Inviscid Flow on Moving Grids with Multiscale Space and Time Adaptivity

Philipp Lamby[1], Ralf Massjung[1], Siegfried Müller[1], and Youssef Stiriba[2]

[1] Institut für Geometrie und Praktische Mathematik, RWTH Aachen, D-52056 Aachen, Germany lamby,massjung,mueller@igpm.rwth-aachen.de
[2] Universitat "Rovira i Virgili" - ETSEQ, Departament Enginyeria Mecànica, Av. Paisos Catalans, 26, 43007 Tarragona, Spain youssef.stiriba@urv.net

Summary. A fully adaptive multiscale finite volume scheme for solving the 2D compressible Euler equations on moving grids is presented. The scheme uses a multiscale analysis based on biorthogonal wavelets to adapt the grid in space. Refinement in time is performed using a locally varying time stepping strategy that has been recently developed. The CFL condition is satisfied locally and the number of grid adaptations is reduced. The performance of the scheme using global and local multilevel time stepping, respectively, is investigated by a flow past an oscillating boundary.

1 Introduction

The solutions of hyperbolic conservation laws typically exhibit locally steep gradients and large regions where they are smooth. To account for the highly nonuniform spatial behavior, we need numerical schemes that adequately resolve the different scales, i.e., use a high resolution only near sharp transition regions and singularities but a moderate resolution in regions with smooth, slowly varying behavior of the solution.

In [2, 7] multiresolution techniques have been used to construct locally refined meshes on which the discretization is performed. The basic idea is to represent the cell averages on a given highest level of resolution as cell averages on some coarse level where the fine scale information is encoded in arrays of *detail coefficients* of ascending resolution. If the detail information of a cell is small, the grid is locally coarsened. By now the fully adaptive multiresolution concept has been applied by several groups with great success to different real world applications, cf. [1] and references cited there.

This work has been performed with funding by the Deutsche Forschungsgemeinschaft in the Collaborative Research Center SFB 401 "Flow Modulation and Fluid-Structure Interaction at Airplane Wings" of the RWTH Aachen, Germany and the "Ramón y Cajal" program of the Ministerio de Educacion y Ciencia, Spain.

So far a short-coming of this approach has been the lack of *temporal adaptivity*, i.e., all cell averages are evolved in time by the same time step size Δt satisfying the CFL condition for the cells on the *finest* mesh. Recently, a local time stepping strategy has been incorporated to the concept of fully adaptive multiresolution schemes, cf. [8, 5]. This has to be adjusted to the requirement that the resulting scheme provides an accuracy that is comparable to the accuracy of the reference mesh.

In the present work we apply this concept to 2D inviscid compressible fluid flows taking into account moving boundaries. This flow is governed by the arbitrary Lagrangian Eulerian (ALE) formulation of the Euler equations, cf. Sect. 2, that are discretized by a finite volume scheme, cf. Sect. 3. The efficiency of the reference scheme is improved by employing multiscale-based grid adaptation and local multilevel time stepping strategies, cf. Sect. 4. The adaptive scheme is applied to an oscillating boundary problem. Here we focus on the gain by the multilevel time stepping in comparison to the global time stepping, cf. Sect. 5.

2 The ALE Formulation of the Euler Equations

In the present study, inviscid fluid flow is described by the Euler equations for a compressible gas. In order to solve problems in time dependent domains, including moving boundaries, we consider the governing equations in its arbitrary Lagrangian Eulerian formulation. Neglecting body forces and volume supply of energy, the conservation laws for any moving control volume $V \subset \Omega$ of the d-dimensional domain $\Omega \subset \mathbb{R}^d$ with boundary ∂V and outward unit normal vector $\boldsymbol{n}$ on the surface element $dS \subset \partial V$ can be written in integral form as:

$$\frac{\partial}{\partial t} \int_{V(t)} \boldsymbol{u}\, dV + \oint_{\partial V(t)} \boldsymbol{f}(\boldsymbol{u}, \dot{\boldsymbol{x}}) \cdot \boldsymbol{n}\, dS = \boldsymbol{0}\,. \tag{1}$$

