account (unstable) characteristic modes. Here, a local mode analysis should follow more the lines used for elliptic singular perturbation problems, see e.g. [13]. Jespersen has published some results [12], in which he shows that for a subsonic and a supersonic case SGS has a reasonably good smoothing behavior, when applied to a first-order scheme. Of course, the non-symmetric GS-relaxation is only effective if the direction of the relaxation sufficiently conforms with the direction of the characteristics. Although there is no proof, experience shows that for transonic flows the convergence rate for FAS-iteration, with SGS as pre- and post-relaxation, is almost grid-independent.

2.5.4 Initial estimates. For the nonlinear multigrid method as just described, it is important to start with reasonably good initial estimates. These can be obtained by nested iteration. Here the solution is first approximated on the coarsest grid. Then the solution is interpolated onto the next finer grid, where a few FAS-cycles are performed. This procedure is repeated until the required finest grid has been reached. In many cases, for starting the nested iteration, a very crude initial estimate on the coarsest grid can be used. As soon as the solution on the coarsest mesh is approximated with sufficient accuracy, it is interpolated to the finer grid. It can be shown that, for all finer levels, a small, fixed number of multigrid iterations is sufficient to obtain truncation error accuracy.

THEOREM Consider a sequence of discretizations $N_{h_l}(q_{h_l})=r_{h_l}$, $l=0,1,2,...,L$. with $h_{l-1}/h_l>C_1$. If the discrete equations are relatively convergent of order p, i.e. if

$$||P_{l,l-1}q_{l-1}-q_l||\leqslant C_0 h_{l-1}^p, \tag{40}$$

and the convergence of the iteration cycle is independent of h, i.e. for the iterates q_l^n in the iterative solution process we have

$$||q_l^{n+1}-q_l||\leqslant C_2||q_l^n-q_l||, \tag{41}$$

then, with N cycles on each level, the result $\tilde{q}_h=q_h^N$ of the nested iteration process satisfies

$$||\tilde{q}_h-q_h||<\frac{C_1^p C_2^N}{1-C_1^p C_2^N||P||}C_0 h_l^p, \tag{42}$$

assuming that $C_1^p C_2^N||P||<1$, where $||P||=\sup_l||P_{l,l-1}||$. □

Hence, the interpolation used to obtain the first guess on each level should be of sufficiently high order to comply with the accuracy of the discretization. In our case, where the discretization is first-order accurate, the first-order prolongation $P_{h,2h}$ as used in the Galerkin approximation is not accurate enough, and a second-order interpolation is necessary.

2.6 CONCLUSION

We have seen that for many steady Euler flow computations, good multigrid efficiency can be obtained [9,10,15,19]. A good sequence of first-order discretizations is obtained by the consistent use of the finite volume technique. It yields a conservative discretization and it induces both the prolongations and the restrictions for the multigrid algorithm. The result is a nested sequence of Galerkin discretizations. Probably the most important ingredient in the finite volume discretization is the choice of a good numerical flux function. The flux function chosen (Osher's) allows a completely consistent treatment of the interior and the boundary of the domain. Both at the domain boundary and in the interior, Riemann invariants are used for transferring information across cell faces. Further, Osher's numerical flux function has smooth derivatives, which allows the use of Newton's method in the relaxation. A slight variant of Osher's approximate Riemann solver (the P-variant) leads to a favorable efficiency.

By the use of nested iteration, sufficiently accurate initial estimates can be obtained (for the cost of about 1⅓ FAS-cycle). Already for some practically interesting problems, only a single FAS-cycle (with $p=q=\sigma=1$ and SGS3-relaxation) appears to be sufficient for obtaining truncation error accuracy. This means that the (non-isenthalpic) steady Euler equations can be solved by an amount of work that is equivalent with about 1⅓×2 symmetric Gauss-Seidel relaxation sweeps.

