NUMERICAL EXPERIMENTS WITH MULTILEVEL SCHEMES FOR CONSERVATION LAWS

G. CHIAVASSA AND R. DONAT
Departament de Matematica Aplicada,
Universidad de Valencia,
46100 Burjassot, Valencia, Spain
Emails: guillaume.chiavassa@uv.es
donat@uv.es

Abstract. Main steps of a point-value multilevel algorithm are presented and numerical results for a two dimensional test case of gas dynamics are discussed in terms of quality and efficiency.

1. Introduction and general framework

We are concerned with solving the initial value problem for the two dimensional compressible Euler equations :

$$\left\{ \begin{array}{l} \partial_t \vec{U} + \vec{f}(\vec{U})_x + \vec{g}(\vec{U})_y = \vec{0} \\ \vec{U}(x,y,0) = \vec{U}_0(x,y) \end{array} \right. \tag{1}$$

with $\vec{U} = (\rho, m_x, m_y, E)^t$ and $\vec{f}$ and $\vec{g}$ the classical physical flux functions. The discretization procedure of Shu and Osher (Shu and Osher, 1989) is applied to (1) and the unknowns at each time step $t_n = n\delta t$ are the values of $\vec{U}$ on a Cartesian grid $\mathcal{G}^0 : \vec{U}^n_{ij} = \vec{U}(x_i, y_j, t_n)$. In the conservative discretized form of (1) appears the numerical divergence:

$$\mathcal{D}(\vec{U})_{ij} = \frac{\vec{F}_{i+1/2,j} - \vec{F}_{i-1/2,j}}{\delta x} + \frac{\vec{G}_{i,j+1/2} - \vec{G}_{i,j-1/2}}{\delta y} \tag{2}$$

where $\vec{F}_{i+1/2,j}$ and $\vec{G}_{i,j+1/2}$ are the numerical fluxes in the x and y directions defined from the functions $\vec{F}(\vec{U}_{i-k,j}, .., \vec{U}_{i+m,j})$ and $\vec{G}(\vec{U}_{i,j-k}, .., \vec{U}_{i,j+m})$. High order shock-capturing schemes are needed to evaluate these numerical fluxes in order to represent accurately the solution without unphysical

oscillations. In our experiments, we will use Marquina's scheme (Donat and Marquina, 1996) with a piecewise hyperbolic reconstruction procedure of order 3 (PHM). Such schemes are very efficient, but suffer from a large computational cost due in particular to their strongly nonlinear structure. In order to reduce this cost, Harten proposed (Harten, 1995) to use a multiresolution representation of the flow to analyze its local smoothness and to replace the expensive flux computation in the smooth zones by a cheaper procedure. These so called *multilevel schemes* have been successfully tested for one dimensional Euler equations in (Harten, 1995) and for two dimensional scalar equations in (Bihari and Harten, 1997). In (Sjögreen, 1995), Sjörgreen implements a "dimension by dimension" multilevel algorithm for a bi-dimensional compressible Euler computation. This algorithm uses only one-dimensional procedures which leads to an easier implementation, but tends to be less efficient, in our experimentations, than a fully two-dimensional one (Chiavassa and Donat, 1999).
In this paper, we present a two dimensional algorithm developed in the point values context whereas the previous ones have always been derived in the cell-average framework. A complete description of the multilevevel strategy can be found in (Chiavassa and Donat, 1999), and only the main steps are presented in section 2. In section 3, this algorithm is applied to a classical CFD problem, and the results are discussed in terms of quality and efficiency.

2. Description of the algorithm

The three principal ingredients of the multilevel algorithm are the multiresolution transform, the thresholding procedure and the multilevel computation of the numerical divergence.

2.1. MULTIRESOLUTION TRANSFORM OF DATA

This transform has been initially presented by Harten in (Harten, 1996) and the reader is referred to this paper for details. One first defines a set of embedded grids $\{\mathcal{G}^l,\ l = 1, .., L\}$ by

$$(x_i, y_j) \ \in \ \mathcal{G}^l \Longleftrightarrow (x_{2^l i}, y_{2^l j}) \ \in \ \mathcal{G}^0. \tag{3}$$

The approximation of a function v on $\mathcal{G}^l$ is therefore obtained from its approximation on $\mathcal{G}^0$ by :

$$v^l_{ij} = v^0_{2^l i \ 2^l j} \quad i = 0, .., Nx/2^l \ \ j = 0, .., Ny/2^l \tag{4}$$

if $N_p = (N_x + 1) \times (N_y + 1)$ is the size of $\mathcal{G}^0$.
Now, the *wavelet* or *detail* coefficients at level l, d^l_{ij}, are computed as the

difference between the approximation of v on $\mathcal{G}^l$ and on the finer grid $\mathcal{G}^{l-1}$. Essentially, these coefficients are evaluated comparing the exact values of v on $\mathcal{G}^{l-1}$ with some approximated values $\tilde{v}_{ij}^{l-1}$ obtained by a polynomial interpolation procedure using the values v_{ij}^l.
The important property is that the detail coefficients give an easy representation of the smoothness of v since their size depends directly on the local regularity of v (Harten, 1996). Consequently it will help to localize the non-smooth structures of the solution, like shocks or contact discontinuities.