This system of conservation laws has to be supplemented by initial values and boundary conditions, respectively. Here $\boldsymbol{u} = (\rho,\, \rho \boldsymbol{v},\, \rho E)^T$ denotes the vector of the unknown conserved quantities and $\boldsymbol{f}^c$ represents the convective flux:

$$\boldsymbol{f}(\boldsymbol{u}, \dot{\boldsymbol{x}}) = \begin{pmatrix} \rho(\boldsymbol{v} - \dot{\boldsymbol{x}}) \\ \rho(\boldsymbol{v} - \dot{\boldsymbol{x}}) \circ \boldsymbol{v} + p\, \boldsymbol{I} \\ \rho E(\boldsymbol{v} - \dot{\boldsymbol{x}}) + p \boldsymbol{v} \end{pmatrix} = \boldsymbol{f}(\boldsymbol{u}, \boldsymbol{0}) - \boldsymbol{u} \circ \dot{\boldsymbol{x}}, \tag{2}$$

where ρ denotes the density, p the static pressure, $\boldsymbol{v}$ the velocity vector of the fluid and E the total energy. Here $\circ$ is the dyadic product. The motion of the grid is considered by the convective fluxes, where $\dot{\boldsymbol{x}}$ expresses the grid velocity. The static pressure is related to the specific internal energy according to the equation of state for a perfect gas $p = \rho\,(\gamma - 1)\left(E - 1/2\, \boldsymbol{v}^2\right)$, where γ is the ratio of specific heats, which is taken as 1.4 for air.

3 Finite Volume Discretization

The balance equations (1) are solved approximately by a finite volume method. For this purpose the finite fluid domain $\Omega(t)$ is split into a finite set of moving subdomains, the cells $V_i(t)$, such that all $V_i(t)$ are disjoint at each instant of time and that their union gives $\Omega(t)$. Furthermore let $\mathcal{N}(i)$ be the set of cells that have a common edge with the cell i, and for $j \in \mathcal{N}(i)$ let $e_{ij}(t) := \partial V_i(t) \cap \partial V_j(t)$ be the interface between the cells i and j. The time interval is discretized by $t^{n+1} = t^n + \Delta t$ assuming a constant time step size. On this particular discretization the finite volume scheme can be written as

$$|V_i^{n+1}|\, \boldsymbol{v}_i^{n+1} = |V_i^n|\, \boldsymbol{v}_i^n - \Delta t \sum_{j \in \mathcal{N}(i)} |e_{ij}|\, \boldsymbol{F}(\boldsymbol{v}_i^n, \boldsymbol{v}_j^n, \dot{\boldsymbol{x}}_{ij}, \boldsymbol{n}_{ij}) \tag{3}$$

using an explicit time discretization to compute the approximated cell averages $\boldsymbol{v}_i^{n+1}$ on the new time level. Here the numerical flux function $\boldsymbol{F}(\boldsymbol{u}, \boldsymbol{v}, \dot{\boldsymbol{x}}, \boldsymbol{n})$ is an approximation for the flux $\boldsymbol{f}(\boldsymbol{u}, \dot{\boldsymbol{x}}, \boldsymbol{n})$ in normal direction on the edge e_{ij}. It is assume to be *consistent*, i.e.,

$$\boldsymbol{F}(\boldsymbol{u}, \boldsymbol{u}, \dot{\boldsymbol{x}}, \boldsymbol{n}) = \boldsymbol{f}(\boldsymbol{u}, \dot{\boldsymbol{x}}, \boldsymbol{n}) := \boldsymbol{f}(\boldsymbol{u}, \dot{\boldsymbol{x}}) \cdot \boldsymbol{n}. \tag{4}$$

For simplicity of presentation we neglect that due to higher order reconstruction the numerical flux usually depends on an enlarged stencil of cell averages.

3.1 Grid Generation and Grid Movement.

For the simulation of moving boundaries the grid generator has to cope with time dependent domain boundaries. To accomplish this task efficiently we employ for each time level t^n a *parametric mapping* $\mathbf{x} : [0,1]^2 \to \Omega$ from a logical space to the physical domain $\Omega(t^n)$. In this setting grid cells are the images of the corresponding cells in logical space, i.e., $V_i = \boldsymbol{x}(R_i)$ corresponding to the interval $R_i \subset [0,1]^2$. Then the discrete grid is determined simply by function evaluation.

