1991

H.T.M. van der Maarel

Adaptive multigrid for the steady Euler equations

Department of Numerical Mathematics Report NM-R9122 December

CWI, nationaal instituut voor onderzoek op het gebied van wiskunde en informatica

CWI is the research institute of the Stichting Mathematisch Centrum, which was founded on February 11, 1946, as a non-profit institution aiming at the promotion of mathematics, computer science, and their applications. It is sponsored by the Dutch Government through the Netherlands organization for scientific research (NWO).

Adaptive Multigrid for the Steady Euler Equations

H.T.M. van der Maarel
CWI
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

Abstract

We present a method to solve the steady Euler equations by using an upwind finite-volume discretization on a locally refined grid and multigrid convergence acceleration. The locally refined grid consists of grids in a locally nested sequence. We describe the discretization with emphasis on grid interfaces and accuracy. We describe the multigrid algorithm and give some computational examples.

1980 Mathematics Subject Classification: 35L65, 65N30.

Key Words and Phrases: Euler equations, finite-volume discretization, multigrid, defect correction, adaptive local mesh refinement.

Note: This research was performed as part of a Brite/Euram Area 5 project, under Contract No. AERO-0003C.

1 Introduction

Multigrid is known to be a very efficient method for solving boundary value problems, governed by partial differential equations. It has also proved to be well-suited for the Euler equations of fluid dynamics as shown by various researchers. To further improve the efficiency of fluid dynamics computations we introduce a multigrid method involving grids on finer and finer levels which do not cover the complete computational domain and which are constructed during the computation. This procedure results in a composite grid with locally refined regions. Examples of this approach for elliptic problems have already been given through MLAT [1] and FAC [5].

In our approach the refinements of the grid on some level form the grids on the next finer levels. The generation of local refinements results in a number of levels with grids of high grid point density only covering a part of the computational domain.

The set of equations resulting from the first-order consistent finite-volume discretization on the composite grid is solved by *nonlinear multigrid* iteration (FAS). The equations from the higher-order discretization are solved by *defect correction* iteration, where the first-order discretization is used as the less accurate operator and the multigrid method is used to invert this operator. A computation is started on some basic grid. After a number of nonlinear multigrid iterations a new level of refinement is generated. The initial solution in all newly generated cells is found from an interpolation on the next coarser grid. Then, the multigrid iteration is started with the new level as top level. After a few cycles, again cells may be refined both on the top level and on lower levels. Further, cells that seem to have become superfluous may be discarded (coarsening). These steps may be repeated until a satisfactory solution is obtained.

Report NM-R9122
ISSN 0169-0388
CWI
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

2 Geometric structure

We consider the two-dimensional, steady Euler equations defined on a domain $\Omega \subset \mathbb{R}^2$ and boundary conditions defined on the boundary $\partial\Omega$ of Ω. For the discretization the domain Ω is approximated by a regular partitioning of disjoint *quadrilateral cells*.

We consider grids on different levels of refinement, on each level employing a regular partitioning of one or more subdomains of Ω. The grid on level l is denoted by Ω^l, $l \in \{0, 1, 2, \ldots, L\}$. The coarsest grid is Ω^0 and completely covers Ω. Each cell on the grid Ω^l on level $l > 0$ is a member of a disjoint division of a cell on Ω^{l-1} into a set of 2×2 smaller cells. The cells on Ω^{l-1} and the cells on Ω^l are coexistent, i.e. the coarse cells overlaid by the fine cell are not removed when the fine cells are generated. Except for the cells on Ω^0 each cell is one of the four descendants (*kids*) of a cell on a coarser (lower) level (*parent*). All cells in the geometric structure are related to each other through a so-called *quad-tree* structure.

Each cell on Ω^l has either *none* or *one* neighboring cell at each side. The location of each cell on Ω^l is determined by a set of coordinates $(i, j) \in \mathbb{Z}^2$. This integer coordinate system is chosen such that for Ω^0 the smallest coordinate in each direction is zero. A kid of a cell $\Omega^l_{i,j}$ is either $\Omega^{l+1}_{2i,2j}$, $\Omega^{l+1}_{2i+1,2j}$, $\Omega^{l+1}_{2i,2j+1}$ or $\Omega^{l+1}_{2i+1,2j+1}$.

The edge $\partial\Omega^l_{i,j}$ of a cell $\Omega^l_{i,j}$ consists of four rectilinear sides, which we identify through their relative locations by $\partial\Omega^l_{i,j,k}$, $k \in \{N, E, S, W\}$, where $\{N, E, S, W\}$ is the set of wind directions.

The grid Ω^l consists of a part Ω^l_f with cells that have been refined (kids residing on level $l+1$), and a part Ω^l_c with cells that have not been refined (have no kids). The grids Ω^l_f and Ω^{l+1} cover the same part of Ω. The set of all non-refined cells is called the *composite* grid Ω_c.

The part of the boundary of Ω^l which does not coincide with a part of $\partial\Omega$ is called a *green* boundary and denoted by $\partial\Omega^l_g$. The part that does coincide with a part of $\partial\Omega$ is denoted by $\partial\Omega^l$. In adaptive multigrid computations we are interested in the solution on the composite grid which is constructed during the computation.