3. DEFECT CORRECTION FOR HIGHER-ORDER EULER COMPUTATIONS

3.1 SECOND-ORDER DISCRETIZATION

The first-order discretization introduced in section 2.1 has a number of advantages: it is conservative, monotonous and it gives a sharp representation of discontinuities (shocks and contact discontinuities), as long as these are aligned with the mesh. Further, it allows an efficient solution of the discrete equations by a multigrid method. Disadvantages are: the low order of accuracy (many points are required to find an accurate representation of a smooth solution) and the fact that it is highly diffusive for oblique discontinuities. (Oblique discontinuities are smeared out over a large number of cells.) For a first-order (upwind) scheme these are well-known facts which have led to the search for higher-order methods.

A key property of the first-order discretization, that we also want to have in a second-order scheme, is the conservation of q. Conservation allows discontinuities to be captured as weak solutions of (6) and avoids the necessity of a shock fitting technique. Therefore, we consider only schemes that are still based on (12), and we select a new $f_{i,j,k}(q^k_{i,j}, q^k_{i,j,k})$ such that we get a better approximation to (11) than with (16).

Higher-order discretizations can be obtained in two different ways. Higher-order interpolation can be performed either for the states (i.e. in X_h) or for the fluxes (i.e. in Y_h). The first approach, called the MUSCL-approach, is used in e.g. [2,4,23], the second approach in e.g. [27,33]. In the MUSCL-approach, in (12), $q^k_{i,j}$ and $q^k_{i,j,k}$ are obtained by some interpolation in $q_h = \{q_{i,j}\}$. In the other approach, $f_{i,j,k}(q^k_{i,j}, q^k_{i,j,k})$ is obtained by some interpolation in $f_h = \{f_{i,j,k}(q_{i,j}, q_{i,j,k})\}$. In the following we restrict ourselves to the more common MUSCL-approach.

From the point of view of finite volume discretization, a straightforward way to form a more accurate approximation is to replace the first-order approximation (16) with its piecewise constant approximation $\tilde{q}(x,y)$ over cells, by a piecewise bilinear function $\tilde{q}(x,y)$ on a set of 2×2 cells (a *superbox*). Such a superbox at the h-level corresponds with a single cell at the $2h$-level. Across the boundaries of the superbox, $\tilde{q}(x,y)$ can be discontinuous. In the superbox $\tilde{q}(x,y)$ is determined by $q_{2i,2j}$, $q_{2i+1,2j}$, $q_{2i,2j+1}$, $q_{2i+1,2j+1}$. Using such a bilinear function, we see that the central difference approximation is used for flux computations inside the superboxes. At the superbox boundaries, interpolation is made from the left and the right, and the approximate Riemann solver is used to compute the flux at the boundary. We denote the corresponding discrete operator by N^S_h. It is easily shown that the superbox scheme is second-order accurate in the sense that

$$\overline{R}_{2h,h}(N^S_h(R_h q) - \overline{R}_h N(q)) = \mathcal{O}(h^2). \tag{43}$$

Instead of the finite volume superbox scheme, we can adopt a finite difference approach. Interpolation from the left and right can be used to obtain the states $q^l_{i,j,k}$ and $q^r_{i,j,k}$ at the left and right cell faces, respectively. The simplest second-order scheme is the central differencing scheme. Here the interpolation leads to a loss of all upwind properties. It simply leads to $f(q^l, q^r) = f(\frac{1}{2}(q^l + q^r))$ for the numerical

flux function. In contrast with the first-order scheme, the central difference scheme may even be anti-diffusive, which may lead to instabilities. When a central scheme is used alone, an additional diffusion (dissipation) term should be added to stabilize the solution method [11].

To improve the stability behavior, it is better to take into account the domain of dependence of the solution (the direction of the characteristics) and to distinguish, at each cell face, between interpolated values from the left and from the right. For simplicity of notation we shall exemplify this only for the one-dimensional case. Generalization to two dimensions is straightforward. In the one-dimensional case, eq. (13) reduces to $f_{i+\frac{1}{2}}-f_{i-\frac{1}{2}}=0$, where $f_{i+\frac{1}{2}}=f(q^l_{i+\frac{1}{2}},q^r_{i+\frac{1}{2}})$. Introducing $\Delta q_{i+\frac{1}{2}}=q_{i+1}-q_i$, we find for the second-order upwind interpolated values $q^l_{i+\frac{1}{2}}$ and $q^r_{i+\frac{1}{2}}$:

$$q^l_{i+\frac{1}{2}}=q_i+\tfrac{1}{2}\Delta q_{i-\frac{1}{2}}, \tag{44a}$$

$$q^r_{i+\frac{1}{2}}=q_{i+1}-\tfrac{1}{2}\Delta q_{i+1\frac{1}{2}}. \tag{44b}$$

Notice that on a non-equidistant grid, second-order accuracy for $h\to 0$ is guaranteed only if the grid is sufficiently smooth.

Though stability properties of these one-sided approximations are better than those of central approximations, stability and monotonicity are still not guaranteed. The usual way to force monotonicity is by introducing for each k-th state vector component ($k=1,2,3,4$) a limiting function [31,35], and to interpolate by

$$q^{l\,(k)}_{i+\frac{1}{2}}=q^{(k)}_i+\tfrac{1}{2}\psi^{l\,(k)}_{i+\frac{1}{2}}\Delta q^{(k)}_{i-\frac{1}{2}} \tag{45a}$$

$$q^{r\,(k)}_{i-\frac{1}{2}}=q^{(k)}_i-\tfrac{1}{2}\psi^{r\,(k)}_{i-\frac{1}{2}}\Delta q^{(k)}_{i+\frac{1}{2}}, \tag{45b}$$

where the limiting functions $\psi^{l\,(k)}=\psi(R^{(k)})$ and $\psi^{r\,(k)}=\psi(1/R^{(k)})$ are chosen, depending on the ratio $R^{(k)}=\Delta q^{(k)}_{i+\frac{1}{2}}/\Delta q^{(k)}_{i-\frac{1}{2}}$, such that $q^{r\,(k)}_{i-\frac{1}{2}}$ lies between $q^{(k)}_{i-1}$ and $q^{(k)}_i$, and $q^{l\,(k)}_{i+\frac{1}{2}}$ between $q^{(k)}_i$ and $q^{(k)}_{i+1}$ [31,35]. One possible choice is the Van Albada limiter [1]:

$$\psi(R)=\frac{R^2+R}{R^2+1}. \tag{46}$$

In [23], Van Leer proposes still another higher-order discretization; a linear combination of the one-sided and central interpolation. Parametrized by κ it reads

$$q^l_{i+\frac{1}{2}}=q_i+\tfrac{1}{4}\left[(1-\kappa)\Delta q_{i-\frac{1}{2}}+(1+\kappa)\Delta q_{i+\frac{1}{2}}\right], \tag{47a}$$

$$q^r_{i-\frac{1}{2}}=q_i-\tfrac{1}{4}\left[(1-\kappa)\Delta q_{i+\frac{1}{2}}+(1+\kappa)\Delta q_{i-\frac{1}{2}}\right]. \tag{47b}$$

This general formula contains e.g.: (i) the one-sided second-order scheme (44) ($\kappa=-1$), (ii) Fromm's scheme ($\kappa=0$), (iii) a third-order accurate, upwind biased scheme ($\kappa=\frac{1}{3}$), and (iv) the central difference scheme ($\kappa=1$). In the one-dimensional case, the superbox scheme, N^S_h, corresponds to the use of $\kappa=+1$ for odd i, and $\kappa=-1$ for even i.

The interpolations (45) and (47) are well-defined in the interior cells of the domain. In the cells near the boundary $\partial\Omega^*$, one of the values $\Delta q_{i\pm 1/2}$ is not defined, by the absence of a value q_i corresponding to a point outside Ω^*. Here, some sort of super-box approximation may be used.
In conclusion: with the MUSCL-approach, here we have constructed a higher-order accurate semi-discretization of (7):

$$\frac{\partial}{\partial t}\int_{\Omega} q_h dxdy + N_h^2(q_h)=0. \tag{48}$$

3.2 The solution of the higher-order discrete system

One possible way to find the solution of the steady state equations

$$N_h^2(q_h)=0, \tag{49}$$

is to take an initial guess and to solve the semi-discretized equation (48) for $t\to\infty$, i.e. to compute the time-dependent solution $q_h(t)$ until initial disturbances have died out sufficiently. However, this process may be slow. Just as for the first-order discretized equations, we take the fully implicit approach and try to solve the system

