Aachen Institute for Advanced Study in Computational Engineering Science

Preprint: AICES-2011/08-02

29/August/2011

A High-Order Discontinuous Galerkin Discretization with Multiwavelet-Based Grid Adaptation for Compressible Flows

F. Iacono, G. May, S. Müller, R. Schäfer

RWTH AACHEN UNIVERSITY

AICES

Financial support from the Deutsche Forschungsgemeinschaft (German Research Association) through grant GSC 111 is gratefully acknowledged.

List of AICES technical reports: http://www.aices.rwth-aachen.de/preprints

A High-Order Discontinuous Galerkin Discretization with Multiwavelet-Based Grid Adaptation for Compressible Flows

Francesca Iacono[a,1], Georg May[a,2], Siegfried Müller[b,*], Roland Schäfer[b,3]

[a] *Aachen Institute for Advanced Study in Computational Engineering Science, RWTH Aachen University, Schinkelstraße 2, 52056, Aachen, Germany*

[b] *Institut für Geometrie und Praktische Mathematik, RWTH Aachen University, Templergraben 55, 52056, Aachen, Germany*

Abstract

The modern vision of a flow solver necessarily includes adaptivity. In particular, mesh adaptivity enables the solution strategy to allocate the resources efficiently, in that cells are concentrated in areas where they are needed, as opposed to uniform mesh refinement. Multiresolution-based mesh adaptivity using biorthogonal wavelets has been quite successful with finite volume solvers for compressible fluid flow. The extension of the multiresolution-based mesh adaptation concept to high-order discontinuous Galerkin discretizations can be performed using multiwavelets, which allow for higher-order vanishing moments, while maintaining local support. An implementation for scalar one-dimensional conservation laws has already been developed and tested. We extend here this strategy to systems of equations, in particular to the equations governing inviscid compressible flows. While the accuracy of the reference scheme on uniform grid is preserved, storage requirements, and concurrently computational time, are heavily reduced.

Keywords: Grid adaptivity, Multiresolution analysis, High-order methods, Multiwavelet, Discontinuous Galerkin, Conservation laws

1. Introduction

Compressible fluid flows are characterized by structures which have different length scales, such as shocks, contact discontinuities, viscous layers, or vortices. Given the desired accuracy, the smallest scale governs the number of grid points in the meshing process. If the whole domain is refined uniformly, this can lead to an unnecessarily large number of cells, i.e., to a large number of DOFs. This is especially true for high-order discretizations such as discontinuous Galerkin (DG) methods, since there can be quite a few DOFs in every cell. This is where *adaptive grid* methods [1–5] come into play. Adaptive mesh refinement enables one to capture very small scales of shock waves by lower-order approximations, without degrading the overall accuracy. The grid adaptivity process consists of adding cells in regions of the domain which are marked by a 'sensor'. In this way, the solution DOFs are 'localized' and resources can be allocated more efficiently compared to a uniform grid refinement. Grid adaptivity aims both at achieving accuracy and saving resources, by resolving flow features of relevance and at the same time being able to decide which features are negligible, without knowing a priori the flow field.

We decouple the different aspects of the solution strategy and concentrate our interest on mesh adaptation (note that we do not consider *p*-adaptivity). In particular, we assume that the discretization strategy is given and look for the optimal computational mesh. In this context, a mesh is considered 'optimal' if it allows to reach the target accuracy with the lowest number of cells. *Multiresolution-based mesh adaptivity* using biorthogonal wavelets has been quite successful with finite volume solvers for compressible fluid flow [6]. The rationale behind its design is, given a reference scheme on a uniform fine mesh, to find an adaptive (coarser) grid on which computations are actually performed.

*E-mail address: mueller@igpm.rwth-aachen.de

[1] E-mail address: iacono@aices.rwth-aachen.de

[2] E-mail address: may@aices.rwth-aachen.de

[3] E-mail address: schaefer@igpm.rwth-aachen.de

This adaptive grid is obtained starting from a coarse grid (a uniform coarsening of the fine uniform mesh) and refining only where needed, in a way that the accuracy of the reference uniform fine discretization is preserved.

Adaptive grid methods for hyperbolic equations in one space dimension made one of their first appearances in 1983, when Harten and Hyman described a 'self-adjusting grid method' [7]. In [1] Berger identified an adaptive method for the solution of hyperbolic PDEs using finite difference techniques. The method proposed in [1] belongs to the large family of *error-estimator-based grid adaptivity* techniques. Another possibility is given by approaches based on *multiscale-based grid adaptivity*. The concept of multiscale (or multiresolution) representation goes back to classical work in the field of wavelets [8–11]. Thanks to *multiresolution analysis*, wavelets are constructed to build a basis in $L^2(\mathbb{R})$: wavelet bases deliver approximations which are localized both in scale and frequency. The core idea for our work was identified by Harten in the framework of finite volume (FV) methods [12]. A hierarchy of nested grids at different resolutions offers the chance of selecting locally the appropriate level of resolution. Given two different levels of resolution, the cell averages on the finest level can be represented as a combination of the cell averages on the coarser level and of arrays of *detail coefficients* (or wavelet coefficients, according to the terminology in [13]), containing the finer-scale information. This different format reveals insight into the local behavior of the solution and, on this basis, grid adaptivity can be performed in that only the cells containing information relevant on the fine scale are included in the adaptive grid.

Away from discontinuities of the solution, the detail coefficients become smaller as one goes to finer grids, with a rate which is determined by the local regularity of the underlying solution and the order of vanishing moments of the wavelets [14]; see [15] for the analogous proof in the discrete multiresolution framework. In a neighborhood of a discontinuity of the solution, the detail coefficients remain the same size (under the normalization used here), independent of the level of resolution. It is in this respect that finding the multiresolution representation of the cell averages corresponds to analyzing the local regularity of the solution. In Harten's scheme, however, this information is not exploited to diminish the number of DOFs, but rather to substitute costly flux evaluations with approximate values cheaply interpolated from the coarser grid. This approach has been called 'semi-adaptive' by Cohen in [16], because the numerical flux vector is replaced by its adaptive approximation, while the solution is still described and evolved on a uniform grid. Harten's idea has been then transferred to 'fully' adaptive grid methods in [17], where data compression is applied in regions where the solution is locally smooth and a locally refined grid is determined based on the compressed data set.

In this multiscale-based technique, a FV discretization on a fine uniform reference grid and the corresponding discretization error are taken as a reference. Cells are discarded as long as the accuracy of the reference discretization is preserved, so that less DOFs provide the same accuracy which would be provided by the fine reference grid. In order to realize a given error tolerance, one has ideally to bound the thresholding error by the discretization error of the reference scheme. Note that, even though a reference uniform fine grid must be available, in general no computations have to be performed on it. Besides, while grid adaptivity is usually based on the evaluation of some error estimator at every time step, here an estimate for the error is only employed in the analysis of the scalar case to prove which choices of the threshold value are 'safe'. The fact that an error estimator is not directly needed in the adaptivity process is a benefit, since error estimators are typically not available for compressible flow equations. Multiscale-based mesh adaptivity using biorthogonal wavelets has been quite successful with FV solvers for compressible fluid flows: see for example [6, 18–20]. Biorthogonal wavelets allow the approximation of functions defined on finite domains as well. In this context, the higher the number of vanishing moments, the faster the decay of the coefficients, provided that the solution is sufficiently regular. This is expressed by the *cancellation property* of wavelets [21]. For biorthogonal wavelets to realize more vanishing moments, however, one has to extend the support of the wavelet functions. This is a severe drawback with regard to complex geometries and parallelization.

In [22] the method has been extended to high-order discretizations by means of a DG method, where by high-order we refer to methods that have order of consistency larger than two. The extension of the multiscale-based mesh adaptivity concept to high-order DG discretizations has been done using so-called *multiwavelets* [23], which allow for higher-order vanishing moments, while being defined on only one mesh element. An implementation for scalar one-dimensional conservation laws has been developed and tested in [22] and [24].

The main contribution of this work is the extension to systems of equations, in particular to the equations governing one-dimensional inviscid compressible flows [25–27]. The paper is organized as follows. In Section 2 we

elaborate on the reference DG scheme, which is performed on a uniform fine grid. In Section 3 we introduce and discuss the multiwavelet-based adaptive strategy. We then consider in Section 4 numerical examples of the adaptive strategy applied to the time-dependent one-dimensional Euler equations. The reduction in the number of DOFs and computational time shows to be very favorable. Even though a rigorous analysis as the one for scalar equations [22, 24] cannot be performed for systems, empirical rules for the choice of the thresholding tolerance give satisfactory results. Other adaptive strategies have been developed which share some of the features of our approach [28–33]. In Section 5, we describe some of these techniques and compare them to our method. Based on numerical results, the multiresolution-based adaptivity proposed here appears to be the most efficient among the methods which apply multiwavelets for data analysis in the framework of high-order discretization of compressible flow problems. Also, it is underpinned by the rigorous error analysis for the scalar case, whereas the other methods present only qualitative results.

2. The Reference Modal Discontinuous Galerkin Scheme

We consider one-dimensional hyperbolic conservation laws. Here we restrict ourselves to problems that are *one-dimensional* in space, but the presented multiresolution-based grid adaptivity concept can be extended to the multi-dimensional case, as it has been done in the FV context [6]. Let t be the time variable and x the space variable. We consider the divergence form of a system of $r > 0$ homogeneous nonlinear hyperbolic PDEs:

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{f}(\mathbf{u})}{\partial x} = \mathbf{0}, \tag{1}$$

where $\mathbf{u} : \Omega \times [0, T] \to \mathbb{R}^r$ (with $\Omega \subset \mathbb{R}$ and $T > 0$), are the conserved quantities, and the flux vector is $\mathbf{f} : \mathbb{R}^r \to \mathbb{R}^r$, with $\mathbf{f} = (f_1, \ldots, f_r)^T$. The flux can be a nonlinear function of $\mathbf{u}$. These equations are subject to initial data

$$\mathbf{u}(x, 0) = \mathbf{u}_0(x), \tag{2}$$

and to suitable boundary conditions.

We describe the *non-adaptive reference scheme*, based on a modal DG discretization. In view of computations, we consider only compact domains $\Omega = [a, b]$ for the space variable and we take a uniform spatial discretization into N_E cells:

$$V^{(k)} := (x_k, x_{k+1}), \quad k \in \mathcal{I} := \{1, \ldots, N_E\}, \tag{3}$$

with $x_k := a + kh$, where $h = \dfrac{b-a}{N_E}$ is the constant grid size. We approximate the entropy solution of (1) with a modal DG discretization with polynomials of degree p. Let S^p be

$$S^p := \left\{ v \in L^2(\Omega) : \, v|_{V^{(k)}} \in \Pi_p\left(V^{(k)}\right), \forall k \in \mathcal{I} \right\}, \tag{4}$$

where $\Pi_p(V^{(k)})$ is the space of polynomials of degree at most p on $V^{(k)}$. Here we list the main steps of the modal DG scheme. Two sets of basis functions for the space S^p are considered: not only the *primal basis* $\Phi := \left\{ \varphi_i^{(k)}, \, k \in \mathcal{I}, \, i = 1, \ldots, N_p \right\}$, but also the *dual basis* $\tilde{\Phi} := \left\{ \tilde{\varphi}_i^{(k)}, \, k \in \mathcal{I}, \, i = 1, \ldots, N_p \right\}$. It should be noted that the number of basis functions is $N_p = p + 1$. The functions under consideration are expanded in the primal basis, whereas the dual basis is used to encode the coefficients of the expansion in each cell [19, 20]. We choose primal and dual basis so that they are *biorthogonal*

$$\left\langle \varphi_i^{(k)}, \tilde{\varphi}_{i'}^{(k')} \right\rangle_{L^2(\Omega)} = \delta_{i,i'} \, \delta_{k,k'} \tag{5}$$

and they have *compact support*

$$\operatorname{supp}\left(\varphi_i^{(k)}\right) = \operatorname{supp}\left(\tilde{\varphi}_i^{(k)}\right) = V^{(k)}. \tag{6}$$

In particular, we choose Φ and $\tilde{\Phi}$ to be the *Legendre polynomials* 'shifted' on Ω and normalized with respect to L^∞ and L^1, respectively. Here 'normalizing' means rescaling so that the norm of each basis function is independent of the specific cell $V^{(k)}$:

$$\left\| \varphi_i^{(k)} \right\|_{L^\infty(\Omega)} \lesssim 1, \qquad \left\| \tilde{\varphi}_i^{(k)} \right\|_{L^1(\Omega)} \lesssim 1, \tag{7}$$

where '$\lesssim$' indicates an upper estimate up to some uniform constant. We can now express the approximate solution $\mathbf{u}_h \in (S^p)^r$ as an expansion in the basis Φ:

$$\mathbf{u}_h(x,t) := \sum_{k\in\mathcal{I}} \sum_{i=1}^{N_p} \mathbf{u}_i^{(k)}(t)\, \varphi_i^{(k)}(x), \quad x \in \Omega, \tag{8}$$

where the coefficients $\mathbf{u}_i^{(k)}$ are obtained by means of the biorthogonality relation (5):

$$\mathbf{u}_i^{(k)}(t) = \int_a^b \mathbf{u}_h(x,t)\, \tilde{\varphi}_i^{(k)}(x)\, dx, \qquad \forall k \in \mathcal{I},\ \forall i = 1,\dots,N_p. \tag{9}$$