3 Discretization

3.1 The set of discrete equations

The two-dimensional, steady Euler equations are discretized on a sequence of grids $\{\Omega^l\}_0^L$. By the rotational invariance of the Euler equations they can be written as

$$\oint_{\partial\Omega^*} T^{-1} f(Tq)\, ds = 0, \tag{3.1}$$

for any $\Omega^* \subset \Omega$ with boundary $\partial\Omega^*$. T is a rotation matrix which transforms q to a coordinate system matching $\partial\Omega^*$. A discretization of the equations on a composite grid is obtained by demanding a discrete version of (3.1) to hold for each finite-volume $\Omega^l_{i,j} \subset \Omega_c$. The mean value of the flux $f(q(x,y))$ across the kth cell face $\partial\Omega^l_{i,j,k}$ is denoted by

$$f^l_{i,j,k} = \frac{1}{s^l_{i,j,k}} \int_{\partial\Omega^l_{i,j,k}} T^{-1} f(Tq)\, ds, \tag{3.2}$$

where $s^l_{i,j,k}$ is the length of $\partial\Omega^l_{i,j,k}$. In the upwind discretization the flux $f^l_{i,j,k}$ is approximated by Osher's approximate Riemann solver (Godunov-type scheme), in the P-variant [3], denoted by $F(q^L, q^R)$. The states q^L and q^R are the left and right states of the Riemann problem, and are determined in a *projection* stage preceding the actual flux evaluation. These cell face states are determined by interpolations with a bias to the left and right side of a cell face respectively.

In the finite-volume approach we consider $q^l_{i,j}$ in a cell $\Omega^l_{i,j}$ to be an approximation of the mean value of $q(x,y)$ over $\Omega^l_{i,j}$. The mean flux $f^l_{i,j,k}$ is approximated by the numerical flux $F^l_{i,j,k}$, defined by

$$F^l_{i,j,k}((q^L)^l_{i,j,k}, (q^R)^l_{i,j,k}) = (T^l_{i,j,k})^{-1} F(T^l_{i,j,k}(q^L)^l_{i,j,k}, T^l_{i,j,k}(q^R)^l_{i,j,k}). \tag{3.3}$$

Here $T^l_{i,j,k}$ is the rotation matrix on $\partial\Omega^l_{i,j,k}$ and $(q^L)^l_{i,j,k}$ and $(q^R)^l_{i,j,k}$ are the left and right Riemann states respectively, for $\partial\Omega^l_{i,j,k}$. The set of discrete equations on the composite grid is given by

$$\sum_{k=N,E,S,W} F^l_{i,j,k}((q^L)^l_{i,j,k}, (q^R)^l_{i,j,k})\, s^l_{i,j,k} = 0, \quad \forall\Omega^l_{i,j} \subset \Omega_c. \tag{3.4}$$

The set of equations (3.4) is under-determined when there is more than one grid level in the composite grid. Additional equations are easily obtained by

$$q^l_{i,j} = \frac{1}{4}\left(q^{l+1}_{2i,2j} + q^{l+1}_{2i+1,2j} + q^{l+1}_{2i,2j+1} + q^{l+1}_{2i+1,2j+1}\right), \quad \forall\Omega^l_{i,j} \subset \Omega^l_f. \tag{3.5}$$

Equations (3.4) together with (3.5) define the set of equations to be satisfied by an approximate solution on a composite grid.

3.2 The projection stage

3.2.1 Projection I

The projection stage for a first-order accurate discretization is based on a piecewise constant interpolation. Consider for example the eastern face of a cell, where its eastern neighbor exists. As usual in first-order Godunov-type schemes, for this situation the left and right states for $\partial\Omega^l_{i,j,E} \not\subset \partial\Omega^l_g \cup \partial\Omega^l$ are taken as

$$(q^L)^l_{i,j,E} = q^l_{i,j}, \tag{3.6a}$$
$$(q^R)^l_{i,j,E} = q^l_{i+1,j}. \tag{3.6b}$$

Next we compute the flux across $\partial\Omega^{l+1}_{2i+1,2j+1,E}$, the eastern cell face of the north-eastern kid of $\Omega^l_{i,j}$ and where $\Omega^{l+1}_{2i+2,2j+1}$ does not exist, but $\Omega^l_{i+1,j}$ does. We introduce a virtual state $w^{l+1}_{2i+2,2j+1}$, associated with the location where $\Omega^{l+1}_{2i+2,2j+1}$ would be and determined by interpolation on the next coarser grids. Applying (3.6) for $\partial\Omega^{l+1}_{2i+1,2j+1,E}$ using the virtual state gives

$$(q^L)^{l+1}_{2i+1,2j+1,E} = q^{l+1}_{2i+1,2j+1}, \tag{3.7a}$$
$$(q^R)^{l+1}_{2i+1,2j+1,E} = w^{l+1}_{2i+2,2j+1}. \tag{3.7b}$$

For the other cell faces ($k = N, S, W$), formulae similar to (3.6)–(3.7) are used. We will call (3.6) and (3.7) projection I.