For the representation of such a parameter mapping we use tensor product B-splines, i.e., $\boldsymbol{x}(u,v) = \sum_{i=0}^{N} \sum_{j=0}^{M} \boldsymbol{p}_{i,j}\, N_{i,p_u,U}(u)\, N_{j,p_v,V}(v)$. Here $N_{i,p,T}$ denotes the i-th normalized B-spline of order p with respect to the knot vector T. In our applications we usually choose cubic splines ($p = 4$), cf. [1].

The $\boldsymbol{p}_{ij}$ are the *control points* that are not to be confused with grid points. Typically, the number of control is much smaller than the number of grid points in the discrete grid. This makes grid deformation by parametric B-spline mappings highly efficient; only few control points have to be moved instead of all the grid points in the discrete grid. More elaborate details on grid generation via B-Splines can be found in [4].

From the grid functions we compute a space-time grid function that is realized by a two-level time discretization: before the timestep $t^n \to t^{n+1}$ is performed the grid generation module provides two grid representations $\boldsymbol{x}(\boldsymbol{\xi}, t^n)$ and $\boldsymbol{x}(\boldsymbol{\xi}, t^{n+1})$ at time levels t^n and t^{n+1}, respectively. Then for $t \in (t^n, t^{n+1})$ the grid function is determined by linear interpolation.

3.2 The Geometric Conservation Law.

The "geometric conservation laws" are discrete consistency conditions for the finite volume scheme. They stem from the requirement, that a reasonable numerical method should at least be able to maintain a constant flow field: if $\boldsymbol{u}(\boldsymbol{x},t) = \boldsymbol{u}_\infty$ for all $(\boldsymbol{x},t)$, then we require that the numerical solution fulfills $\boldsymbol{u}_i^n = \boldsymbol{u}_\infty$ for all index pairs (i,n), too. In the special case of a stationary grid we then get for each cell V_i the consistency condition for discretizations of the form (3)

$$\mathbf{0} = \sum_{j \in N(i)} |e_{ij}| \boldsymbol{n}_{ij}. \tag{5}$$

What people usually understand to be "the" geometric conservation law stems form the requirement, that the constant homogeneous flow should also be reproduced if the mesh is moving. If we assume equation (5) to be satisfied, we end up for each cell V_i with the condition

$$|V_i^{n+1}| - |V_i^n| = \Delta t \sum_{j \in N(i)} |e_{ij}| \kappa_{ij}. \tag{6}$$

Here $\kappa_{ij} = \boldsymbol{n}_{ij} \cdot \dot{\boldsymbol{x}}_{ij}$ denotes the normal grid velocity on the face e_{ij}. The grid generator has to provide the quantities $|e_{ij}|$, $\boldsymbol{n}_{ij}$, κ_{ij} and $|V_i|$ for the flow solver such that the consistency conditions (5) and (6) hold. On a curvilinear grid where these quantities are not uniquely defined this can be achieved by evaluating the integrals

$$\boldsymbol{N}_{ij} := \int_{e_{ij}(t)} \mathbf{n}(s,t) ds, \quad S_{ij} := \int_{t^n}^{t^{n+1}} \int_{e_{ij}(t)} \dot{\boldsymbol{x}}(s,t) \cdot \boldsymbol{n}_{ij}(s,t) \; ds dt.$$

exactly and then setting

$$|e_{ij}| := ||\boldsymbol{N}_{ij}||_2, \quad \boldsymbol{n}_{ij} := \boldsymbol{N}_{ij}/|e_{ij}|, \quad \kappa_{ij} := S_{ij}/(\Delta t\, |e_{ij}|). \tag{7}$$

3.3 The Numerical Flux.

The fluxes in normal direction are approximated by an approximate Riemann solver. Since the cell edges are time-dependent we have to take into account the grid movement when solving the Riemann problem at the interfaces. For this purpose, we exploit the rotational and Galilean invariance of the underlying balance equations (1). Then we can rewrite the fluxes in normal direction as

$$\boldsymbol{f}(\boldsymbol{u}, \dot{\boldsymbol{x}}, \boldsymbol{n}) = \boldsymbol{S}\, \boldsymbol{f}(\boldsymbol{S}^{-1}\, \boldsymbol{u}, \mathbf{0}, \boldsymbol{n}) \quad \text{with} \quad \boldsymbol{S} = \begin{pmatrix} 1 & \mathbf{0}^T & 0 \\ \dot{\boldsymbol{x}} & \boldsymbol{I} & \mathbf{0} \\ \frac{1}{2}\dot{\boldsymbol{x}}^2 & \dot{\boldsymbol{x}}^T & 1 \end{pmatrix},$$

cf. [6]. Carrying this identity over to the numerical flux we obtain

$$\boldsymbol{F}(\boldsymbol{u}_l, \boldsymbol{u}_r, \dot{\boldsymbol{x}}, \boldsymbol{n}) = \boldsymbol{S}\,\boldsymbol{F}(\boldsymbol{S}^{-1}\,\boldsymbol{u}_l, \boldsymbol{S}^{-1}\,\boldsymbol{u}_r, \boldsymbol{0}, \boldsymbol{n}). \tag{8}$$