$$N_h^2(q_h)=r_h \tag{50}$$

directly. However, if we try to solve the higher-order system (49) in the same manner as we solve the first-order equations, we may expect difficulties because the nonlinear equations (49) are less stable. The higher-order discretizations are less diffusive, and (as already mentioned) in the case of central differences they may even be 'anti-diffusive'. This may lead not only to non-monotonous solutions, but it can also cause a Gauss-Seidel relaxation not to reduce the rapidly varying error components. A local mode analysis of the smoothing properties of GS-relaxation for first- and higher-order upwind Euler discretizations can be found in [12]. There, the flux splitting upwind scheme of Steger and Warming is analyzed. Similar results apply for Osher's scheme. Further numerical evidence that convergence of a relaxation process for a higher-order upwind discretization is slower than for a first-order upwind discretization, is found in [24,25], where Van Leer's flux splitting is applied.
To obtain higher-order accurate solutions, we do not solve the system $N_h^2(q_h)=0$ as such. We use the first-order operator N_h^1 as described in section 2, to find a higher-order accurate approximation in a defect correction iteration:

$$N_h^1(q_h^1)=0, \tag{51a}$$

$$N_h^1(q_h^{n+1})=N_h^1(q_h^n)-N_h^2(q_h^n), \quad n=1,2,...,N. \tag{51b}$$

Both theory [5] and practice [6] show that if the problem is smooth enough, already q_h^2 is second-order accurate. If the solution is not smooth (i.e. when higher-order derivatives are dominating), there is no reason to expect the solution of (49) to be more accurate than the solution of (51a). Nevertheless, in [6,7,8,14] evidence is given that only a few defect correction steps may improve the (non-smooth) solution significantly.

In fact, we may use $q_h^{n+1}-q_h^n$ as an error indicator. In the smooth parts of the solution $q_h^{1+n}-q_h^1=\mathcal{O}(h)$ and $q_h^{2+n}-q_h^2=\mathcal{O}(h^2)$. Where these differences are larger, i.e. $\mathcal{O}(h^0)$, the solution is not smooth (relative to the the grid used). There grid adaptation is to be considered rather than the choice of a higher-order method, if a more accurate solution is wanted. Eq. (51b) describes an iterative process, in which a first-order system has to be solved (iteratively) in each step. In practice the inner iteration can be kept restricted to a single FAS-cycle [14].

In a multigrid context, where solutions on more grids are available, it is also natural to consider other approaches for computing higher-order solutions, such as: (i) Richardson extrapolation, and (ii) τ-extrapolation. Both extrapolation methods can well be used to find a more accurate solution if the solution is smooth [6]. A drawback is that both methods rely on the existence of an asymptotic expansion of the truncation error for $h\to 0$, and (in general) no a-priori information exists about the validity of such an assumption. Another disadvantage is that the accurate solution (for Richardson extrapolation) or the estimate for the truncation error (for τ-extrapolation) is obtained at the one-but-finest level. Because we want not only a higher order of accuracy, but also a more accurate representation of possible discontinuities, it is advised to use Richardson extrapolation (only) as a cheap means to find a higher-order initial estimate for the iteration process (51b). Since the evaluation of $N_h^2(q_h)$ is hardly more expensive than the evaluation of $N_h^1(q_h)$, the costs to compute the defect in (51b) are of the same order as the evaluation of the relative truncation error $\tau_{2h,h}(q_h)=N_{2h}^1(R_{2h,h}q_h)-R_{2h,h}N_h^1(q_h)$. This makes us to prefer defect correction, rather than τ-extrapolation.