In order to derive evolution equations for these coefficients, we multiply (1) by the dual basis function $\tilde{\varphi}_i^{(k)}$, we integrate over its support $V^{(k)}$, substitute $\mathbf{u}_h(x,t)$ with its expansion (8) and integrate by parts. In this way, exploiting the biorthogonality property (5) and the locality of the supports of the basis functions, we obtain the variational formulation:

$$\frac{d\mathbf{u}_i^{(k)}}{dt} + \left[\mathbf{f}(\mathbf{u}_h)\, \tilde{\varphi}_i^{(k)}\right]_{x_k^+}^{x_{k+1}^-} - \int_{V^{(k)}} \mathbf{f}(\mathbf{u}_h)\, \frac{d\tilde{\varphi}_i^{(k)}}{dx}\, dx = \mathbf{0}, \tag{10}$$

where x_k^+ and x_k^- denote the limit values at $x = x_k$ from the right and from the left, respectively. The flux evaluations are substituted by numerical fluxes:

$$\mathbf{F}^{(k)} \approx \mathbf{f}(\mathbf{u}_h), \tag{11}$$

which are defined only at cell interfaces as

$$\begin{aligned} \mathbf{F}^{(k)}(x_k,\cdot) &:= \mathbf{F}\left(\mathbf{u}_h^{(k-1)}(x_k,\cdot),\, \mathbf{u}_h^{(k)}(x_k,\cdot)\right), \\ \mathbf{F}^{(k)}(x_{k+1},\cdot) &:= \mathbf{F}\left(\mathbf{u}_h^{(k)}(x_{k+1},\cdot),\, \mathbf{u}_h^{(k+1)}(x_{k+1},\cdot)\right), \end{aligned} \tag{12}$$

where $\mathbf{u}_h^{(k)} := \mathbf{u}_h|_{V^{(k)}}$. In this way, we obtain the semi-discretized DG formulation:

$$\frac{d\mathbf{u}_i^{(k)}}{dt} = -\left[\mathbf{F}^{(k)}\, \tilde{\varphi}_i^{(k)}\right]_{x_k}^{x_{k+1}} + \int_{V^{(k)}} \mathbf{f}(\mathbf{u}_h)\, \frac{d\tilde{\varphi}_i^{(k)}}{dx}\, dx. \tag{13}$$

The derivation of the semi-discretized scheme continues with the application of a quadrature formula to the integral: in particular, we choose Gauss-Legendre quadrature nodes [34].

At time instant $t = t^n$, $n \in \mathbb{N}$, we define

$$\mathbf{U}^n := \left\{\mathbf{u}_i^{(k),n},\, k \in \mathcal{I}, i = 1,\dots,N_p\right\} \in \mathbb{R}^{r\cdot N_E\cdot N_{p+1}}, \tag{14}$$

where $\mathbf{u}_i^{(k),n} := \mathbf{u}_i^{(k)}(t^n) \in \mathbb{R}^r$. We advance in time by means of the s-stage, q-th order explicit strong-stability-preserving Runge-Kutta method SSPRK(s,q). In particular, we will be using the SSPRK(5,4) scheme [35] for our computations in Section 4.

Limiting is needed to suppress oscillations near discontinuities which arise in hyperbolic equations. In our DG scheme, we apply a limiter obtained by making some changes to Shu's TVB modification [36] of van Leer's limiter [37]. This limiter [24] directly corrects the approximated solution in the whole cell, defining how the coefficients of order larger than 1 have to be modified. Moreover, it has a formulation independent of the polynomial degree p and satisfies some desired properties, cf. [24]. The limiting procedure is described in Appendix A.

The algorithm for the reference DG method is given in Algorithm 2.1. This standard algorithm is given merely as term of comparison for the adaptive procedure, which will be described in Algorithm 3.1. The reference DG scheme has been validated for scalar problems [24] and for systems of equations [25, 26].

Algorithm 2.1. *DG discretization on uniform non-adaptive grid*

1: ***for*** $n = 1, 2, \dots$ ***do***

2: ***for*** $\sigma = 1, \ldots, s$ ***do***
3: *Apply limiting operator*
4: *Compute residual*
5: *Perform σ-th stage of the Runge-Kutta solution method*
6: ***end for***
7: ***end for***

3. Discontinuous Galerkin Discretization with Multiscale-Based Grid Adaptivity

As already mentioned, our aim is to accelerate the convergence of a reference DG scheme on a uniform grid by using an adaptive mesh. In building this adaptive mesh no error larger than the discretization error of the reference scheme should be introduced. We use a multiscale-based grid adaptation technique. In the following we summarize the basic conceptual ideas and the main steps of the algorithm of the DG method employing multiwavelet-based adaptation. A detailed derivation can be found in [22, 24].

Nested Grid Hierarchy. It is necessary to introduce additional notation. The starting point for the multiresolution analysis is a coarse uniform mesh $\mathcal{G}_1$ with $N_E^{(1)}$ cells. Then one subdivides recursively ($L-1$ times) the cells of the uniform mesh $\mathcal{G}_1$ in their mid-point (dyadic subdivision). This results in a hierarchy of nested grids $\mathcal{G}_\ell$ with increasing resolution $\ell = 1, \ldots, L$ and $N_E^{(\ell)} = N_E^{(1)} 2^{\ell-1}$. We choose $\mathcal{G}_1$ and L such that the highest level of resolution $\ell = L$ corresponds to the uniform mesh associated with the DG discretization described in Section 2, i.e., $N_E^{(L)} = N_E$. In the following we refer to the mesh $\mathcal{G}_L$ as *reference mesh* and to the associated DG discretization as *reference scheme*. Let $\mathcal{I}_\ell := \{1, \ldots, N_E^{(\ell)}\}$ be the set of cells on level ℓ. In Figure 1 a dyadic grid hierarchy $\mathcal{G}_\ell = \left\{V^{(\ell,k)},\, k \in \mathcal{I}_\ell\right\}$ with $L = 4$ is illustrated. In particular, $V^{(\ell,k)} := (x_{\ell,k}, x_{\ell,k+1})$ is the k-th cell on level ℓ, with $x_{\ell,k} := a + (k-1)\, h_\ell$, and grid size

$$h_\ell = \frac{b-a}{N_E^{(\ell)}}. \tag{15}$$

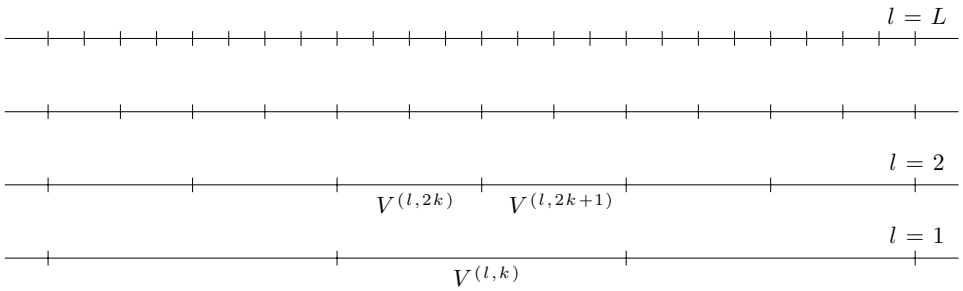


Figure 1: Sequence of nested dyadic one-dimensional grids.

Multiresolution Analysis. Let $\mathbf{U}^{(\ell)} = \left\{\mathbf{u}_i^{(\ell,k)},\, k \in \mathcal{I}_\ell,\ i = 1, \ldots, N_p\right\}$ be the array of (single-scale) coefficients representing the discretized flow field at some fixed time level t^n on level ℓ. On the set of grids described above we apply the concept of *multiresolution analysis* [8]. We consider the set of scaling functions $\Phi_\ell := \left\{\varphi_i^{(\ell,k)},\, k \in \mathcal{I}_\ell,\, i = 1, \ldots, N_p\right\}$ of shifted and L^∞-scaled Legendre polynomials. The set Φ_ℓ forms a basis for

$$S_\ell^p := \left\{v \in L^2([a,b]) :\, v|_{V^{(\ell,k)}} \in \Pi_p(V^{(\ell,k)}), \forall k \in \mathcal{I}_\ell\right\}, \tag{16}$$

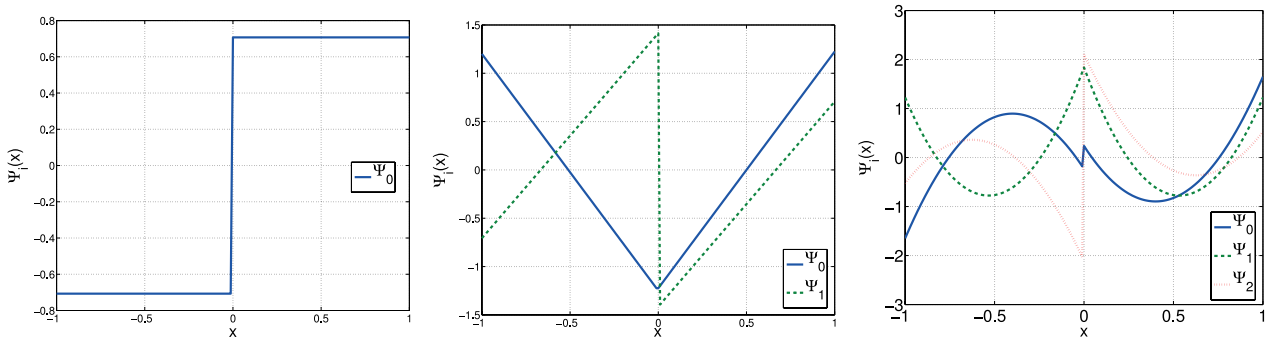
with $\dim\left(S_\ell^p\right) = N_E^{(\ell)} N_p$ and

$$S_1^p \subset \cdots \subset S_\ell^p \subset S_{\ell+1}^p \subset \cdots \subset S_L^p. \tag{17}$$

Multiwavelets $\Psi_\ell := \left\{\psi_i^{(\ell,k)},\ k \in \mathcal{I}_\ell,\ i = 1,\dots,N_p\right\}$ on level ℓ are then chosen so that they form a basis for the orthogonal complement space $S_{\ell+1}^p \backslash S_\ell^p$, whose dimension is $N_E^{(\ell)} N_p$. The union of Φ_ℓ and Ψ_ℓ forms a basis for $S_{\ell+1}^p$, i.e.,

$$\operatorname{span}(\Phi_\ell) \oplus \operatorname{span}(\Psi_\ell) = S_{\ell+1}^p. \tag{18}$$

Here we apply *Alpert's multiwavelets* [38]. Note that in the multiresolution analysis based on *wavelets*, a family of functions obtained from a *single* scaling function generates a basis for S_ℓ^p by the operations of dilation and translation. In the multiresolution analysis based on *multiwavelets*, a family of functions obtained from a *set* of scaling functions generates a basis for S_ℓ^p by the operations of dilation and translation. As we mentioned in Section 1, our choice falls on multiwavelets rather than on wavelets because of their better features: in particular, they have local support and at the same time provide high-order vanishing moments. On each level we consider Alpert's multiwavelets $\psi_i^{(\ell,k)}$, which are piecewise polynomials and form an orthogonal system to the scaling functions $\varphi_i^{(\ell,k)}$ [38, 39]. Whereas wavelets are in general continuous functions, this regularity is lost when passing to Alpert's multiwavelets. In Figure 2 the Alpert's multiwavelets for polynomial degree $p = 0$, 1 and 2 are depicted.



(a) $p = 0$ (b) $p = 1$ (c) $p = 2$

Figure 2: Alpert's multiwavelets of polynomial degree p.

Let us refer to $\varphi_i^{(\ell,k)}$ and $\psi_i^{(\ell,k)}$ as the primal *scaling functions* and primal *wavelet functions*, respectively. By $\tilde{\varphi}_i^{(\ell,k)}$ and $\tilde{\psi}_i^{(\ell,k)}$ we will denote the dual scaling functions and dual wavelet functions, respectively. Again, the primal and dual functions are normalized with respect to L^∞ and L^1, respectively, see equation (7). Thus the functions of the primal and dual systems have the following properties.