Hence, we may derive a numerical flux over moving edges from standard numerical fluxes on stationary grids. Note, that in the computations only the normal grid velocity κ is essentially needed. To perform the transformation (8) step by step it is sufficient to use $\kappa\,\boldsymbol{n}$ instead of $\dot{\boldsymbol{x}}$. This is admissible provided that the numerical flux is rotational invariant.

In the present work we use Roe's approximate Riemann solver. In order to avoid non-physical expansion shocks we use Harten's entropy fix. The spatial and temporal accuracy are improved by using a quasi one-dimensional second-order ENO reconstruction and Taylor expansion according to [3]. Here the reconstruction is applied to the characteristic variables.

4 Adaptive Multiscale Method

The efficiency of the reference finite volume scheme presented in Section 3 is significantly improved by employing recent multiscale-based grid adaptation techniques. Here we briefly summarize the basic conceptual ideas. For technical details we refer the reader to the book [7] and [1], respectively.

4.1 Multiscale-Based Spatial Grid Adaptation

Step 1: Multiscale analysis. The fundamental idea is to present the cell averages $\hat{\boldsymbol{u}}_L$ representing the discretized flow field at fixed time level t^n on a given uniform highest level of resolution $l = L$ (*reference mesh*) associated with a given finite volume discretization (*reference scheme*) as cell averages on some coarsest level $l = 0$ where the fine scale information is encoded in arrays of *detail coefficients* $\boldsymbol{d}_l$, $l = 0, \ldots, L-1$ of ascending resolution, see Figure 2.

The multiscale decomposition is performed on a hierarchy of *nested* grids $\mathcal{G}_l$ with increasing resolution $l = 0, \ldots, L$ determined by dyadic grid refinement of the logical space, see Figure 1. Note that this grid hierarchy can be efficiently realized by the parametric B-spline mappings in Section 3.1.

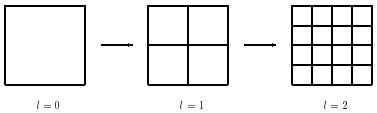

Fig. 1. Sequence of nested grids

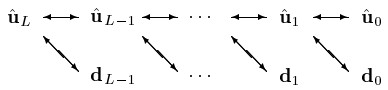

Fig. 2. Multiscale transformation

Step 2: Thresholding. It can be shown that the detail coefficients become small with increasing refinement level when the underlying function is smooth. In order to compress the original data this motivates us to discard all detail coefficients $d_{l,\boldsymbol{k}}$ whose absolute values fall below a level-dependent threshold value $\varepsilon_l = 2^{l-L}\varepsilon$. Let $\mathcal{D}_{L,\epsilon}$ be the set of *significant details*. The ideal strategy would be to determine the threshold value ϵ such that the *discretization error*

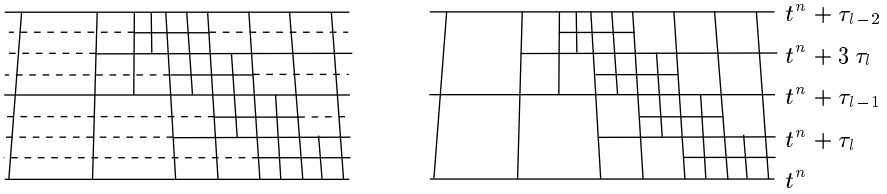

Fig. 3. Synchronized time evolution on space-time grid

of the reference scheme, i.e., difference between exact solution and reference scheme, and the *perturbation error*, i.e., the difference between the reference scheme and the adaptive scheme, are balanced, see [2].