3.3 The complete multigrid algorithm

We aim at the efficient computation of the approximate solution q_h of the second-order discretized Euler equations (49) on a given mesh with $h=h_L$, where we assume that a number of L coarser meshes exists, for which $h_l\approx 2^{L-l}h_L$. We denote the level of multigrid refinement again by l, and the approximate solution at level l again by q_l. As explained in section 2.5.4, the coarser grids, $l<L$, are also used in the construction of the initial estimates for the iteration processes. With FASCYCLE $(N_l q_l=r_l)$ denoting a single FAS-cycle as described in section 2.5, the algorithm used to obtain the initial estimate and further iterates in the defect correction process, is as follows:

0. start with an approximation for q_0;
1a. $l:=0$;
1b. **for** j **from** 1 **to** k_l **do** FASCYCLE $(N_l^1 q_l=0)$ **enddo**;
2. **for** l **from** 0 **to** $L-1$ **do**
2a. $\quad q_{l+1}:=P_{l+1,l}^2 q_l$;
2b. $\quad$ **for** j **from** 1 **to** k_{l+1} **do** FASCYCLE $(N_{l+1}^1 q_{l+1}=0)$ **enddo**;
2. **enddo**;
3. $q_L:=q_L+P_{L,L-1}^S(R_{L-1,L}^1 q_L-q_{L-1})$;
4. **for** n **from** 1 **to** N **do**
4a. $\quad r_L:=N_L^1(q_L)-N_L^2(q_L)$;
4b. $\quad$ **for** j **from** 1 **to** k_d **do** FASCYCLE $(N_L^1 q_L=r_L)$ **enddo**;
4. **enddo**

Stage 1 is an FAS-iteration process to obtain a first-order accurate initial estimate at level 0. Stage 2 is the nested iteration to obtain the solution of $N_h^1(q_h)=0$ up to truncation error accuracy. The prolongation $P_{l+1,l}^2$ is a bilinear interpolation procedure and, hence, accurate enough to retain the first-order accuracy on the finer mesh. Asymptotically, the discretization error for q_l is bounded by $Ch_l=\mathcal{O}(2^{L-l}h)$ for $h_L=h\to 0$. Now the theorem in section 2.5.4 shows that, for a fixed $k_l=k$ at all levels, the iteration error at level l is $\approx Ch_l\mu^k/(1-2\mu^k)$, where μ is an upper bound for the FAS-convergence factor. Therefore, to obtain a first-order accurate initial estimate for iteration (51b) it is not necessary to reduce the iteration error in q_l by a factor much smaller than $\mu^k\approx\frac{1}{3}$. This means that in stage 2, for all $l>0$, only a single FAS-step may be sufficient: $k=1$. Not being sure about the validity of the asymptotic assumption, in practice we set $k=2$. Stage 3 is a Richardson extrapolation step to (eventually) find a second-order initial estimate for iteration (51b). The prolongation $P_{L,L-1}^S$ and the restriction $R_{L-1,L}^1$ are piecewise bilinear interpolation over superboxes and averaging over cells, respectively, such that $R_{L-1,L}^1 P_{L,L-1}^S = I_{L-1}$ is the identity operator, and $P_{L,L-1}^S R_{L-1,L}^1$ a projection operator. With the asymptotic expansion for the error e in q_h as

$$q_h = R_h q + h^p R_h e + \mathcal{O}(h^{p+1}), \tag{52}$$

where q is the exact solution, for $p=1$ we obtain the second-order extrapolation

$$R_{2h}q = 2R_{2h,h}q_h - q_{2h} + \mathcal{O}(h^2). \tag{53}$$

We find the extrapolated value of q_h in stage 3 as the sum of (53) and $(I_L - P_{L,L-1}^S R_{L-1,L}^1)q_L \in \mathrm{Ker}(R_{2h})$. We notice that formally the approximation of q_L after stage 3 is still $\mathcal{O}(h)$, unless q_{L-1} is an $\mathcal{O}(h^2)$ approximation, and unless stage 2 can reduce the (smooth) error component $R_h e$ by a factor $\mathcal{O}(h)$. Nevertheless, in practice we see that already for small values of k, the Richardson extrapolation can reduce the error significantly [6]. Stage 4 finally, is the defect correction iteration (51b). If this iteration starts with a first-order initial approximation, for second-order accuracy it may be sufficient to take $N=1$. This necessitates an improvement of the error by a factor $\mathcal{O}(h)$ in the iteration 4b, i.e. we need $k_d=\mathcal{O}(\log(h))$. However, since the FAS-iteration is the expensive part of the computation in stage 4, for most purposes we take $k_d=1$ and a sufficiently large number for N.