3.2.2 Projection II

For higher-order accuracy the projection stage is based on higher-order interpolation of the states. This interpolation can be done with a limiter to suppress spurious wiggling of the solution (as in e.g. [6]), or without a limiter such as the piecewise linear κ-schemes [9], (as in e.g. [2]). Again, consider the eastern cell face of $\Omega^l_{i,j}$. For the situation not involving a green boundary or a domain boundary, a standard κ-scheme or a limiter scheme may be applied

directly. When there is no green boundary involved, in both a κ-scheme or a limiter scheme a projection depends on the states in at most three cells in a row, denoted by

$$(q^L)^l_{i,j,E} = C(q^l_{i-1,j},\, q^l_{i,j},\, q^l_{i+1,j}), \tag{3.8a}$$

$$(q^R)^l_{i,j,E} = C(q^l_{i+2,j},\, q^l_{i+1,j},\, q^l_{i,j}), \tag{3.8b}$$

where C denotes either a κ-scheme or a limiter scheme.

Next consider the situation where a green boundary is involved. This is e.g. the case for $\partial\Omega^{l+1}_{2i,2j+1,E}$ and $\partial\Omega^{l+1}_{2i+1,2j+1,E}$, when $\Omega^{l+1}_{2i+2,2j+1}$ and $\Omega^{l+1}_{2i+3,2j+1}$ do not exist and $\Omega^l_{i+1,j}$ does. We proceed in the way of section (3.2.1) and construct virtual states $w^{l+1}_{2i+2,2j+1}$ and $w^{l+1}_{2i+3,2j+1}$, associated with the locations where $\Omega^{l+1}_{2i+2,2j+1}$ and $\Omega^{l+1}_{2i+3,2j+1}$ would be. Again they are found from an interpolation involving q^n, $n \le l$. The left and right states are found in the way of (3.8). For $\partial\Omega^{l+1}_{2i+1,2j+1,E} \subset \partial\Omega^{l+1}_g$ we have

$$(q^R)^{l+1}_{2i,2j+1,E} = C\left(w^{l+1}_{2i+2,2j+1},\, q^{l+1}_{2i+1,2j+1},\, q^{l+1}_{2i,2j+1}\right), \tag{3.9a}$$

$$(q^L)^{l+1}_{2i+1,2j+1,E} = C\left(q^{l+1}_{2i,2j+1},\, q^{l+1}_{2i+1,2j+1},\, w^{l+1}_{2i+2,2j+1}\right), \tag{3.9b}$$

$$(q^R)^{l+1}_{2i+1,2j+1,E} = C\left(w^{l+1}_{2i+3,2j+1},\, w^{l+1}_{2i+2,2j+1},\, q^{l+1}_{2i+1,2j+1}\right). \tag{3.9c}$$

For the flux computation for the other cell faces ($k = N, S, W$), formulae similar to (3.8) and (3.9) are used. We will call (3.8) and (3.9) projection II.

4 Accuracy

4.1 Virtual states and consistency

Let the restriction operator $\overline{R}^l_\infty$ be defined as the operator delivering the mean value of its argument over a cell and the restriction operator R^l_∞ delivering the integral of its argument over the cell, for each cell on level l. From here on we will omit the arguments of the numerical flux and, unless other arguments are given, we will denote by $F^l_{i,j,k}$ the flux $F^l_{i,j,k}((q^L)^l_{i,j,k}, (q^R)^l_{i,j,k})$. We assume that we have a smooth composite grid and across the green boundaries a difference in levels of only one. Then with the projections I and II a numerical flux $F^l_{i,j,k}$ depends on q^l and q^{l-1} only. Denoting the corresponding system of equations by $N^l(q^l,\, q^{l-1}) = 0$, we have for all $\Omega^l_{i,j} \subset \Omega_c$

$$\left\{N^l(q^{l-1}, q^l)\right\}^l_{i,j} = \sum_{k=N,E,S,W} F^l_{i,j,k}\, s^l_{i,j,k}. \tag{4.1}$$

Moreover, let $N(q) = 0$ be the set of equations when (3.1) is applied to all $\Omega^l_{i,j} \subset \Omega_c$, i.e.

$$\left\{N(q)\right\}^l_{i,j} = \sum_{k=N,E,S,W} f^l_{i,j,k}\, s^l_{i,j,k}. \tag{4.2}$$

In a finite-volume discretization the local truncation error $\tau^l(q)$ of a discretization of $N(q)$ may be defined by

$$\left\{\tau^l(q)\right\}^l_{i,j} = \frac{1}{A^l_{i,j}} \left\{N^l(\overline{R}^{l-1}_\infty q,\, \overline{R}^l_\infty q) - N(q)\right\}^l_{i,j}, \quad \forall \Omega^l_{i,j} \subset \Omega_c, \tag{4.3}$$

for any admissible state function $q(x, y)$ and where $A^l_{i,j}$ is the area of $\Omega^l_{i,j}$. By definition, a pth-order consistent discretization is obtained, if for $l \to \infty$ the local truncation error satisfies

$$\left\{\tau^l(q)\right\}^l_{i,j} = \mathcal{O}(h^p_l), \quad \forall \Omega^l_{i,j} \subset \Omega_c, \tag{4.4}$$

where h_l satisfies $A^l_{i,j} = \mathcal{O}(h_l^2)$. On a smooth enough grid which converges to a uniform grid as $l \to \infty$, the projections I and II for the discretization in a cell where *no* green boundaries are involved, yield a first-order and second-order local truncation error respectively. With the $\kappa = 1/3$-scheme, accuracy of order higher than two may be found in practice ([4]).