(i) They are locally supported:

$$\operatorname{supp}\left(\varphi_i^{(\ell,k)}\right) = \operatorname{supp}\left(\tilde{\varphi}_i^{(\ell,k)}\right) = \operatorname{supp}\left(\psi_i^{(\ell,k)}\right) = \operatorname{supp}\left(\tilde{\psi}_i^{(\ell,k)}\right) = V^{(\ell,k)}. \tag{19}$$

(ii) Each wavelet $\tilde{\psi}_i^{(\ell,k)}$, $\ell = 1,\dots,L$, $k = 1,\dots,N_E^{(\ell)}$, $i = 1,\dots,N_p$ (recall that $N_p = p+1$ here), provides vanishing moments:

$$\left\langle P, \tilde{\psi}_i^{(\ell,k)} \right\rangle_{L^2([a,b])} = 0, \qquad \forall P \in \Pi_{p+i}([a,b]). \tag{20}$$

(iii) They are biorthogonal:

$$\left\langle \varphi_i^{(\ell,k)}, \tilde{\varphi}_{i'}^{(\ell,k')} \right\rangle_{L^2([a,b])} = \delta_{i,i'}\,\delta_{k,k'}, \qquad \langle \varphi_i^{(\ell,k)}, \tilde{\psi}_{i'}^{(\ell,k')} \rangle_{L^2([a,b])} = 0, \tag{21}$$

$$\left\langle \psi_i^{(\ell,k)}, \tilde{\psi}_{i'}^{(\ell,k')} \right\rangle_{L^2([a,b])} = \delta_{i,i'}\,\delta_{k,k'}, \qquad \left\langle \psi_i^{(\ell,k)}, \tilde{\varphi}_{i'}^{(\ell,k')} \right\rangle_{L^2([a,b])} = 0, \tag{22}$$

for all $\ell = 1,\dots,L, k,k' = 1,\dots,N_E^{(\ell)}, i,i' = 1,\dots,N_p$.

(iv) There exists the *two-scale decomposition*

$$\tilde{\varphi}_i^{(\ell,k)} = \sum_{i'=1}^{N_p} \sum_{s=1}^{2} \tilde{m}_{i,i'}^{1,s}\, \tilde{\varphi}_{i'}^{(\ell+1,\,2k+s)}, \qquad i = 1,\dots,N_p, \tag{23}$$

$$\tilde{\psi}_i^{(\ell,k)} = \sum_{i'=1}^{N_p} \sum_{s=1}^{2} \tilde{m}_{i,i'}^{2,s}\, \tilde{\varphi}_{i'}^{(\ell+1,\,2k+s)}, \qquad i = 1,\dots,N_p, \tag{24}$$

and its inverse

$$\tilde{\varphi}_i^{(\ell+1,\,2k+s)} = \sum_{i'=1}^{N_p} \left(\tilde{g}_{i,i'}^{1,s}\, \tilde{\varphi}_{i'}^{(\ell,k)} + \tilde{g}_{i,i'}^{2,s}\, \tilde{\psi}_{i'}^{(\ell,k)} \right), \quad s = 1,2,\ i = 1,\dots,N_p, \tag{25}$$

where the mask coefficients are determined as

$$\tilde{m}_{i,i'}^{1,s} = \left\langle \tilde{\varphi}_i^{(\ell,k)},\, \varphi_{i'}^{(\ell+1,\,2k+s)} \right\rangle_{L^2([a,b])}, \quad \tilde{m}_{i,i'}^{2,s} = \left\langle \tilde{\psi}_i^{(\ell,k)},\, \varphi_{i'}^{(\ell+1,\,2k+s)} \right\rangle_{L^2([a,b])}, \tag{26}$$

$$\tilde{g}_{i,i'}^{1,s} = \left\langle \varphi_i^{(\ell+1,\,2k+s)},\, \tilde{\varphi}_{i'}^{(\ell,k)} \right\rangle_{L^2([a,b])}, \qquad \tilde{g}_{i,i'}^{2,s} = \left\langle \varphi_i^{(\ell+1,\,2k+s)},\, \tilde{\psi}_{i'}^{(\ell,k)} \right\rangle_{L^2([a,b])}. \tag{27}$$

Equations (23)-(25) express the relations between the basis functions on level ℓ and on level $\ell+1$. These relations exist as a consequence of (18).

Note that, in particular, $\varphi_1^{(\ell,k)}$ coincides with the L^1-normalized characteristic function, i.e., $\varphi_1^{(\ell,k)} = h_\ell^{-1} \chi_{V^{(\ell,k)}}$. Hence, the zero-order coefficients $\mathbf{u}_1^{(\ell,k)}$ coincide with cell averages $\overline{\mathbf{v}}_{\ell,k}$.

Multiscale Transformation. Due to property (iv), we determine the corresponding two-scale decomposition of the solution approximation on level $\ell+1$:

$$\mathbf{u}_h^{(\ell+1)}(x,t) = \sum_{k\in\mathcal{I}_\ell} \sum_{i=1}^{N_p} \left(\mathbf{u}_i^{(\ell,k)}(t)\, \varphi_i^{(\ell,k)}(x) + \mathbf{d}_i^{(\ell,k)}(t)\, \psi_i^{(\ell,k)}(x) \right), \quad x \in \Omega, \tag{28}$$

where, due to the biorthogonality property (iii),

$$\mathbf{u}_i^{(\ell,k)}(t) := \left\langle \mathbf{u}_h^{(\ell)}(\cdot,t),\, \tilde{\varphi}_i^{(\ell,k)} \right\rangle_{L^2(V^{(\ell,k)})}, \qquad \mathbf{d}_i^{(\ell,k)}(t) := \left\langle \mathbf{u}_h^{(\ell)}(\cdot,t),\, \tilde{\psi}_i^{(\ell,k)} \right\rangle_{L^2(V^{(\ell,k)})}. \tag{29}$$

For further technicalities, we refer the interested reader to [22]. The two-scale transformation (28) can be applied recursively, transforming the *single-scale* decomposition of the approximate solution on level L

$$\mathbf{u}_h^{(L)} = \sum_{k\in\mathcal{I}_L} \sum_{i=1}^{N_p} \mathbf{u}_i^{(L,k)}\, \varphi_i^{(L,k)} \tag{30}$$

into the *multiscale* decomposition

$$\mathbf{u}_h^{(L)} = \sum_{k\in\mathcal{I}_1} \sum_{i=1}^{N_p} \mathbf{u}_i^{(1,k)}\, \varphi_i^{(1,k)} + \sum_{\ell=1}^{L-1} \sum_{k\in\mathcal{I}_\ell} \sum_{i=1}^{N_p} \mathbf{d}_i^{(\ell,k)}\, \psi_i^{(\ell,k)}. \tag{31}$$

By this procedure, we essentially have rewritten the array of single-scale coefficients $\mathbf{U}^{(L)}$ as

1. a sequence of single-scale coefficients $\mathbf{U}^{(1)} = \left\{ \mathbf{u}_i^{(1,k)},\, k \in \mathcal{I}_1, i = 1,\dots,N_p \right\}$ representing an approximate solution on the coarsest level $\ell = 1$, and
2. $L-1$ arrays of so-called *detail coefficients* $\mathbf{d}^{(\ell)} = \left\{ \mathbf{d}_i^{(\ell,k)},\, k \in \mathcal{I}_\ell,\ i = 1,\dots,N_p \right\}$, $\ell = 1,\dots,L-1$, of ascending resolution, which carry the multiscale information.

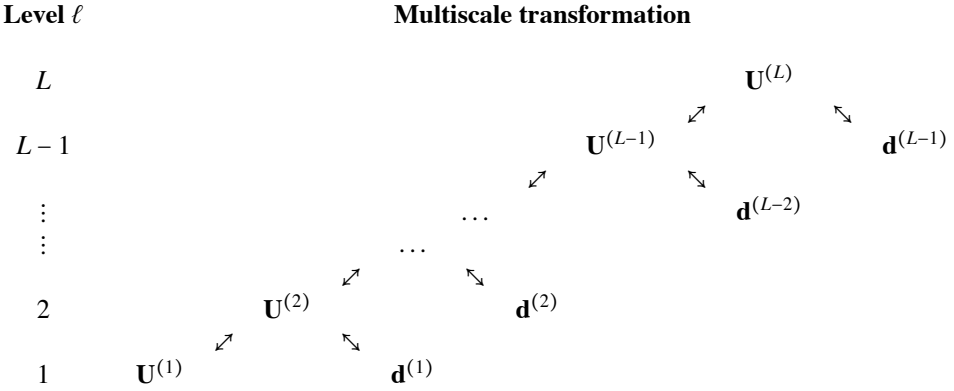


Figure 3: Multiscale transformation: it allows to switch easily between the single-fine-scale representation $\mathbf{U}^{(L)}$ and the multiscale representation offered by $\mathbf{U}^{(1)}$ and $\mathbf{d}^{(\ell)}$, $\ell = 1, \dots, L-1$.

Detail coefficients can be seen as carriers of individual features of the solution (in other words, *fluctuations*), which, if added to the lowest-resolution information, 'enrich' it up to the level L of resolution. Equation (31) expresses the *multiscale transformation*: see Figure 3.

Thresholding. We apply thresholding with a prescribed tolerance ε. Let us introduce the level-dependent threshold value $\varepsilon_\ell = \alpha\, 2^{\ell - L}\varepsilon$, where α is a constant. In the implementation we use $\alpha = 1/16$ in order to avoid spurious refinement triggered by the thresholding procedure, see [24] for details. We discard all the detail coefficients $\mathbf{d}_i^{(\ell,k)} \in \mathbb{R}^r$ whose absolute values, scaled with the maximum value of the solution, fall componentwise below ε_ℓ:

$$\mathbf{d}^{(\ell,k),thr} := \begin{cases} \mathbf{d}^{(\ell,k)}, & \text{if } \max\limits_{\substack{i=1,\dots,N_p \\ j=1,\dots,r}} \left(\dfrac{\left| \left(\mathbf{d}_i^{(\ell,k)} \right)_j \right|}{\max \left\{ \max\limits_{\substack{\ell=1,\dots,L \\ k \in \mathcal{I}_\ell}} \left| \left(\mathbf{u}_1^{(\ell,k)} \right)_j \right|, \, 1 \right\}} \right) > \varepsilon_\ell, \\ \mathbf{0}, & \text{else,} \end{cases} \tag{32}$$

where $\left(\mathbf{u}_1^{(\ell,k)} \right)_j$ is the cell average of the j-th conserved quantity in cell $V^{(\ell,k)}$. This step, called *hard thresholding*, is the keystone of the adaptivity process: null details correspond to cells which will not be included in the adaptive grid. The more null details, the smaller the number of DOFs in the adaptive grid. The detail coefficients which are not discarded are called *significant details*, because the information that they carry is assumed to be 'ε-significant' for the approximation of the solution. The set of significant details is characterized by the index set

$$\mathcal{D}_\varepsilon := \left\{ (\ell, k) : \ \mathbf{d}^{(\ell,k),thr} \neq \mathbf{0}, \ \ell = 1, \dots, L-1, \ k \in \mathcal{I}_\ell \right\}. \tag{33}$$

The thresholding process inserts an error of order ε to the solution. In particular, since the basis is orthonormal, the magnitude of the discarded coefficients provides information on the norm of the perturbation error. This justifies why we discard the smaller coefficients. Besides, thanks to the fact that we have normalized the basis and to property (ii), the detail coefficients become smaller with increasing refinement level when the underlying function is smooth [24]. As a result, the thresholding procedure works indirectly also as a detector of discontinuities. In regions where the solution is regular the details become small and are discarded; at discontinuities the solution is not regular, and the details usually stay of the same size (significant) for increasing refinement level.

Prediction. Since the flow field evolves in time, grid adaptivity is performed after each evolution step to provide the adaptive grid updated at the new time level. In order to guarantee the *reliability* of the adaptive scheme in that no

significant future feature of the solution is lost, we have to 'predict' which details will be significant at the new time level $n+1$. *Prediction* can only be based on the information available at the old time level n, i.e., on the set $\mathcal{D}_\varepsilon^n$ of significant details at time level n. If $\tilde{\mathcal{D}}_\varepsilon^{n+1}$ is the prediction set, we want to ensure

$$\mathcal{D}_\varepsilon^n \cup \mathcal{D}_\varepsilon^{n+1} \subset \tilde{\mathcal{D}}_\varepsilon^{n+1}. \tag{34}$$

At the same time, we want to avoid 'overprediction' as much as possible, since it leads to useless computational overhead. We extend the prediction strategy which was used for scalar equations and apply it to each conserved quantity. Details on the prediction process can be found in [22, 24].