ACKNOWLEDGEMENT

The authors want to acknowledge the cooperation of Stefan Spekreijse and Jurgen Rusch in this CFD research.

REFERENCES

1. Van Albada, G.D., Van Leer, B. and Roberts, W.W., 'A Comparative Study of Computational Methods in Cosmic Gasdynamics', Astronomy and Astrophysics, Vol. 108, p. 76-84, 1982.

2. Anderson, W.K., Thomas, J.L. and Van Leer, B., 'A Comparison of Finite Volume Flux Vector Splittings for the Euler Equations', AIAA Paper 85-0122, 1985.

3. Brandt, A., 'Multi-Level Adaptive Solutions to Boundary-Value Problems', Mathematics of Computation, Vol. 31, p. 333-390, 1977.

4. Colella, P. and Woodward, P.R., 'The Piecewise Parabolic Method (PPM) for Gas-Dynamical Simulations', Journal of Computational Physics, Vol. 54, p. 174-201, 1984.

5. Hackbusch, W., *'Multi-Grid Methods and Applications'*, Springer, Berlin, 1985.

6. Hemker, P.W., 'Defect Correction and Higher Order Schemes for the Multi Grid Solution of the Steady Euler Equations', Proceedings of the Second European Conference on Multigrid Methods, Cologne, Lecture Notes in Mathematics, Vol. 1228, p. 149-165, Springer, Berlin, 1986.

7. Hemker, P.W. and Koren, B., 'A Non-Linear Multigrid Method for the Steady Euler Equations', Proceedings of a GAMM Workshop on the Numerical Simulation of Compressible Euler Flows, Rocquencourt, Notes on Numerical Fluid Mechanics, Vol. 26, p. 175-196, Vieweg, Braunschweig, 1989.

8. Hemker, P.W., Koren, B. and Spekreijse, S.P., 'A Nonlinear Multigrid Method for the Efficient Solution of the Steady Euler Equations', Proceedings of the 10-th International Conference on Numerical Methods in Fluid Dynamics, Beijing, Lecture Notes in Physics, Vol. 264, p. 308-313, Springer, Berlin, 1986.

9. Hemker, P.W. and Spekreijse, S.P., 'Multigrid Solution of the Steady Euler Equations', Proceedings of Advances in Multi-Grid Methods, Oberwolfach, Notes on Numerical Fluid Mechanics, Vol. 11, p. 33-44, Springer, Berlin, 1985.

10. Hemker, P.W. and Spekreijse, S.P., 'Multiple Grid and Osher's Scheme for the Efficient Solution of the Steady Euler Equations', Applied Numerical Mathematics, Vol. 2, p. 475-493, 1986.

11. Jameson, A., Schmidt, W. and Turkel, E., 'Numerical Solution of the Euler Equations by Finite Volume Methods using Runge-Kutta Time Stepping Schemes', AIAA Paper 81-1259, 1981.

Jespersen, D.C., 'Design and Implementation of a Multigrid Code for the Euler Equations', Applied Mathematics and Computation, Vol. 13, p. 357-374, 1983.

Kettler, R., 'Analysis and Comparison of Relaxation Schemes in Robust Multigrid and Preconditioned Conjugate Gradient Methods', Proceedings of the First European Conference on Multigrid Methods, Cologne, Lecture Notes in Mathematics, Vol. 960, p. 502-534, Springer, Berlin, 1982.

Koren, B., 'Defect Correction and Multigrid for an Efficient and Accurate Computation of Airfoil Flows', Journal of Computational Physics, Vol. 77, p. 183-206, 1988.

Koren, B., 'Euler Flow Solutions for Transonic Shock Wave - Boundary Layer Interaction', International Journal for Numerical Methods in Fluids, Vol. 9, p. 59-73, 1989.