The virtual states for the projections I and II involving a green boundary are determined by $\mathcal{O}(h_l^p)$ accurate interpolation, with $p = 1$ (e.g. piecewise constant) for projection I and $p = 2$ (e.g. bilinear) for projection II. The local truncation error of a discrete equation involving a green boundary is then only $\mathcal{O}(h_l^{p-1})$. However, the discretization using this interpolation can be shown to satisfy a weaker form of consistency. On a smooth enough grid an $\mathcal{O}(h_l^p)$ accurate interpolation to compute a virtual state leads to an $\mathcal{O}(h_l^p)$ accurate computation of the mean flux across a cell face, when projection I is used for $p = 1$ and projection II for $p = 2$. Therefore, for all cells $\Omega^{l+n}_{2^n i+r,2^n j+s}$ which are the descendants of a cell $\Omega^l_{i,j} \subset \Omega_c$ after n times refining $\Omega^l_{i,j}$, the equations satisfy the weaker form of consistency:

$$\frac{1}{A^l_{i,j}} \sum_{r,s=0}^{2^n-1} \{\tau^{l+n}(q)\}^{l+n}_{2^n i+r,2^n j+s} A^{l+n}_{2^n i+r,2^n j+s} = \mathcal{O}(h^p_{l+n}). \tag{4.5}$$

This is asymptotically 2^{-np} times the local truncation error on $\Omega^l_{i,j}$ when not refined.

We conducted an experiment to show that the projections I and II and the virtual states determined by piecewise constant interpolation for projection I and bilinear interpolation for projection II, give first-order and higher-order accurate solutions respectively, in the presence of a green boundary. We consider the subsonic flow through a channel with a 'sine-form' bump, defined by the shape $y_w(x)$ of the lower wall, given by

$$y_w(x) = \begin{cases} 0, & x \le 1, \quad x \ge 3, \\ 0.1(1 - \cos((x-1)\pi)), & 1 < x < 3. \end{cases} \tag{4.6}$$

The upper wall is at $y = 1$ and the computational domain extends from $x = 0$ to $x = 4$. The boundary conditions are 'over-specified'. On both $x = 0$ (inlet) and $x = 4$ (outlet) we specify the Mach number: $M = 0.5$, the density: $\rho = 1$ and the velocity: $u = 1$, $v = 0$. Along the upper and lower wall non-permeability is imposed: $un_x + vn_y = 0$. For this problem the exact solution has constant entropy, i.e. constant $p\rho^{-\gamma}$, where γ is the ratio of specific heats. In Fig. 1 we show iso-line plots of the entropy error, obtained on the grids shown. In projection II the $\kappa = 1/3$-scheme is used. The local truncation error of an equation involving a green boundary satisfies (4.5) end the solutions found are first-order and higher-order accurate for projections I and II respectively.

If we want to use an estimate of the local truncation error of the equations involving a green boundary in the refinement criterion, we need to be sure that the truncation error is $\mathcal{O}(h_l^p)$. This is obtained when the projections themselves are pth-order accurate interpolations *and* when they satisfy

$$(q^L)^l_{i,j,E} - (q^L)^l_{i,j,W} = q^l_{i,j,E} - q^l_{i,j,W} + \mathcal{O}(h_l^{p+1}), \tag{4.7a}$$

$$(q^R)^l_{i,j,E} - (q^R)^l_{i,j,W} = q^l_{i,j,E} - q^l_{i,j,W} + \mathcal{O}(h_l^{p+1}), \tag{4.7b}$$

and similarly for E and W replaced by N and S. The state $q^l_{i,j,k}$ denotes the mean value of $q(x,y)$ along $\partial\Omega^l_{i,j,k}$, and the left and right states are determined by the projections. When no green boundaries or physical boundaries are involved (4.7) are satisfied both by projection I ($p = 1$) and II ($p = 2$).

When a green boundary is involved (4.7) can be satisfied by computing the virtual states with an $\mathcal{O}(h_l^{p+1})$ accurate interpolation. Instead of using a piecewise constant interpolation in

projection I, we could use a bilinear interpolation (as in the example used with projection II) or the more compact interpolation

$$w^{l+1}_{2i+2,2j+1} = \frac{3}{4}q^l_{i+1,j} + \frac{1}{4}q^l_{i,j+1}, \tag{4.8}$$

and similar formulae for other virtual states. By Taylor series expansion this can be shown to be second-order accurate on a smooth enough grid ([8]). We favor the more compact interpolation (4.8). For projection II we use the third-order accurate interpolation given by

$$w^{l+1}_{2i+2,2j+1} = \frac{1}{16}\left(q^l_{i,j} + 17q^l_{i+1,j} - 2q^l_{i+2,j} + q^l_{i,j+1} + q^l_{i+1,j+1} - 2q^l_{i+1,j-1}\right), \tag{4.9}$$

and similar formulae for other virtual states.