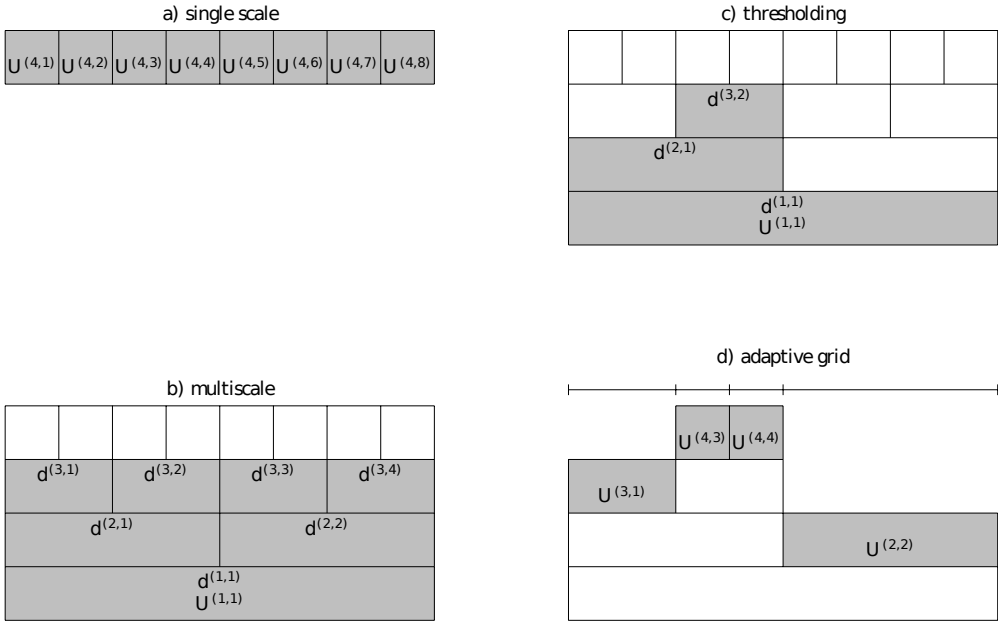


Figure 4: Example of multiwavelet-based grid adaptation on a small set of 4 grids (3 levels of refinement). (a) Single-scale representation $\mathbf{U}^{(L)}$ for $\mathbf{u}_h^{(L)}$ on level $L = 4$. (b) Multiscale representation based on the multiresolution analysis for $\mathbf{u}_h^{(L)}$. (c) Thresholding on the multiscale representation. (d) Construction of the adaptive grid.

Grid Adaptation. By means of the prediction set $\tilde{\mathcal{D}}_\varepsilon^{n+1}$ a *locally refined grid* is determined. For this purpose, we recursively check, proceeding levelwise from coarse to fine, whether there exists a significant detail in a cell. If there is one, we refine the corresponding cell. Time evolution is then performed on the adaptive grid. Figure 4 offers a graphical illustration of the grid adaptivity process: an example of multiwavelet-based grid adaptivity on a small set of $L = 4$ grids (3 levels of refinement) is presented. In the single-scale representation for $\mathbf{u}_h^{(L)}$ on level $L = 4$, expressed by (30), the grey-colored area highlights that the coefficients needed for this representation live just on level L. In the multiscale representation, based on the multiresolution analysis for $\mathbf{u}_h^{(L)}$, as described in (31), the grey-colored area tells us that the coefficients of this representation live on levels $\ell = 1$ to $L-1$. Thresholding is then applied on the multiscale representation through hard thresholding, cf. equation (32). The grey-colored cells correspond to the index set $\mathcal{D}_\varepsilon$, whereas white-colored cells correspond to discarded details. In the construction of the adaptive grid, the grey-colored cells compose the grid resulting from thresholding. By applying inverse multiscale transformation, the single-scale coefficients on the adaptive grid are computed.

Limiting. Multiscale analysis supports limiting strategies in a way tightly related to mesh adaptivity. In regions of very steep gradients, or at discontinuities, mesh refinement in conjunction with lower-order approximation is reasonable. In practical implementation, this corresponds to applying the limiting procedure described in Algorithm Appendix A.1 *only* on the cells of the highest level of refinement L. The underlying assumption is that cells on the finest level of discretization are in regions where the treatment with low-order approximation becomes necessary. This

assumption is supported by the multiresolution analysis, which, as we saw, acts as a discontinuity detector. In cases where this assumption actually holds, one is ensured that limiters are not active in smooth regions, such as non-sonic critical points.

Adaptive Multiresolution-Based DG Scheme. The multiscale transformation (31) and its inverse allow to pass from the single-scale representation on the adaptive grid to the multiscale representation, and vice versa, cf. Figure 4. Though the solution could be evolved in time in both forms, here the single-scale representation coefficients are evolved, since they are also needed explicitly in the limiting process, and since the evaluation of $\mathbf{u}_h^{(L)}$ with respect to the local single-scale functions is cheaper than with respect to the multiwavelet functions. The whole procedure is summed up in Algorithm 3.1. Therein prediction and thresholding are performed on the multiscale coefficients, while all the other operations are carried out on the single-scale coefficients.

Algorithm 3.1. *DG discretization with multiwavelet-based grid adaptivity*

Based on the initial data $\boldsymbol{u}_0$, find the single-scale decomposition (30) *for* $n = 0$
Set $\mathcal{D}_\varepsilon^0$ to contain all the cells on all levels $\ell = 1, \dots, L$
for $n = 1, 2, \dots$ ***do***
    Perform prediction
    Apply inverse multiscale transformation
    for $\sigma = 1, \dots, s$ ***do***
        Apply limiting operator on the highest refinement level $\ell = L$
        Compute residual
        Perform σ-th stage of the Runge-Kutta scheme
    end for
    Apply multiscale transformation
    Apply hard thresholding
end for

Choice of Threshold Value. We focus now our attention on the perturbation analysis, with the goal of controlling the error which is introduced by the adaptive process. The thresholding process generally inserts an error of order ε to the solution. If this additional error is larger than the discretization error of the numerical scheme, it can damage the spatial or the temporal accuracy of the reference scheme. The main idea behind this technique is to accelerate the convergence of the DG scheme without losing accuracy. This means that one should balance the *discretization error* of the reference scheme, i.e., the difference between the exact solution and the reference scheme, and the *perturbation error*, i.e., the difference between the reference DG scheme and the adaptive scheme. Let $\overline{\mathbf{U}}^n_{exact} \in \mathbb{R}^{r \cdot N_E^{(L)}}$ be the projection of the 'exact' solution (a solution computed on a fine enough grid) onto level L, $\overline{\mathbf{U}}^n_{ref,L} \in \mathbb{R}^{r \cdot N_E^{(L)}}$ be the reference non-adaptive DG scheme performed on the uniform grid on level L, and $\overline{\mathbf{U}}^n_{adap,L} \in \mathbb{R}^{r \cdot N_E^{(L)}}$ be the projection of adaptive DG scheme onto level L. The bar indicates that the error analysis is performed on the cell averages, for the same reasons why the convergence analysis is performed for the scheme in the means and then a perturbation argument is applied [22, 40]. Then we define:

- the *discretization error* $\tau_L^n := \overline{\mathbf{U}}^n_{exact} - \overline{\mathbf{U}}^n_{ref,L}$,
- the *perturbation error* $\mathbf{e}_L^n := \overline{\mathbf{U}}^n_{ref,L} - \overline{\mathbf{U}}^n_{adap,L}$, and
- the *full error* $\eta_L^n := \overline{\mathbf{U}}^n_{exact} - \overline{\mathbf{U}}^n_{adap,L}$.

The full error can be estimated by

$$\|\eta_L^n\|_{L^1(\Omega)} \leq \|\tau_L^n\|_{L^1(\Omega)} + \|\mathbf{e}_L^n\|_{L^1(\Omega)} \leq 2\,Tol. \tag{35}$$

The ideal strategy would consist of the following two steps:

1. given the accuracy $2\,Tol$ that one wants to achieve, choose L and N_0 such that $\|\tau_L^n\|_{L^1(\Omega)} \leq Tol$;
2. determine the *optimal threshold value* ε_{opt} such that the perturbation error is bounded by Tol as well.

Thus, the discretization error of the reference scheme and the perturbation error are balanced as in (35). Following this strategy, one may guarantee that the DG scheme on the adaptive grid preserves the accuracy of the DG discretization on the full uniform reference grid. This corresponds to choosing $Tol = \|\tau_L^n\|_{L^1(\Omega)}$, as we will do in the computations in Section 4. Hence we will be looking for ε_{opt} as the largest ε such that $\|\mathbf{e}_L^n\|_{L^1(\Omega)} \leq \|\tau_L^n\|_{L^1(\Omega)}$.

In the scalar case it was proven in that the L^1-norm of the perturbation error can be bounded by a term proportional to $\varepsilon + h_L^\alpha$, where $\alpha > 1$ is a constant coming from stability assumptions on the reference scheme [22]. In practice, however, ε_{opt} is not known a priori. Even in the scalar case where theoretical arguments motivate particular choices of ε, only a rough estimate is available. Empirically, the estimate

$$\varepsilon \sim h_L^\beta \tag{36}$$

gives good results, where the power β can be taken to be 1 for the scalar case [22] and 0.5 for the system case, as we show in the following.

4. Numerical Results

All numerical data in this chapter were produced using a sequential code (C++) for one-dimensional problems. Different test cases were run to validate our method. We chose test cases which have already been used in the literature [41, 42], also in the framework of DG methods [43].

In particular, we solve the time-dependent one-dimensional Euler equations, expressed by (1) with $r = 3$ and

$$\mathbf{u} = \begin{pmatrix} \rho \\ \rho v \\ \rho e_{tot} \end{pmatrix}, \quad \mathbf{f}(\mathbf{u}) = \begin{pmatrix} \rho v \\ \rho v^2 + p \\ (\rho e_{tot} + p)v \end{pmatrix}. \tag{37}$$

The Euler equations express conservation of the mass, of the momentum ρv and of the specific total energy $e_{tot} = e_{kin} + e_{int}$. The variable ρ is the density, the variable v is the velocity in x direction,

$$e_{kin} = \frac{1}{2}v^2 \tag{38}$$

is the specific kinetic energy, and e_{int} the specific internal energy. The variable p is the pressure. The system is closed by means of equation (38) and by a relation defining e_{int} in terms of variables of the system. We assume the gas to be thermally and calorically perfect, so that the equation of state

$$e_{int} = \frac{p}{(\gamma - 1)\rho}, \tag{39}$$

where $\gamma = 1.4$ is the ratio of specific heats for air at standard conditions [44], holds and closes the system.

In Section 4.1 we consider Sod's problem [45], thereby testing the behavior of the method for a classical Riemann problem. Section 4.2 is devoted to the problem of interaction of two blast waves, which was introduced in [46, 47]. Finally, the adaptive procedure applied to the interaction between a moving shock and density sine waves is described in Section 4.3. For every test case we have:

- an 'exact' solution;
- a reference solution on the uniformly refined mesh, as described in Section 2, and
- adaptive solutions for different values of the tolerance ε, computed via the adaptive scheme of Section 3.

For the 'exact' solution we did not use an analytically exact solution, but a solution computed on a very fine mesh.

4.1. Sod's Problem

The first adaptive test case that we consider is Sod's problem [45], where fluids in two different regions of a shock tube have different initial conditions and are separated by a diaphragm, till at time $t = 0$ the diaphragm is broken. We consider $\Omega = [0, 1]$ and take initial conditions:

$$\mathbf{u}_0(x) = \begin{cases} \mathbf{u}_{0,L}, & \text{if } x \leq 0.5, \\ \mathbf{u}_{0,R}, & \text{if } x > 0.5, \end{cases} \tag{40}$$

with

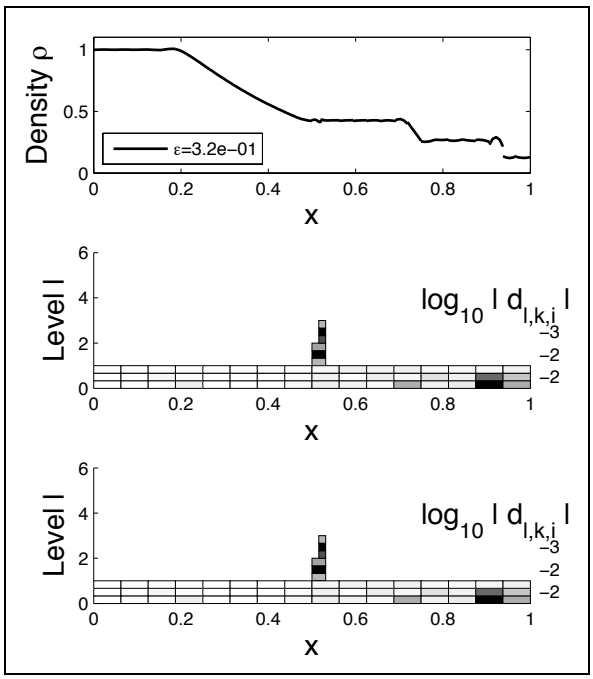
$$\mathbf{u}_{0,L} = \begin{pmatrix} 1 \\ 0 \\ 2.5 \end{pmatrix}, \qquad \mathbf{u}_{0,R} = \begin{pmatrix} 0.125 \\ 0 \\ 0.25 \end{pmatrix}. \tag{41}$$

Constant boundary conditions are imposed at the boundaries.