2.2. THE THRESHOLDING PROCEDURE

This procedure associates to each detail coefficient d_{ij}^l a coefficient b_{ij}^l taking value 0 or 1 according to the size of $|d_{ij}^l|$.
For a given parameter ϵ, a tolerance parameter at level l is defined as $\epsilon_l = \epsilon 2^{-l}$. Then, if the size of $|d_{ij}^l|$ is larger than ϵ_l, which means that the point $(x_{2^l i}, y_{2^l j})$ is in a non-smooth region, the value of b_{ij}^l and of its neighboring points are set to 1. Moreover, a test is also performed to check the decrease rate of the detail coefficients to take into account a possible formation of shocks (see (Bihari and Harten, 1997) for details).
Performing this procedure for all the scales $l = L, .., 1$, results in a mask, associated to the finest grid $\mathcal{G}^0$, which will be used to determine the procedure to evaluate the numerical divergence $\mathcal{D}$.

2.3. MULTILEVEL COMPUTATION OF $\mathcal{D}$

Taking for example the simplest time evolution scheme,

$$\vec{U}_{ij}^{n+1} = \vec{U}_{ij}^n - \delta t \ \mathcal{D}_{ij}(\vec{U}^n), \tag{5}$$

the algorithm is justified intuitively by the following observation :
if no non-smooth zone has been detected in $\vec{U}^n$ around the point (x_i, y_j) and if no shock-formation has been forecasted by the thresholding procedure, the region around (x_i, y_j) remains smooth for $\vec{U}^{n+1}$. Indeed, due to relation (5), the numerical divergence is as smooth as the solution at this point and can be computed with a simple polynomial interpolation procedure. The algorithm is as follows :

- $\mathcal{D}_{ij}(\vec{U}^n)$ is computed using the high order scheme for all the points on the coarsest grid $\mathcal{G}^L$.
- Once the divergence is known on grid $\mathcal{G}^l$, its values on the finer grid $\mathcal{G}^{l-1}$ are according to the mask coefficients :

  if $b_{ij}^l = 1$ $\quad$ $\mathcal{D}_{ij}(\vec{U}^n)$ is computed with the high order scheme.

  if $b_{ij}^l = 0$ $\quad$ $\mathcal{D}_{ij}(\vec{U}^n)$ is interpolated from its values on $\mathcal{G}^l$.

Decreasing l from L to 1 gives us the values of $\mathcal{D}(\vec{U}^n)$ on the finest grid $\mathcal{G}^0$.

At each iteration of the algorithm, we then have to apply the three previous steps to compute $\vec{U}^{n+1}$ from $\vec{U}^n$. To reduce the computational cost, the multiresolution is only performed for the density ρ since this variable displays all the possible discontinuities.
The time evolution is done with a third order Runge-Kutta method (Shu and Osher, 1989) and the multiresolution and thresholding procedures are only applied to the two first intermediate steps. Justifications of these modifications and quantification of their efficiency can be found in (Chiavassa and Donat, 1999).

3. Numerical experiments

The algorithm described in last section is applied to a classical test: the Double Mach reflexion of a strong shock (see e.g. in (Woodward and Colella, 1984)).
On figure 1(top), we display the density field of the reference solution, *i.e.* without multiresolution, computed at time $t = 0.2$ with Marquina's scheme for a grid of 512×128 points.
On figure 1(middle), the corresponding simulation with the multilevel algorithm is displayed and also the adaptive grid used for the computation. This grid is obtained representing only the points of $\mathcal{G}^0$ where the numerical divergence is computed with the high order scheme. The smoothness analysis done from the detail coefficients is very efficient and is able to localize perfectly the discontinuities. Other zones are detected near the bottom boundary for example, due to the presence of small oscillations in both reference and multilevel solutions. These perturbations are only due to the initial and boundary conditions (see (Woodward and Colella, 1984)).
It may be noticed that the mixing of interpolated and resolved values in the numerical divergence does not affect the quality of the results.

In figure 2 we display the L_1 norm of the error between the reference density and the one obtained after applying the multilevel algorithm, i.e.

$$e_1 = \frac{1}{N_p} \sum_{i=0}^{N_x} \sum_{j=0}^{N_y} |\rho_{ij}^n - \rho_{ref_{ij}}^n| \tag{6}$$

versus the tolerance ϵ. The fact that this error decreases at least as ϵ is crucial and ensures that we keep a tight control on the quality of the multilevel simulations. The efficiency θ, is measured as the ratio between the cpu time of the reference computation and that of the multilevel one, at time $t = 0.2$. In table 1 we have reported this gain for different sizes of the