Computation of the virtual states with this $(p+1)$st-order accurate interpolation yields an $\mathcal{O}(h_l^p)$ relative truncation error for the equations involving green boundaries. Note that (4.8) and (4.9) are symmetric with respect to the line through the centers of $\Omega^l_{i+1,j}$ and $\Omega^l_{i,j+1}$, and give the same virtual state when we are concerned with $\partial\Omega^{l+1}_{2i+2,2j+2,S} \subset \partial\Omega^{l+1}_g$.

5 Multigrid and iterative defect correction

The set of discrete equations in (3.4) and (3.5) is solved by multigrid iteration, applied to the set of first-order discretized equations on the sequence of grids. The method we apply is the nonlinear multigrid scheme FAS, as described in [3]. The equations for higher-order accuracy are solved by iterative defect correction as applied in [2].

In order to use multigrid we must specify the grid transfer operators. Two restriction operators, $\overline{R}^l_{l+1}$ and R^l_{l+1}, and the prolongation operator P^{l+1}_l are defined such that we obtain a sequence of nested discretizations on the locally refined grids, and such that upon convergence (3.5) is satisfied. The restriction operator for the solution is defined by $\overline{R}^l_{l+1}$, being the operator taking for a cell on Ω^l the *arithmetic mean* of the solution of the four kids on Ω^{l+1} of the same parent. The restriction operator for the right-hand side, R^l_{l+1}, takes the *sum* of the components of the four kids of the same parent. The prolongation operator, P^{l+1}_l of (a correction of) the solution, is defined to be the piecewise constant interpolation from data on level l, transferred to level $l+1$. We denote the operator N^l on Ω^l with projection I by $N^l_{\rm I}$ and with projection II by $N^l_{\rm II}$. If we use (4.1) for all $\Omega^l_{i,j} \subset \Omega^l_f$, where N^l is the operator $N^l_{\rm I}$, it can be shown that the coarse grid operator restricted to Ω^l_f is a Galerkin approximation of the fine grid operator on Ω^{l+1} (see [3]).

In the nonlinear multigrid algorithm the following equations are used to obtain a solution on the composite grid, or to find the coarse grid correction:

$$N^l_{\rm I}(q^{l-1}, q^l) = r^l, \tag{5.1a}$$

$$r^l = \begin{cases} 0, & \text{on } \Omega^l_c \\ N^l_{\rm I}(q^{l-1}, \overline{R}^l_{l+1}q^{l+1}) - R^l_{l+1}\left(N^{l+1}_{\rm I}(q^l, q^{l+1}) - r^{l+1}\right), & \text{on } \Omega^l_f. \end{cases} \tag{5.1b}$$

Upon convergence of the FAS-scheme, the solution of the set (5.1) satisfies (3.4) and (3.5).

As the smoother, on each level collective, symmetric point Gauss-Seidel relaxation is used, where in each cell $\Omega^l_{i,j}$ visited, the state $q^l_{i,j}$ is updated by Newton iteration applied to the local system $\{N^l_{\rm I}(q^{l-1}, q^l)\}^l_{i,j} = r^l_{i,j}$. The residual tolerance for this Newton iteration is taken such that in all but exceptional cases only one or two iterations are performed. The cells on a level are visited in the usual lexicographical order. After a relaxation sweep has been made, another

sweep is made in the reversed direction (symmetric). This smoother has been shown to be very efficient [4].

The higher-order discretized equations are solved, using an iterative defect correction approach [2]. The set of higher-order discretized equations on Ω_c^l is given by

$$N_{\rm II}^l(q^{l-1}, q^l) = 0. \tag{5.2}$$

The defect correction process solves (5.2) by iteratively solving (5.1a) through the FAS-scheme, where the right-hand side r^l is initially taken zero, but updated during the computation and depends on the defect of the higher-order equations through

$$r^l = \begin{cases} N_{\rm I}^l(q^{l-1}, q^l) - N_{\rm II}^l(q^{l-1}, q^l), & \text{on } \Omega_c^l, \\ N_{\rm I}^l(q^{l-1}, \overline{R}_{l+1}^l q^{l+1}) - R_{l+1}^l \left(N_{\rm I}^{l+1}(q^l, q^{l+1}) - r^{l+1}\right), & \text{on } \Omega_f^l. \end{cases} \tag{5.3}$$

Upon convergence, the solution satisfies (5.2) and (3.5).

6 Results

6.1 Shock-reflection

In this section we show some results of the adaptive multigrid method. First, we consider the shock-reflection problem. The domain is given by $\overline{\Omega}\{(x,y)|0 \le x \le 4,\ 0 \le y \le 1\}$. The boundary conditions are such that $x = 0$ is an inflow boundary with mach number $M = 2.9$ and an incoming shock through $(x, y) = (0, 1)$, which has an angle with the positive x direction of 29°. We use the first-order discretization $N_{\rm I}^l$ from section 3, with piecewise constant interpolation for determining the virtual states . In the refinement criterion the first undivided difference of the density is used. The threshold is taken to be 0.05, which showed to give a broad band of refinements along the shock. The unrefinement threshold is taken to be 0.024, giving a small safety margin with respect to the value 0.025. The number of multigrid cycles (V-cycles) between each refinement cycle is only one.