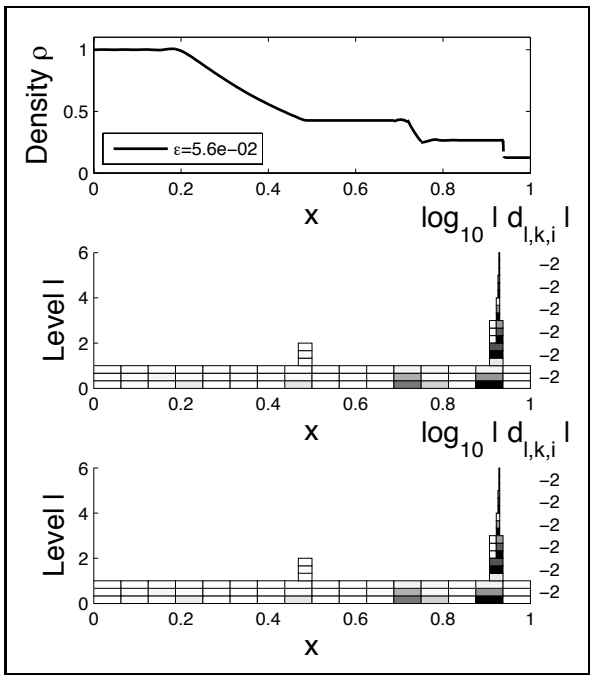
We choose polynomial degree $p = 2$. The numerical flux is determined by Roe's approximate Riemann solver [48]. We choose the CFL number such that the SSPRK(5,4) method is TV-stable, i.e., $CFL = 0.1$. Also, we consider constant $M = 0$ in the limiter, so that a mere TVD limiter is applied. We take $N_E^{(1)} = 16$ cells on the coarsest grid $\mathcal{G}_1$ and consider $L = 7$ nested levels. This corresponds to $1,024$ cells for the reference non-adaptive DG scheme. Computations are performed until $t = 0.25$. We consider different values for ε in the hard thresholding procedure: Figure 5 shows the density profiles for some values of ε and the adaptive grid after thresholding (i.e., after step 12 in Algorithm 3.1), corresponding to the index set $\mathcal{D}_\varepsilon^n$. We show as well the adaptive grid after prediction (i.e., after step 4 in Algorithm 3.1), corresponding to the index set $\tilde{\mathcal{D}}_\varepsilon^n$, on which computations are actually performed. In order to better understand the concept of the adaptive procedure, we also depict the detail coefficients. For each level we represent the cells which are part of the adaptive grid in correspondence of their spatial position. For each cell $V^{(\ell,k)}$ on level ℓ, for each wavelet $i = 1, \ldots, N_2$ (from bottom to top), the absolute value of the maximum detail $\max\limits_{j=1,\ldots,3} \left| \left(\mathbf{d}_i^{(\ell,k)} \right)_j \right|$ is represented by means of a grey scale (different on each level ℓ), where white corresponds to 0 and black to the maximum value on that level. The order of magnitude of $\max\limits_{i=1,\ldots,3} \left| \left(\mathbf{d}_i^{(\ell,k)} \right)_1 \right|$ is given as label on the right axis. We can observe that the details are small and, hence, no grid refinement is triggered when the solution is smooth, whereas large details (black in the figure) appear and the grid is refined up to the highest level in the vicinity of the contact discontinuity and of the shock.

The ultimate goal of our concept is to be *faster* than the reference DG scheme while *preserving* its accuracy. For this purpose we have to find an optimal threshold value ε_{opt} for which the perturbation error introduced by thresholding and the discretization error of the reference scheme performed on the uniform grid on level L are balanced. Here and in the following, the errors are computed for the cell averages on the highest refinement level L in the L^1-norm. We plot in Figure 6 the full error and the perturbation error for different threshold values and the discretization error. The perturbation error is computed on the cell averages of the solution projected onto the *highest* refinement level. The 'exact' solution is here the solution of a DG scheme on a uniform grid with $8,192$ cells, i.e., 3 additional refinement levels compared to our reference scheme. The 'exact' solution is represented in Figure 7. Full error and discretization error are of the same order of magnitude. For $\varepsilon = 0$ the adaptive scheme coincides with the reference scheme, which explains why the full error is bounded from below by the discretization error. In Figure 6(b) the perturbation error is dependent on ε, becoming smaller and smaller for small tolerance values. The growth of the perturbation error, and consequently of the growth of the full error (see Figure 6(a)), is not monotonic with ε. This is in particular true for rather large values of the tolerance. There the threshold value is already so large that important features of the solution are not captured. We can see that for all the chosen values of ε smaller than 0.01 the perturbation error is smaller than the discretization error, i.e., grid adaptivity causes no loss of accuracy. This means that the optimal estimated value of the threshold is $\varepsilon_{opt} = 0.01$, where by 'optimal' we mean most efficient and yet reliable. The value 0.01 is also the largest value of ε such that the full error is smaller than $2\,Tol$, as required in (35), see Figure 6(a).

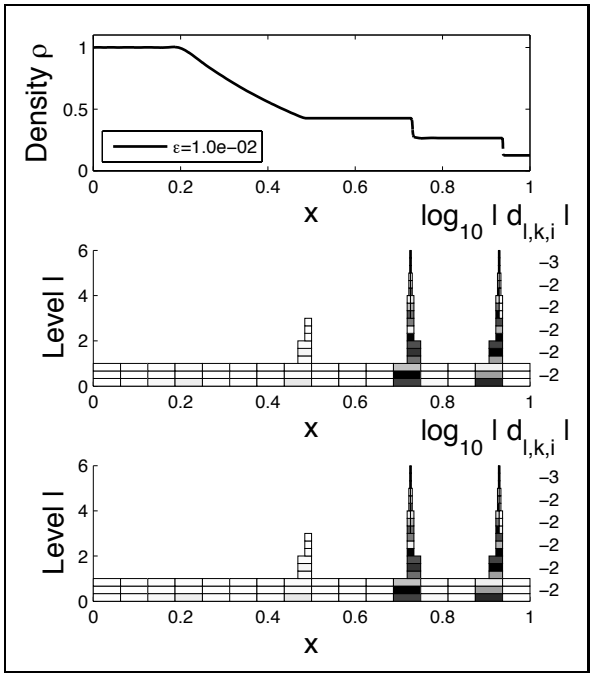
For the optimal value $\varepsilon_{opt} = 0.01$, we plot the polynomial approximation of the adaptive solution together with the reference solution and the 'exact' solution, see Figure 8. In the contact discontinuity all the solutions shows some overshoots, see Figure 8. The overshoots do not penalize the global accuracy of our approximation. If one would



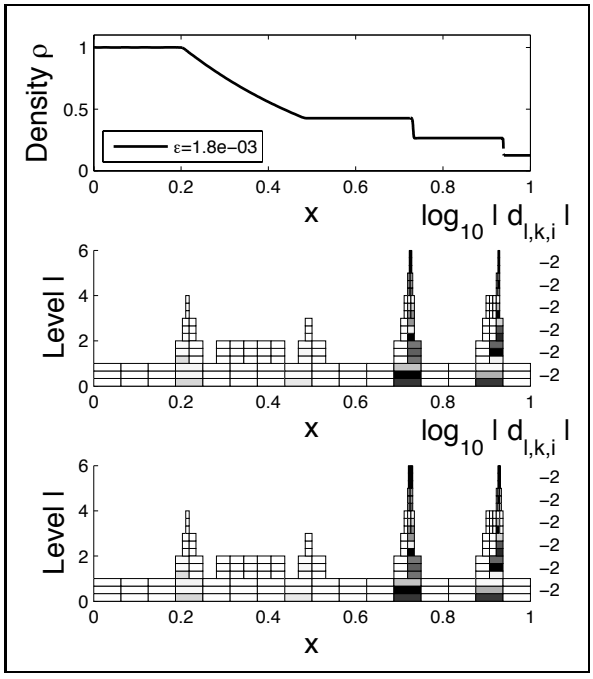
(a) $\varepsilon = 0.32$



(b) $\varepsilon = 0.056$

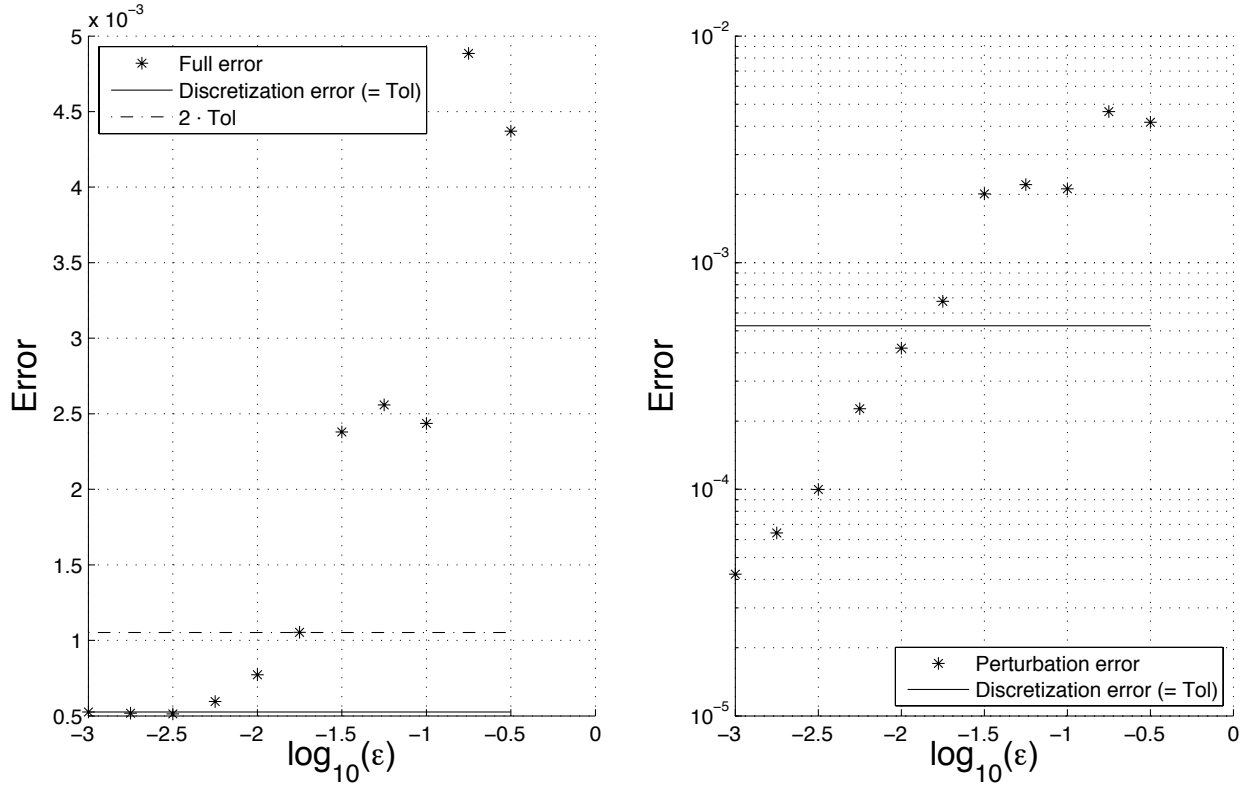


(c) $\varepsilon = 0.010$



(d) $\varepsilon = 0.0018$

Figure 5: Sod's test case with $p = 2$ at time $t = 0.25$ for different values of ε. Top: density profile. Center: representation of the significant details after thresholding ($\mathcal{D}_\varepsilon^n$). Bottom: representation of the significant details after prediction ($\tilde{\mathcal{D}}_\varepsilon^n$).



(a) Full error and discretization error vs. tolerance ε

(b) Perturbation and discretization error vs. tolerance ε

Figure 6: Sod's test case: error analysis.