Step 3: Prediction and grading. Since the flow field evolves in time, grid adaptation is performed after each evolution step to provide the adaptive grid at the *new* time level. In order to guarantee the adaptive scheme to be *reliable* in the sense that no significant future feature of the solution is missed, we have to *predict* all significant details at the new time level $n+1$ by means of the details at the *old* time level n. Let $\tilde{\mathcal{D}}_{L,\epsilon}^{n+1} \supset \mathcal{D}_{L,\epsilon}^{n} \cup \mathcal{D}_{L,\epsilon}^{n+1}$ be the prediction set. The prediction strategy is detailed in [2]. In view of the grid adaptation step this set is additionally inflated such that it corresponds to graded tree.

Step 4: Grid adaptation. By means of the set $\tilde{\mathcal{D}}_{L,\epsilon}^{n+1}$ a locally refined grid is determined. For this purpose, we recursively check proceeding levelwise from coarse to fine whether there exists a significant detail to a cell. If there is one, then we refine the respective cell. We finally obtain the locally refined grid with hanging nodes represented by the index set $\mathcal{G}_{L,\epsilon}$.

4.2 Multilevel Time Stepping

Since the reference scheme (3) is assumed to use an explicit time discretization, the time step size is bounded due to the CFL condition by the smallest cell in the grid. Hence Δt is determined by the highest refinement level L, i.e., $\Delta t = \tau_L$. However, for cells on the coarser scales $l = 0, \ldots, L-1$ we may use $\Delta t = \tau_l = 2^{L-l}\,\tau_L$ to satisfy locally the CFL condition. In [8] a multilevel time stepping strategy has been incorporated recently to the adaptive multiscale finite volume scheme as proposed in [7]. The basic idea is to save flux evaluations where the local CFL condition allows a large time step. The precise time evolution algorithm is schematically described by Fig. 3: In a global time stepping, i.e., using $\Delta t = \tau_L$ for all cells, each vertical line section appearing in Fig. 3 (left) represents a flux evaluation and each horizontal line (dashed or drawn) represents a cell update of $\boldsymbol{u}$ due to the fluxes. In the multilevel time stepping a flux evaluation is only performed at vertical line sections that emanate from a point where at least one drawn horizontal line section emanates from. If a vertical line section emanates from a point, where two dashed horizontal sections emanate from, then we do not recompute the

flux, but keep the flux value from the preceeding vertical line section. Hence fluxes are only computed for the vertical edges in Fig. 3 (right).

Note, that on each intermediate time level (horizontal lines) $\boldsymbol{u}$ is updated for all cells and that grid adaptation is performed at each *even* intermediate time level, i.e., at $t^n + k\,\tau_L$ for k even. Hence it is possible to track, for instance, a shock movement on the intermediate time levels instead of a–priori refining the whole range of influence, see Fig. 3 (right).

Note further, that τ_0 is the time scale at which the grid movement takes place. The grid position is only to the boundary movement at the time levels $t^n, t^{n+1}, \ldots$ and on the intermediate time levels the grid movement is a linear interpolation between the grid positions at t^n and t^{n+1}. This means that the time step size $t^{n+1} - t^n$ is dictated either by the time scale of the boundary movement or the time step size τ_0 according to the CFL condition on the coarsest spacial scale.

5 Numerical Results

This example shows the inviscid flow over a an oscillating plate with prescribed deformation in time. The flow domain extends from -5 to 5 in x-direction and from 0 to 5 in y-direction. At time $t = 0$ the lower boundary starts a periodic oscillation in the interval [0,1] prescribed by a B-Spline representation $\boldsymbol{x}(\xi, t) = \sum_{i=0}^{12} \boldsymbol{p}_i(t) N_{i,4,T}(\xi)$. Here $T = (0, 0, 0, 0, \frac{1}{10}, \frac{2}{10}, \ldots, \frac{9}{10}, 1, 1, 1, 1)$ and the movement of the control points is given by $\boldsymbol{p}_0 = (0, 0)^T$, $\boldsymbol{p}_{12} = (1, 0)$, $\boldsymbol{p}_1 = (\frac{1}{30}, 0)^T$, $\boldsymbol{p}_{11} = (\frac{29}{30}, 0)^T$, and $\boldsymbol{p}_i(t) = (-\frac{1}{10} + \frac{i}{10}, \frac{1}{5} \sin(t/t_{ref}) \sin(\frac{\pi}{8}(i-2)))$ for $i = 2, \ldots, 10$. Due to the simplicity of the geometry the grid deformation is performed using transfinite interpolation techniques. The flow enters the domain from the left hand side with free-stream conditions $\rho_\infty = 1.2929$ [kg/m^3], $p_\infty = 101325$ [Pa], $\boldsymbol{v}_\infty = (165.619, 0)$ [m/s]. The reference time is determined by $t_{ref} = 1./\sqrt{p_\infty/\rho_\infty} = 279.947$ [m/s]. At the boundaries we impose slip conditions, i.e., $\dot{\boldsymbol{x}} \cdot \boldsymbol{n} = \boldsymbol{v} \cdot \boldsymbol{n}$, at the lower boundary and characteristic boundary conditions elsewhere because of the subsonic free-stream conditions ($M_\infty = 0.5$). The grid is adapted after every timestep. The maximum refinement level is $L_{max} = 5$, the threshold $\epsilon = 0.002$, the coarsest grid consist of 1375 cells. After two cycles of the boundary oscillation the number of grid cells varies around 40.000 grid points depending on the phase of the boundary movement.