The adaptive grid uses only 7554 cells on the composite grid and a total number of 10068 cells. The non-adaptive grid uses 49152 cells on the finest grid and a total number of 65532 cells. The finest cells of the composite grid are of the same size as the cells of the non-adaptive grid. In both computations the same coarsest grid of 6×2 cells is used.

Fig. 2a and 2b show iso-plots of the Mach number on both the adaptive and non-adaptive grid respectively. In Fig. 2c the adaptively generated grid is shown and in Fig. 2d the convergence histories of both the adaptive method and the method using uniform grids. The amount of work in a multigrid V-cycle is proportional to the total number of cells present on the composite grid. The residual shown in Fig. 2d is the mean of the discrete L_1-norms on Ω_c of the residual of the four discrete equations, defined for cells on the composite grid. The amount of work for the adaptive method is about ten times less than the amount of work needed when grids fully covering the computational domain are used, while virtually the same solution is obtained.

6.2 Transonic airfoil flow

Here we consider the transonic flow around the NACA0012-airfoil with $M_\infty = 0.8$ and an angle of attack of $\alpha = 1.25°$. The computational domain extends to about 100 chords to all sides.

We use the second-order discretization $N_{\rm II}^l$ that is also second-order accurate when a green boundary is involved. In $N_{\rm II}^l$ the Van Albada limiter [7] is used, since spurious wiggles in the solution are expected when a non-limited scheme is used. In the refinement criterion we use the first undivided differences of the density. In each cell we use the gradient of the density in flow

direction and the gradient perpendicular to the flow direction, multiplied by the square root of the area of the cell. Two thresholds are used, one for each direction. This allows the algorithm not only to find a contact discontinuity, but it also prevents the algorithm from refining in the neighborhood of a shock only. Then, we not only get a good resolution of the shock, but also a good resolution of the expansion region, and this in turn is important for the accurate computation of the the lift and drag coefficients. The first undivided difference refines a cell, when in flow direction it is larger than 0.02, or when this difference in the direction perpendicular to the flow, is larger than 0.004.

The Mach number distributions are shown in Fig. 3a and 3b for the adaptive and non-adaptive grid respectively. The computed lift and drag coefficient on the adaptively generated composite grid are $c_l = 0.3480$, $c_d = 0.0235$. On the non-adaptive grid we find $c_l = 0.3512$ and $c_d = 0.0235$. The difference between these values is less than 10% of the scatter found between reference results listed in [10]. This reference gives $c_l = 0.3632$ and $c_d = 0.0230$ obtained on a grid of 20480 cells, by Schmidt and Jameson. The number of cells on the composite grid (Fig. 3c) is 2995 and a total number of 3980 cells was used in the computation. The non-adaptive grid uses 10240 cells on the finest grid and a total number of cells of 13640. The convergence histories of both the adaptive and non-adaptive case are shown in Fig. 3d. The adaptive computation takes about three times less work than the computation on the non-adaptive grid.

7 Conclusions

We introduced an efficient method to perform Euler flow computations on a locally refined grid. We use an upwind finite-volume discretization on an adaptively generated composite grid. Both first-order and higher-order accurate discretizations are considered.

Across a green boundary a difference of only one level is used. The algorithm may be extended to more general composite grids. From this no gain in efficiency is expected, since it increases the number of possible grid configurations at a green boundary enormously.

On a smooth grid, the local truncation error in a cell bordering a green boundary can be $\mathcal{O}(h_l^{p-1})$ but, for a pth-order accurate discretization in the interior of the grid, the discretization will still satisfy a weaker form of pth-order consistency and the solution may be $\mathcal{O}(h_l^p)$ accurate, as shown by numerical experiments. By applying a $(p+1)$st-order accurate interpolation to find virtual states, a pth-order accurate discretization is obtained in cells where green boundaries are involved. This allows the use of an estimate of the local truncation error in the refinement criterion. However, results are not yet completely satisfactory and this technique is still under investigation at the moment.

Compared with non-adaptive computations, the adaptive method shows a decrease in number of local Newton iterations of a factor three to ten, while the solution is obtained with approximately the same accuracy. No thorough comparison has been made yet between the efficiencies of this method and a comparable method which uses a structured grid and which may be vectorized. Although the present method may not be vectorizable, we think that parallelization will be feasible and hence compensate for the loss in efficiency of possible overhead of the adaptive grid structure.

Acknowledgement

The author wishes to thank Prof. P. Hemker, Dr. B. Koren and Prof. P. Wesseling for their kind support.

References

[1] A. Brandt. Multi-level adaptive solutions to boundary-value problems. *Math. Comp.*, 31(138):333–390, 1977.

[2] P.W. Hemker. Defect correction and higher order schemes for the multi grid solution of the steady Euler equations. In W. Hackbusch and U. Trottenberg, editors, *Multigrid Methods II, Lecture Notes in Mathematics*, pages 149–165, Berlin, 1985. Springer-Verlag.

[3] P.W. Hemker and S.P. Spekreijse. Multiple grid and Osher's scheme for the efficient solution of the steady Euler equations. *Appl. Num. Math.*, 2:475–493, 1986.

[4] B. Koren. *Multigrid and Defect Correction for the Steady Navier-Stokes Equations, An Application to Aerodynamics.* Centre for Mathematics and Computer Science, Amsterdam, 1990. CWI-tract 74.