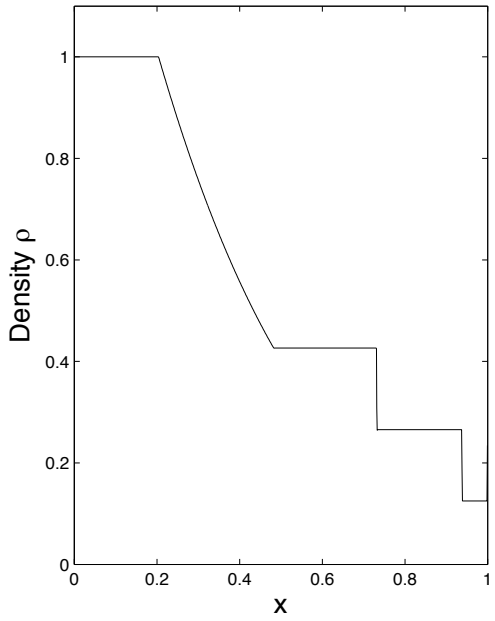
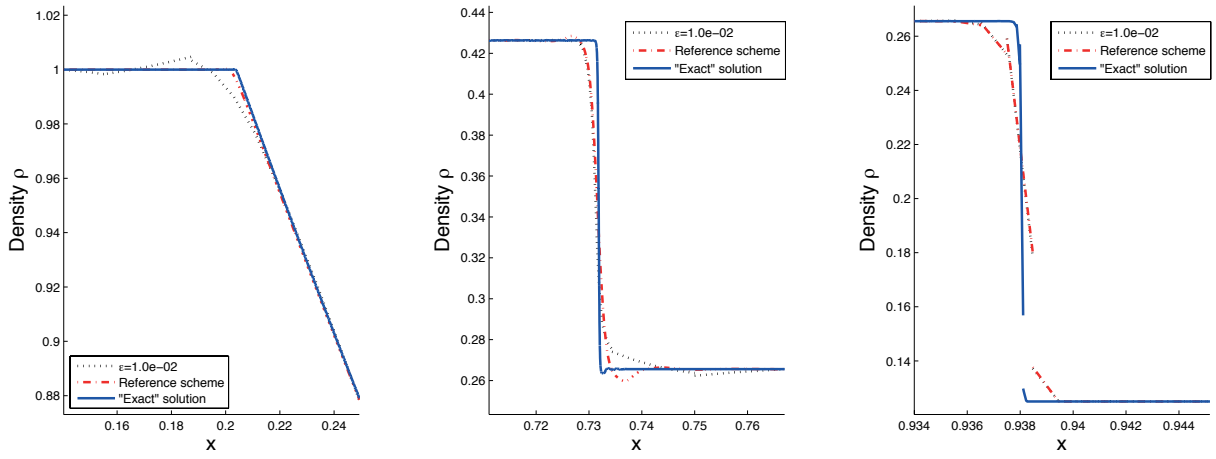


Figure 7: Sod's test case: density profile for the 'exact' solution at time $t = 0.25$.

want to get rid of them, another numerical flux function or larger L should be chosen, so as to improve the quality of the underlying reference solution. As concerns the shock wave, the resolution is bounded by the application of the limiter.



(a) At the expansion wave (b) At the contact discontinuity (c) At the shock wave

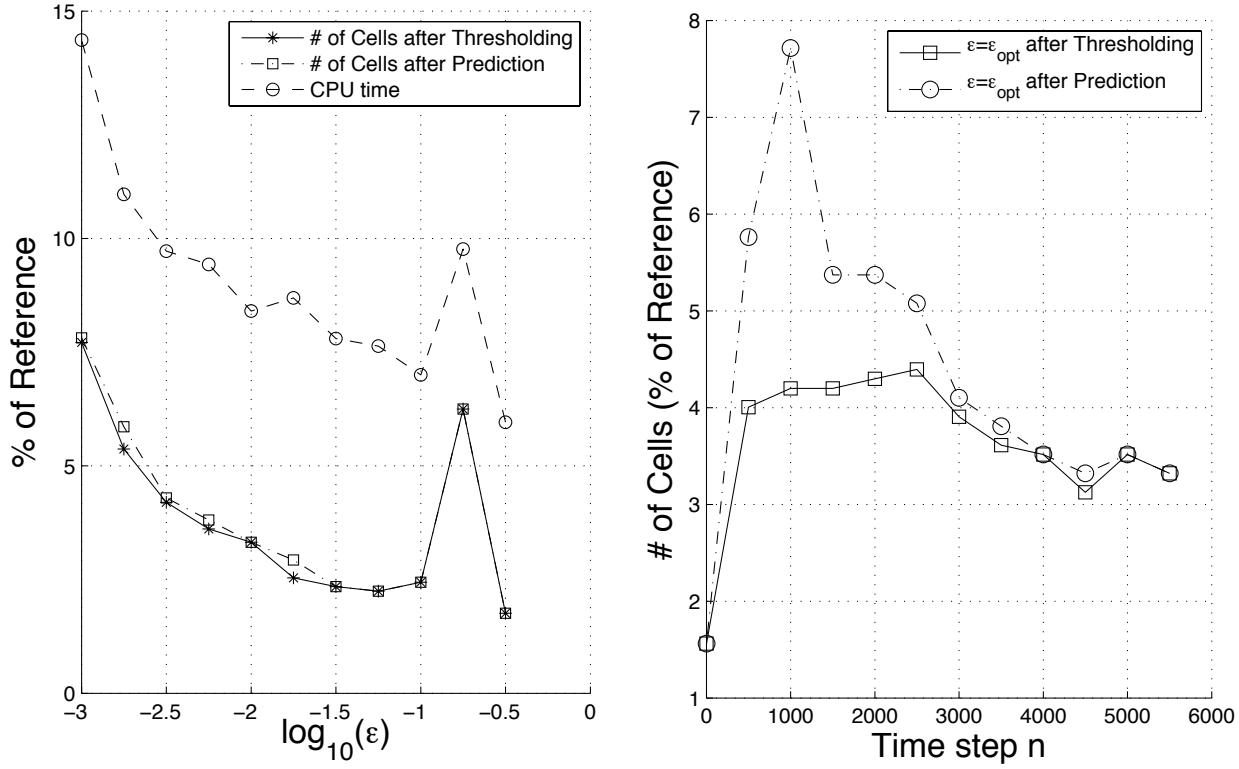
Figure 8: Sod's test case: comparison of the adaptive solution for $\varepsilon_{opt} = 0.01$, the reference non-adaptive solution and the 'exact' solution at time $t = 0.25$. Zoomed-in details of the density profile.

It should be emphasized that our grid adaptivity procedure is able to accelerate the convergence of the reference DG scheme, but at the same time *cannot* improve the quality of the reference scheme. Hence, the smeared capture of contact discontinuities in the reference scheme cannot be overcome by the grid adaptivity. Improvements in that respect would require the implementation of a better underlying DG scheme.

The efficiency essentially depends on the thresholding and prediction steps. In particular, the sparser the adaptive grid, the less DOFs have to be evolved in time. Both the adaptive grid right after thresholding and the adaptive grid after prediction are represented in Figure 5. They correspond to the index set $\mathcal{D}_\varepsilon^n$ of equation (33) and to the index set $\tilde{\mathcal{D}}_\varepsilon^n$, respectively. The adaptive grid after prediction is the grid on which computations are actually performed. In the ideal case, where the prediction procedure selects precisely the cells which will show significant details after time evolution and limiting, the adaptive grid after prediction and the adaptive grid after thresholding coincide, i.e., $\mathcal{D}_\varepsilon^n = \tilde{\mathcal{D}}_\varepsilon^n$. This is however not always the case. Even though in the first part of the computation the prediction procedure inflates the set of significant details up to twice the number of cells in the thresholded set of details, see Figure 9(b), the same procedure is very sharp when the flow is well resolved, as in the final time step, see Figures 5 and 9(a). In particular, the optimal value of the threshold $\varepsilon_{opt} = 0.01$ corresponds to a grid with less than 8 percent of the cells in the reference non-adaptive DG scheme. Our approach shows clearly its efficiency: a compression rate of about 12 is achieved. This corresponds to a computational time that is less than 9 percent of the computational time of the reference scheme. Moreover, it should be noted that the CPU time and compression ratio are not very sensitive to the choice of ε, so that significant gain is still obtained even if one takes a value which is smaller of 1 or 2 orders of magnitude.

4.2. Interaction of Two Blast Waves

We consider a problem involving the generation and interaction of extremely strong nonlinear waves in one dimension: the interaction of two blast waves. Shocks are considered to be *strong* if there is substantial entropy production. We consider a fluid at rest in the domain $\Omega = [0, 1]$. Two regions of width 0.1 near the two reflecting walls, i.e., the



(a) Relative number of cells in the final adaptive grid and relative CPU time vs. ε

(b) Number of cells in the adaptive grid vs. number of time steps for $\varepsilon = \varepsilon_{opt}$

Figure 9: Sod's test case: data compression and speed-up.

regions $[0, 0.1]$ and $[0.9, 1]$, contain hot gas. Namely, we take initial conditions:

$$\mathbf{u}_0(x) = \begin{cases} \mathbf{u}_{0,L}, & \text{if } x \leq 0.1, \\ \mathbf{u}_{0,C}, & \text{if } 0.1 < x \leq 0.9, \\ \mathbf{u}_{0,R}, & \text{if } x > 0.9, \end{cases} \tag{42}$$

with

$$\mathbf{u}_{0,L} = \begin{pmatrix} 1 \\ 0 \\ 2500 \end{pmatrix}, \qquad \mathbf{u}_{0,C} = \begin{pmatrix} 1 \\ 0 \\ 0.025 \end{pmatrix}, \qquad \mathbf{u}_{0,R} = \begin{pmatrix} 1 \\ 0 \\ 250 \end{pmatrix}, \tag{43}$$

corresponding to pressure values $p_L = 1,000$, $p_c = 0.01$ and $p_R = 100$. This test case was introduced by Woodward in [46] and in [47] the evolution of the problem is described in detail. The main development is represented by two strong blast waves which form at time $t = 0$ and then, when they collide, they produce a new contact discontinuity. For the approximation we employ again $p = 2$.

We focus on the solution at time $t = 0.038$, when the interaction between the two blast waves has already taken place. In this and the following computations, the numerical flux is the HLLC approximate Riemann solver [49], which comes from a modification of the HLL solver, as in [50]. As a constant for the limiter we have $M = 10,000$ in this case. We will comment on this below. We choose CFL number 0.1 and use again the explicit SSPRK(5,4) method for time relaxation. First of all we consider the 'exact' solution, namely the solution computed with a DG scheme on a very refined, uniform grid with 6,656 cells, reported in Figure 10. In the density profile one can recognize starting

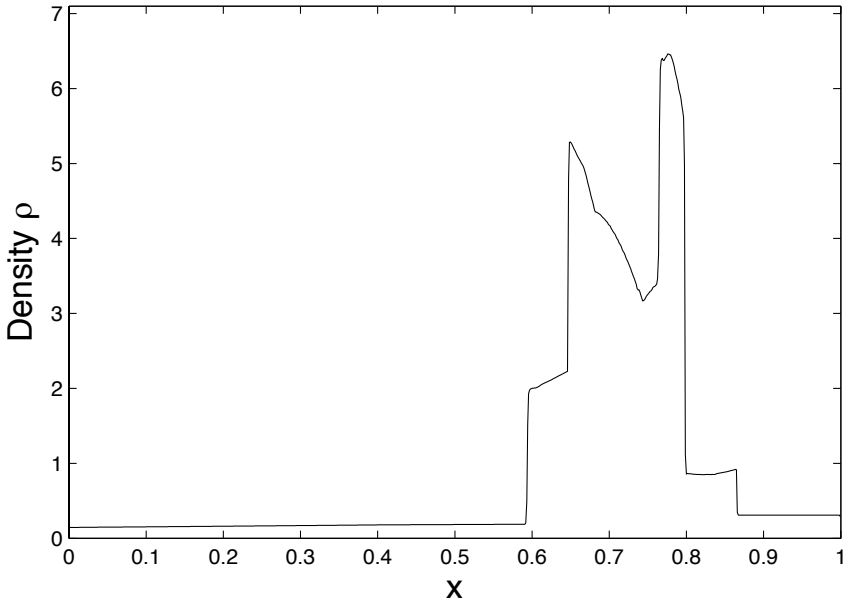


Figure 10: Interaction of two blast waves: density profile for the 'exact' solution at time $t = 0.038$.

from the left first a strong contact discontinuity at about $x = 0.6$, which was originated by the blast wave at $x = 0.1$ and is now traveling to the right. Then, at about $x = 0.65$, there is the reflected component of the left shock (the one which formed at $x = 0.1$) which is now traveling left. The newly created contact discontinuity follows at about $x = 0.76$ and, on the right of it, the reflected components of the right contact discontinuity and of the right shock can be seen at about $x = 0.8$ and $x = 0.87$, respectively. This is a challenging problem, both because of the strength of the shocks and the interactions of different waves.

Zooming in on the 'exact' solution one can observe the phenomenon of *postshock oscillations*, see Figure 12. Low-amplitude non-physical oscillations can be observed. Shock capturing schemes may generate oscillations in the

form of persistent wavy tails attached to the shock front: [47, 51, 52] offer an interesting analysis of this phenomenon. These oscillations are present in both low- and high-order schemes, but appear more clearly in high-order schemes because they are less dissipative. Though some remedies have been tried in the literature [53], to the present time the problem of postshock oscillations has not yet found satisfactory solution. In [51] Arora and Roe even conjectured that sharp shock capturing leads to unavoidable oscillations. The presence of postshock oscillations affects also the reference solution, see Figure 12. Since the adaptive method proposed here only aims at approximating the reference solution by means of a coarser grid, it cannot fix the problems of the underlying reference scheme. Hence, postshock oscillations will affect the quality of the adaptive solutions as well, as we will see below.