The bump is moving periodically up and down which is reflected in Figure 4 where the deflection in the midpoint of the bump is shown. When the bump is moving upwards then a shock occurs at the leeward side because of the acceleration of the flow. The shock weakens and moves in upstream direction when the bump moves downward. This can be deduced from Figure 5 where the Mach number in the midpoint of the bump is plotted versus the dimensionless time t/t_{ref}. When the shock is passing a steep gradient can be seen.

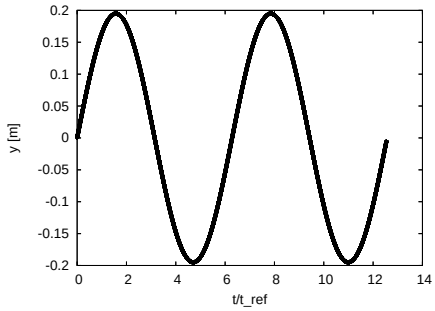

Fig. 4. Deflection at bump midpoint

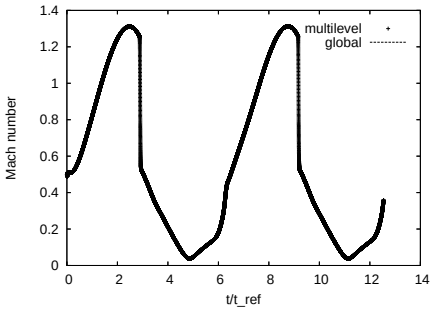

Fig. 5. Mach number at bump midpoint

The computation has been performed using the global and the multilevel time stepping strategy, respectively. Although we perform no grid deformation step for the intermediate time levels in the latter case the accuracy of the solution is not affected as can be concluded from Figure 5. On the other hand we gain a factor of 3.7 in comparison to a global time stepping strategy.

References

1. Bramkamp, F., Lamby, Ph. Müller, S.: An adaptive multiscale finite volume solver for unsteady and steady flow computations. Journal of Computational Physics, **197**, 460–490 (2004)
2. Cohen, A., Kaber, S.M., Müller, S., Postel, M.: Fully adaptive multiresolution finite volume schemes for conservation laws. Math. Comp, **72**, 183-225 (2003)
3. Harten, A., Engquist, B., Osher, S., Chakravarthy, SR.: Uniformly high order accurate essentially non–oscillatory schemes III. Journal of Computational Physics, **71**, 231–303 (1987)
4. Lamby, Ph.: Working Title. Phd Thesis, RWTH Aachen, Aachen (2005)
5. Lamby, Ph., Müller, S., Stiriba, Y.: Solution of shallow water equations using fully adaptive multiscale schemes. International Journal for Numerical Methods in Fluids, **49**, 417–437 (2005)
6. Massjung, R.: Numerical Schemes and Well-Posedness in Nonlinear Aeroelasticity. Phd Thesis, RWTH Aachen, Aachen (2002)
7. Müller, S.: Adaptive Multiscale Schemes for Conservation Laws. in: Lecture Notes on Computational Science and Engineering, vol. 27, Springer, Berlin Heidelberg New York (2002)
8. Müller, S., Stiriba, Y.: Fully adaptive multiscale schemes for conservation laws employing locally varying time stepping. IGPM-Report 238, RWTH Aachen (2004)