Koren, B., 'Multigrid and Defect Correction for the Steady Navier-Stokes Equations', Journal of Computational Physics, Vol. 87, p. 25-46, 1990.

Koren, B., 'Upwind Discretization of the Steady Navier-Stokes Equations', International Journal for Numerical Methods in Fluids, to appear, 1990.

Koren, B., 'Robustness Improvement of Point Gauss-Seidel Relaxation for Steady, Hypersonic Flow Computations', Note NM-N8805, Centre for Mathematics and Computer Science, Amsterdam, 1988.

Koren, B., *'Multigrid and Defect Correction for the Steady Navier-Stokes Equations, Application to Aerodynamics'*, CWI Tract, Centre for Mathematics and Computer Science, Amsterdam, 1990.

Koren, B. and Hemker, P.W., 'Damped, Direction-Dependent Multigrid for Hypersonic Flow Computations', Report NM-R8922, Centre for Mathematics and Computer Science, Amsterdam, 1989.

Lax, P.D., 'Hyperbolic Systems of Conservation Laws II', Communications on Pure and Applied Mathematics, Vol. 10, p. 537-566, 1957.

Van Leer, B., 'Flux-Vector Splitting for the Euler Equations', Proceedings of the Eighth International Conference on Numerical Methods in Fluid Dynamics, Aachen, Lecture Notes in Physics, Vol. 170, p. 507-512, Springer, Berlin, 1982.

23. Van Leer, B., 'Upwind-Difference Methods for Aerodynamic Problems governed by the Euler Equations', Proceedings of the 15-th AMS-SIAM Summer Seminar on Applied Mathematics, Scripps Institution of Oceanography, Lectures in Applied Mathematics, Vol. 22, Part 2, p. 327-336, American Mathematical Society, Providence, RI, 1985.

24. Mulder, W.A., 'Multigrid Relaxation for the Euler Equations', Journal of Computational Physics, Vol. 60, p. 235-252, 1985.

25. Mulder, W.A. and Van Leer, B., 'Experiments with Implicit Upwind Methods for the Euler Equations', Journal of Computational Physics, Vol. 59, p. 232-246, 1985.

26. Osher, S. and Chakravarthy, S.R., 'Upwind Schemes and Boundary Conditions with Applications to Euler Equations in General Geometries', Journal of Computational Physics, Vol. 50, p. 447-481, 1983.

27. Osher, S. and Chakravarthy, S.R., 'High Resolution Schemes and the Entropy Condition', SIAM Journal on Numerical Analysis, Vol. 21, p. 955-984, 1984.

28. Osher, S. and Solomon, F., 'Upwind Difference Schemes for Hyperbolic Systems of Conservation Laws', Mathematics of Computation, Vol. 38, p. 339-374, 1982.

29. Roe, P.L., 'Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes', Journal of Computational Physics, Vol. 43, p. 357-372, 1981.

30. Smoller, J. *'Shock Waves and Reaction-Diffusion Equations'*, Springer, Berlin, 1983.

31. Spekreijse, S.P., 'Multigrid Solution of Monotone Second-Order Discretizations of Hyperbolic Conservation Laws', Mathematics of Computation, Vol. 49, p. 135-155, 1987.

32. Spekreijse, S.P., *'Multigrid Solution of the Steady Euler Equations'*, CWI Tract 46, Centre for Mathematics and Computer Science, Amsterdam, 1988.

33. Steger, J.L., 'A Preliminary Study of Relaxation Methods for the Inviscid Conservative Gasdynamics Equations using Flux Splitting', NASA-CR-3415, 1981.

34. Steger, J.L. and Warming, R.F., 'Flux Vector Splitting of the Inviscid Gasdynamic Equations with Application to Finite-Difference Methods', Journal of Computational Physics, Vol. 40, p. 263-293, 1981.

35. Sweby, P.K., 'High Resolution Schemes using Flux Limiters for Hyperbolic Conservation Laws', SIAM Journal on Numerical Analysis, Vol. 21, p. 995-1011, 1984.