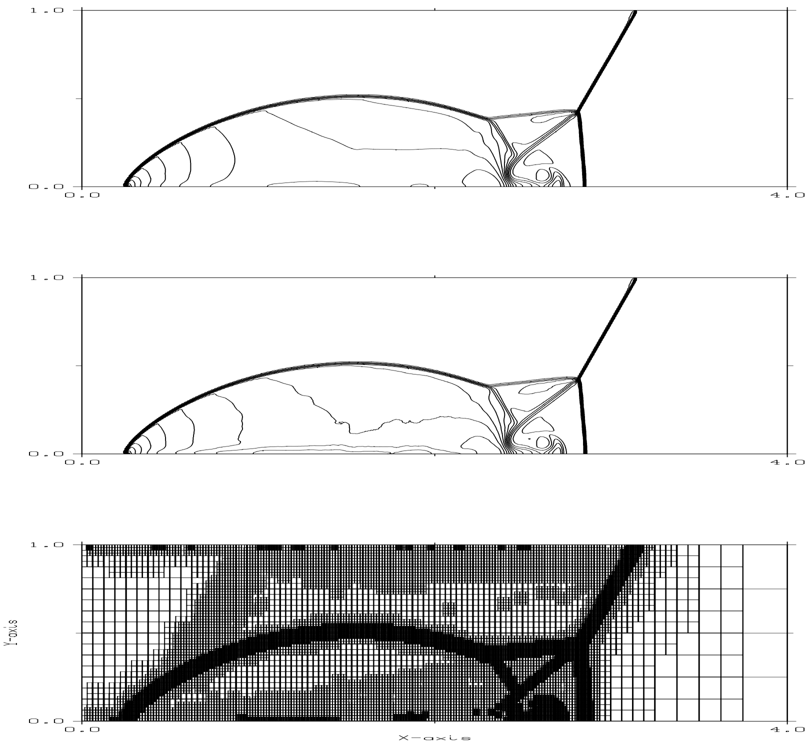

Figure 1. Density field of the reference solution (top), multilevel solution (middle) and corresponding adaptive grid (bottom). Initial grid $\mathcal{G}^0$ contains 512×128 points, number of levels is $L = 5$, and $\epsilon = 5 \times 10^{-3}$.

initial grid $\mathcal{G}^0$. The minimum and maximum percentage of points where the numerical divergence is computed with Marquina's scheme during the simulation is also reported.

It may be seen that the multilevel approach is able to reduce significantly the cpu time. Moreover the gain increases for finer grid computations due to the decrease of the percentage of resolved fluxes. Therefore, this algorithm could be very useful for simulations requiring very fine mesh.

4. Conclusion

In this paper we present the main steps of a multilevel algorithm to be applied to a classical two dimensional test. We observe a significant reduction in execution time without changing the quality of the results. Moreover,

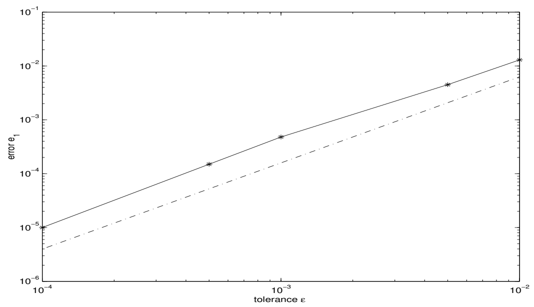

Figure 2. *Error e_1 between the multilevel and reference algorithm versus the tolerance ϵ. The dotted line represents the curve of equation:* $10\ \epsilon^{1.6}$.

grid size $\mathcal{G}^0$	$\%f_{\min}$ – $\%f_{\max}$	cpu gain θ
128×32	17.6 – 52.7	1.7
256×64	8.9 – 33.2	2.45
512×128	4.5 – 23.2	3.8

TABLE 1. *percentage of resolved flux and cpu gain θ at time $t = 0.2$.*

such approach is relatively simple to implement and does not require complex data structures thus, it could be easily introduced in existing CFD codes.

References

G. Chiavassa and R. Donat. (1999) Numerical Experiments with Point-Value Multiresolution for 2D Compressible Flows. Preprint University of Valencia, available at http://www.math.ntnu.no/conservation/2000/013.html.

B.L Bihari and A. Harten (1997) Multiresolution schemes for the numerical solutions of 2D conservation laws. *SIAM J. Sci. Comp.* **18**, pp315-354.

R. Donat and A. Marquina (1996) Capturing shock reflexion: An improved flux formula. *J. Comput. Phys.* **125**, pp 42-58.

A. Harten (1995) Multiresolution algorithms for the numerical solution of hyperbolic conservation laws. *Comm. Pure Appl. Math.* **48(12)**, pp1305-1342.

A. Harten (1996) Multiresolution representation of data: a general framework. *SIAM J. Numer. Anal.* **33**, pp1205–1256.

C. W Shu and S. J. Osher (1989) Efficient implementation of Essentially Non-oscillatory shock-capturing schemes. *J. Comput. Phys.* **83**, pp 32.

B. Sjögreen (1995) Numerical Experiments with the multiresolution scheme for the compressible Euler equations. *J. Comput. Phys.* **117**, pp 251.

P. R. Woodward and P. Colella (1984) The numerical simulation of two-dimensional fluid flow with strong shocks. *J. Comput. Phys.* **54**, pp 115.