We analyze the solutions and their corresponding adaptive grids for different values of the tolerance ε (see Figure 11). We allow for 6 levels of refinement, with $N_1 = 13$ cells on the coarsest grid $\mathcal{G}_1$. Focusing on Figure 11, and in particular on the adaptive grid with $\varepsilon = 0.0018$, one notices that the grid is intensely refined in the area $[0.6, 0.85]$ of the 'density slab', where the solution should be fairly smooth. The presence of postshock oscillations triggers grid refinement, because the numerical solution in those regions has an information content which differs from the smooth nature of the exact solution.

The massive presence of postshock oscillations has some important consequences. The prediction procedure sees these oscillations as 'information', therefore refinement is triggered and the discretization error is very large. All the values smaller than 0.056 chosen for the tolerance ε correspond to a perturbation error smaller than the discretization error. It is not possible to choose values for ε larger than 0.18, because that leads to too coarse grids, unable to capture the features of the solution (the method becomes unstable). In Figure 13 the behavior of the perturbation error is depicted. In this case, the optimal estimated value for the threshold would be $\varepsilon_{opt} = 0.056$.

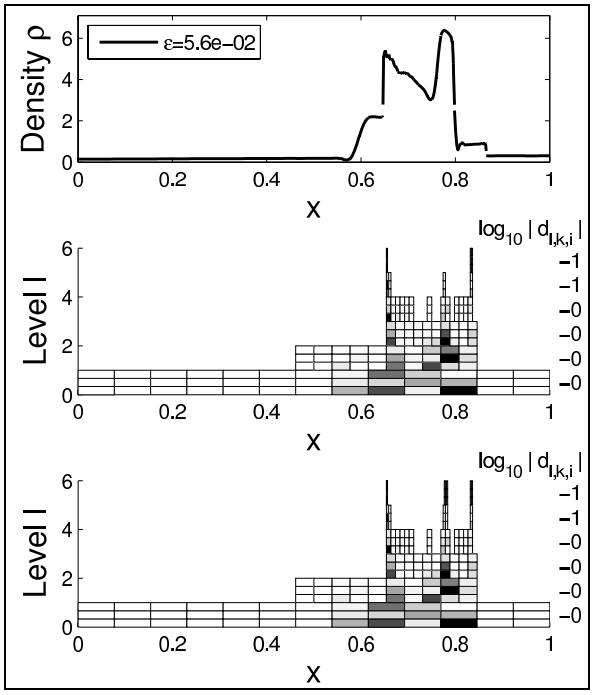
In Figure 14(a) we plot the behavior of the number of cells at the final time step and of the CPU time with respect to the tolerance ε. As one would expect, the number of cells decreases for larger ε, since the grid is less refined. In Figure 14(b) we also study the variation in the number of cells with time. The qualitative behavior of the number of cells after prediction and the behavior of the number of cells after thresholding are similar. We are in the ideal case: the two adaptive grids (after thresholding and after prediction) are almost indistinguishable. This can be clearly observed in Figure 14: prediction gives here an extremely sharp estimate of the set of significant details. In spite of the low quality of the solution given by the reference DG scheme, the adaptive procedure performs very well. If we consider the optimal value $\varepsilon = 0.056$, less than 8 percent of the cells are used in the adaptive procedure with respect to the reference DG scheme performed on the uniform grid.

It is worth to recall that our approach pursues the goal of approximating in the best possible way the reference solution, by means of a coarser grid. If the underlying reference method produces a somehow poor solution to the physical problem, the adaptive procedure cannot recover the exact solution. In this respect, the solution with tolerance $\varepsilon = 0.056$ shows a very good agreement with the reference solution and this with a reduction by a factor of 0.08 in DOFs and by a factor of 0.13 in computational time.

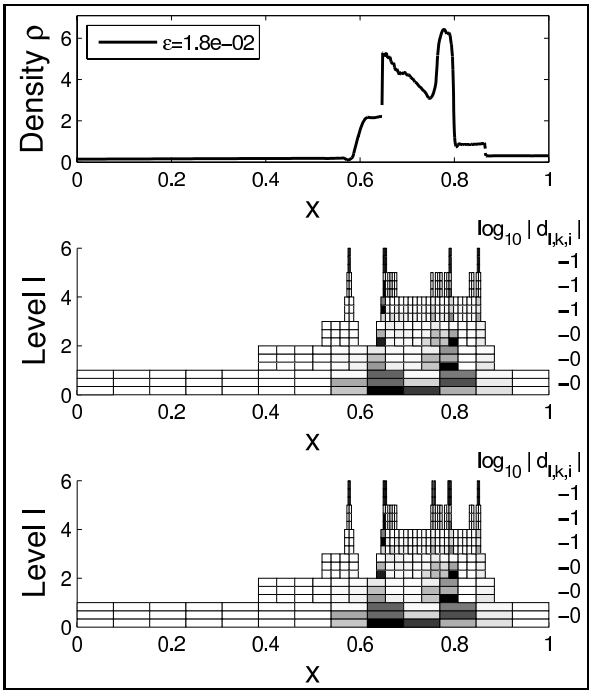
The prediction procedure in our adaptive concept is set to refine the grid in a certain point of the domain up to highest level of refinement L if it is 'predicted' that the limiter will be applied in that point in the following time step. The reason is that, as we mentioned, limiter is applied only on level L. In our experience, this concept works very well everywhere but at nonsonic critical points. At nonsonic critical points this affects the efficiency of the prediction procedure: the adaptive grid is inflated excessively and unnecessarily ('overprediction'). In fact, at the following time step, multiresolution analysis would then reveal that the solution is actually smooth, and the cells resulting from the refinement up to level L would be discarded. To avoid overprediction, it is crucial to choose accurately the constant M, so that no limiter is applied at nonsonic critical points. This is the reason which determined our choice for the constant M in (A.3). A value $M > 0$ corresponds to applying the limiter only for discontinuities which are 'large enough', according to condition (A.3). Future development of the multiresolution-based adaptive concept should include the implementation of a better alternative to limiting, e.g., a vanishing viscosity method.

4.3. Interaction of a Moving Shock with Density Sine Waves

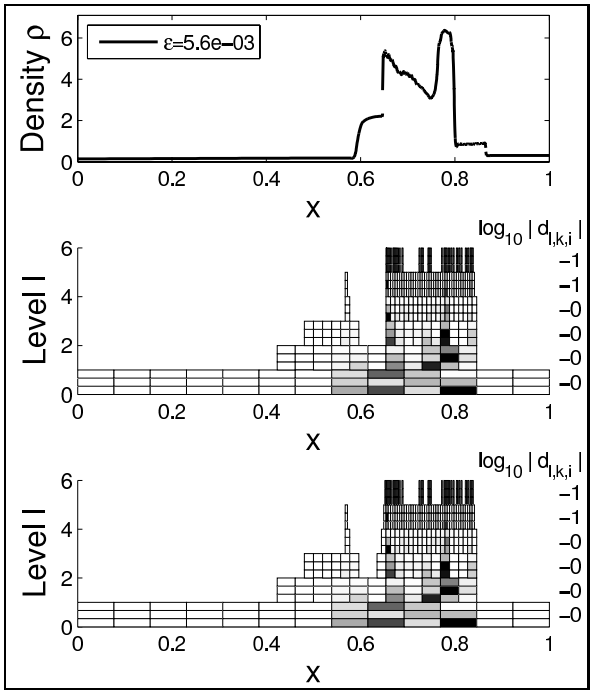
High-order schemes become particularly useful when the problem has some structure in the smooth part of the solution. In order to see this, we consider a test case which involves both shocks and some structure in the solution and choose polynomial degree $p = 3$. We take a test case from [54], with initial conditions including a jump in the



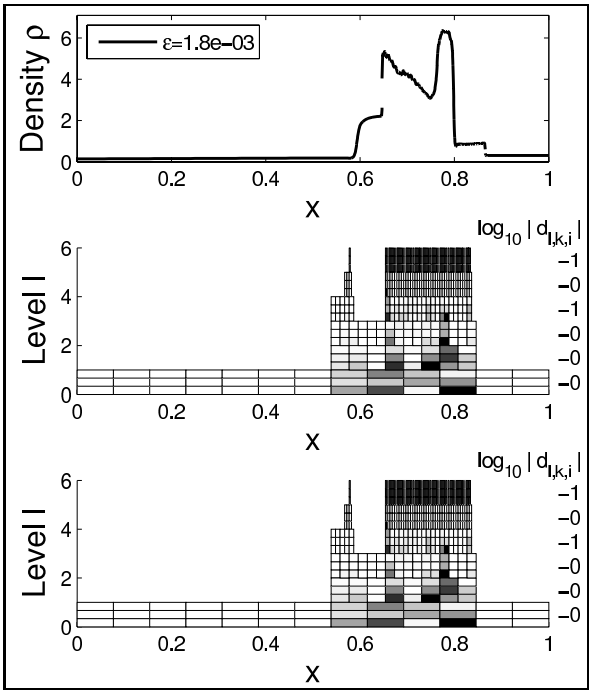
(a) $\varepsilon = 0.056$



(b) $\varepsilon = 0.018$



(c) $\varepsilon = 0.0056$



(d) $\varepsilon = 0.0018$

Figure 11: Interaction of two blast waves at time $t = 0.038$. Top: density profile. Center: representation of the significant details after thresholding ($\mathcal{D}_\varepsilon^n$). Bottom: representation of the significant details after prediction ($\tilde{\mathcal{D}}_\varepsilon^n$).

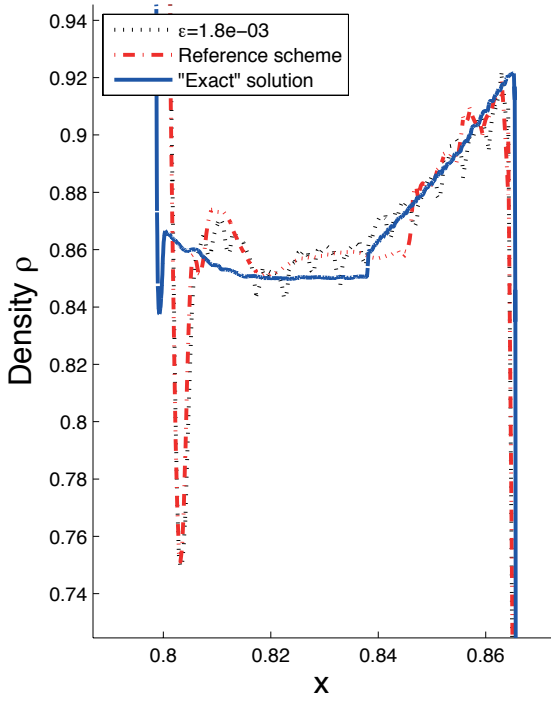
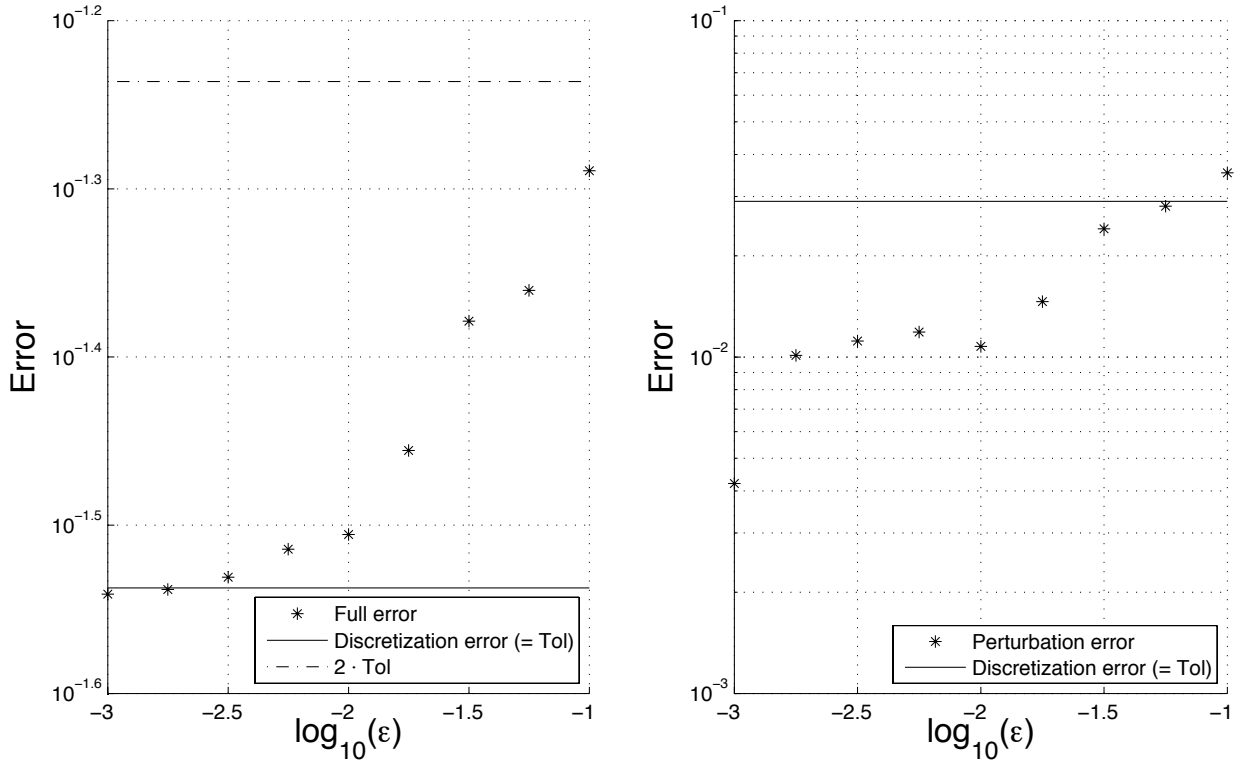