[5] S. McCormick. The Fast Adaptive Composite grid (FAC) method for elliptic equations. *Math. Comp.*, 46(174):439–456, 1986.

[6] P.K. Sweby. High resolution schemes using flux limiters for hyperbolic conservation laws. *SIAM J. Numer. Anal.*, 21:995–1011, 1984.

[7] G.D. van Albada, B. van Leer, and W.W. Roberts. A comparative study of computational methods in cosmic gas dynamics. *Astron. Astrophys.*, 108:76–84, 1982.

[8] H.T.M. Van der Maarel. A solution-adaptive multigrid method for the steady Euler equations. CWI report NM-R91XX, in preperation.

[9] B. van Leer. Upwind difference methods for aerodynamic problems governed by the Euler equations. In B.E. Engquist, S. Osher, and R.C.J. Sommerville, editors, *Large Scale Computations, part 2, Lectures in Applied Mathematics*, pages 327–336. American Mathematical Society, Providence, Rhode Island, 1985.

[10] H. Viviand. Numerical solutions of two-dimensional reference test cases. In *AGARD-AR-211.* AGARD, 1985.

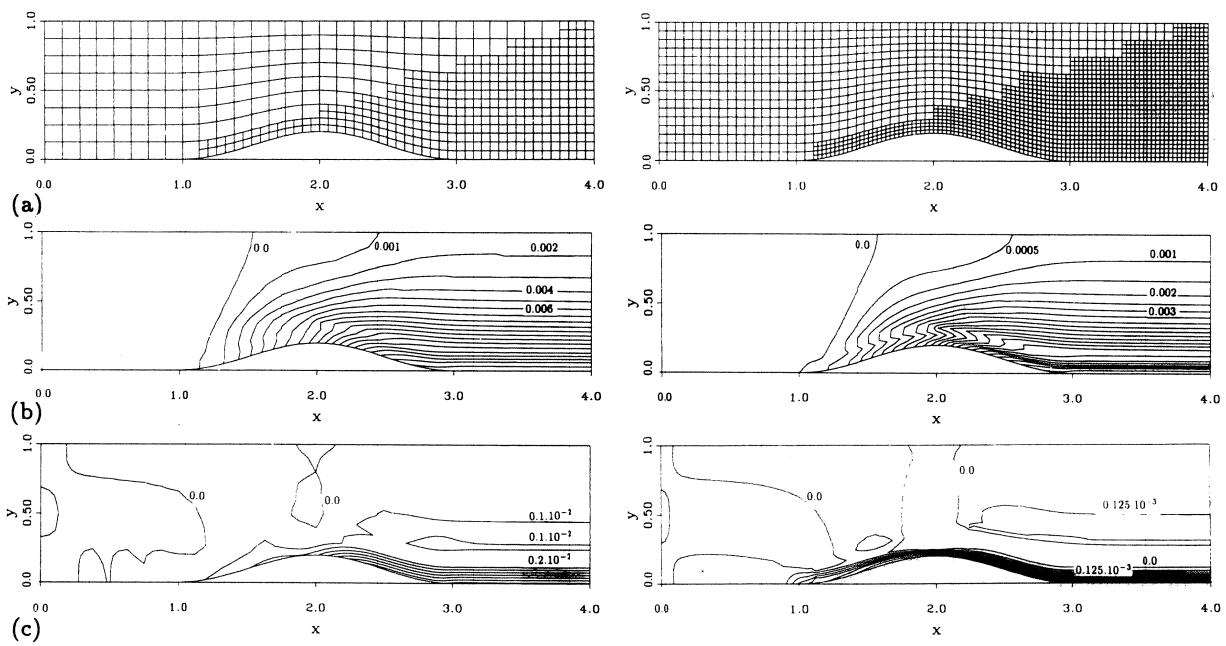

Figure 1: Global errors. (a)Composite grids with green boundary; (b)Error entropy, first-order discretization; (c)Error entropy, second-order discretization.

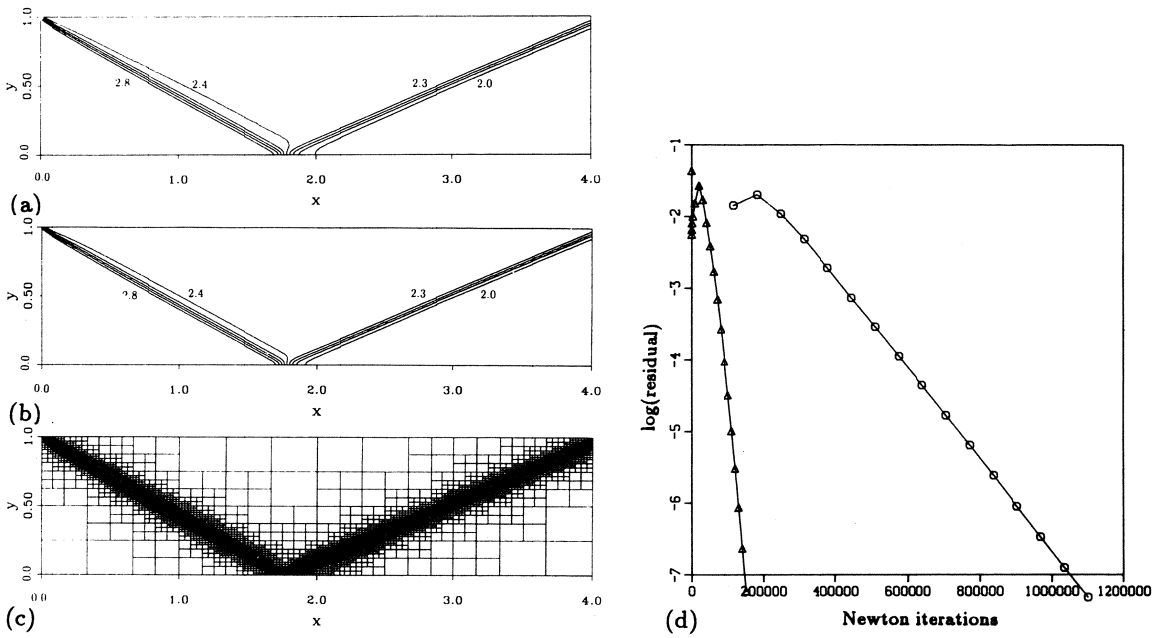

Figure 2: Shock-reflection problem. (a)M on adaptive grid, 7554 cells; (b)M on non-adaptive grid, 128×384 cells; (c)Adaptively generated composite grid; (d)Convergence history, ○:non-adaptive grid, △:adaptive grid.

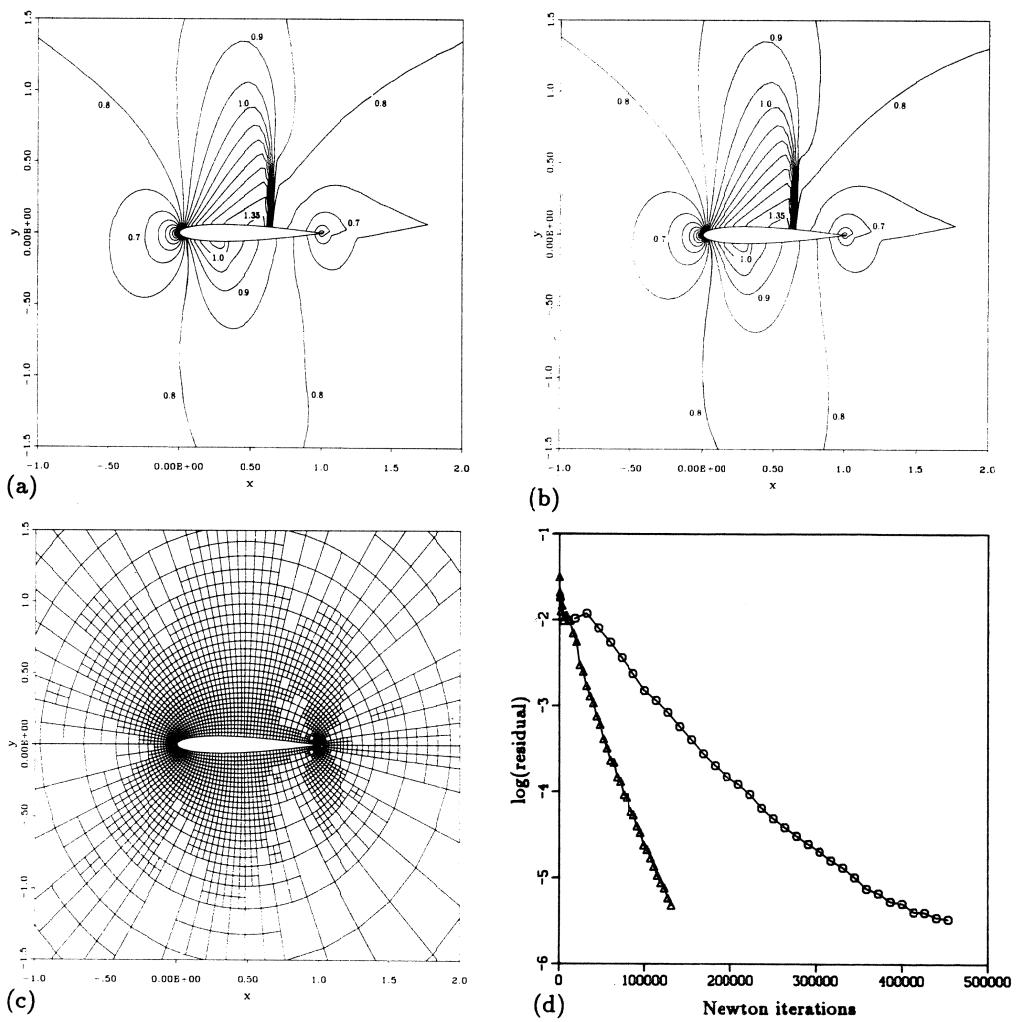

Figure 3: Transonic flow around NACA0012, $M_\infty = 0.8$, $\alpha = 1.25°$. (a)M on adaptive grid, 2995 cells; (b)M on non-adaptive grid, 80×128 cells; (c)Adaptively generated composite grid; (d)Convergence history, ○:non-adaptive grid, △:adaptive grid.