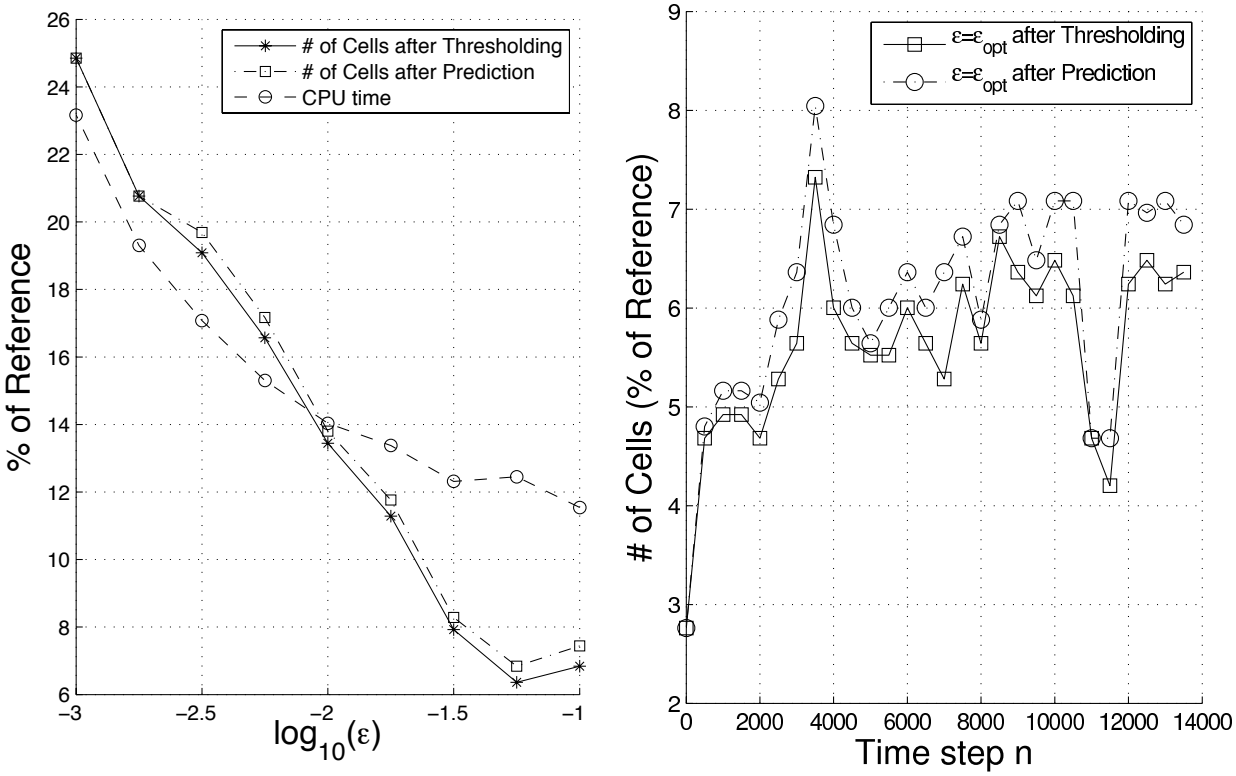
Figure 12: Interaction of two blast waves at time $t = 0.038$. Detail of postshock oscllations.



(a) Full error and discretization error vs. tolerance ε

(b) Perturbation and discretization error vs. tolerance ε

Figure 13: Interaction of two blast waves at time $t = 0.038$: error analysis.



(a) Relative number of cells in the final adaptive grid and relative CPU time vs. ε

(b) Number of cells in the adaptive grid vs. number of time steps for $\varepsilon = \varepsilon_{opt}$

Figure 14: Interaction of two blast waves: data compression and speed-up.

initial quantities and sinusoidal behavior of the density. Again, we consider the one-dimensional Euler equations with

$$\mathbf{u}_0(x) = \begin{cases} \mathbf{u}_{0,L}, & \text{if } x \leq -4, \\ \mathbf{u}_{0,R}(x), & \text{if } x > -4, \end{cases} \tag{44}$$

with conditions derived from

$$\begin{array}{ll} \rho_{0,L} = 3.857143, & \rho_{0,R}(x) = 1 + 0.2\sin(5x), \\ u_{0,L} = 2.629369, & u_{0,R} = 0, \\ p_{0,L} = 10.33333, & p_{0,R} = 1. \end{array} \tag{45}$$

The shock is moving to the right. We consider constant boundary conditions on the domain $\Omega = [-5, 5]$. Globally 6 refinement levels are considered, with $N_E^{(1)} = 20$ cells on the coarsest grid $\mathcal{G}_1$. This corresponds to a uniform grid with 1,280 cells for the non-adaptive reference DG scheme. In this computation, the numerical flux is the HLLC approximate Riemann solver. The constant in the limiter (A.3) was chosen as $M = 1,000$. We choose CFL number 0.1 and the explicit SSPRK(5,4) method.

Figure 15 shows the 'exact' solution, obtained on a uniform grid with $10,240$ cells. The adaptive solution for

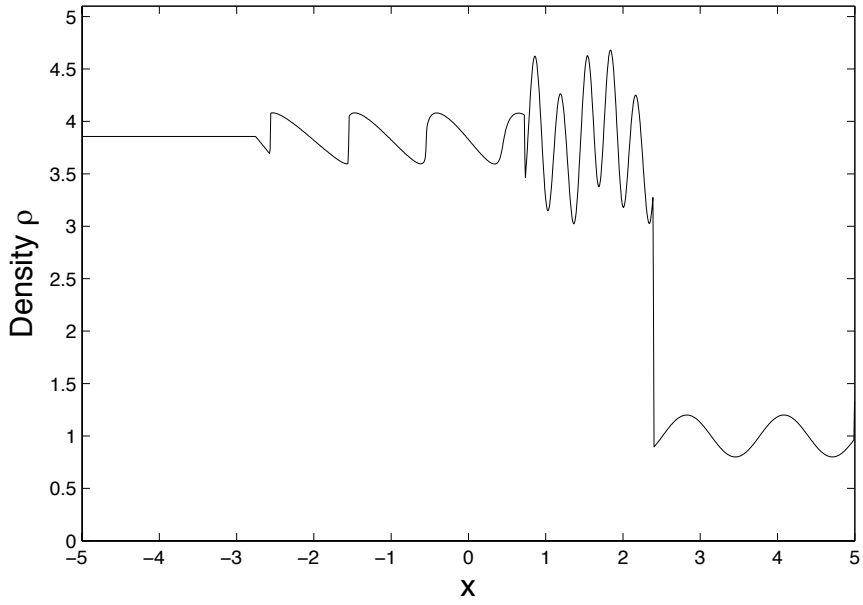
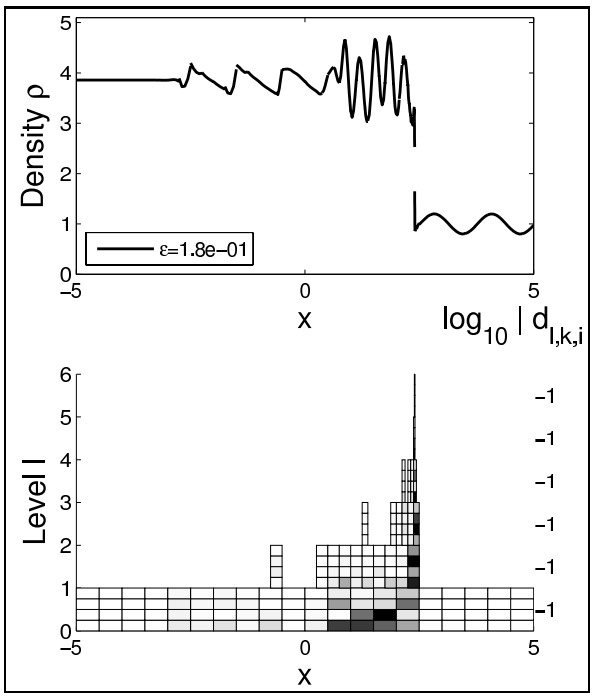


Figure 15: Interaction of a moving shock with density sine waves at time $t = 1.8$: density profile for the 'exact' solution.

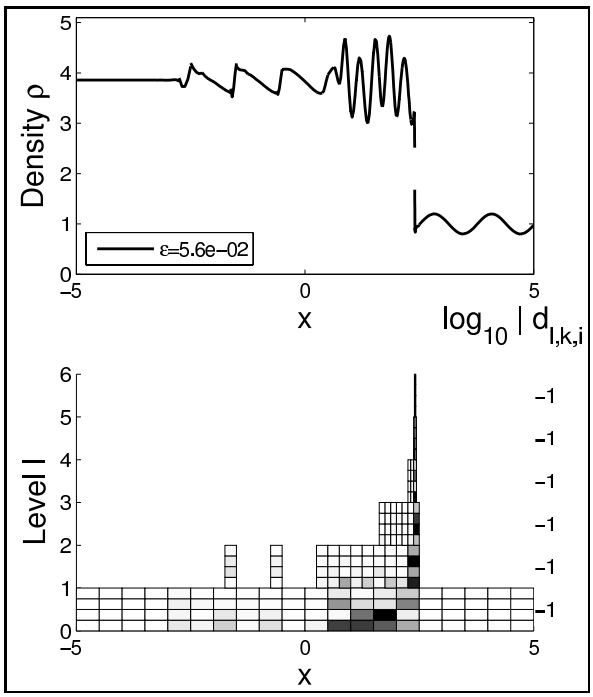
different levels of ε is shown in Figure 16. Grid refinement is triggered both at shocks and in the region where the solution is of sinusoidal type. For smaller values of ε the grid is refined also in correspondence of the other jumps in the solution.

We then examine the behavior of the errors in Figure 17. All the values considered for ε give a perturbation error very close to the discretization error. $\varepsilon_{opt} = 0.056$ is the value which needs the smallest number of DOFs and yet produces a perturbation error smaller than the discretization error.

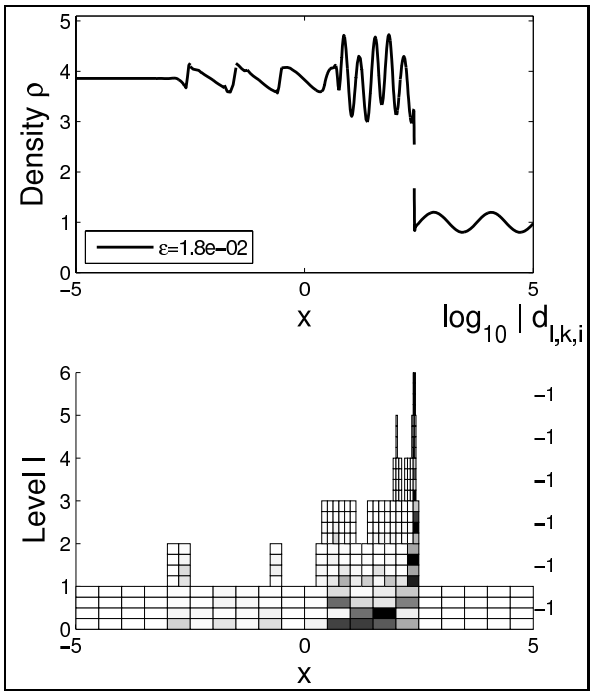
We report in Figure 18 the number of cells in the adaptive grids, scaled with the constant number of cells on the reference discretization. We are in an almost ideal case: predicted adaptive grid and thresholded grid do not differ much in the number of cells. For the optimal value $\varepsilon_{opt} = 0.056$ the number of DOFs in the adaptive grid is less than 4 percent of the number of DOFs in the reference non-adaptive solution, corresponding to a speed-up of more than



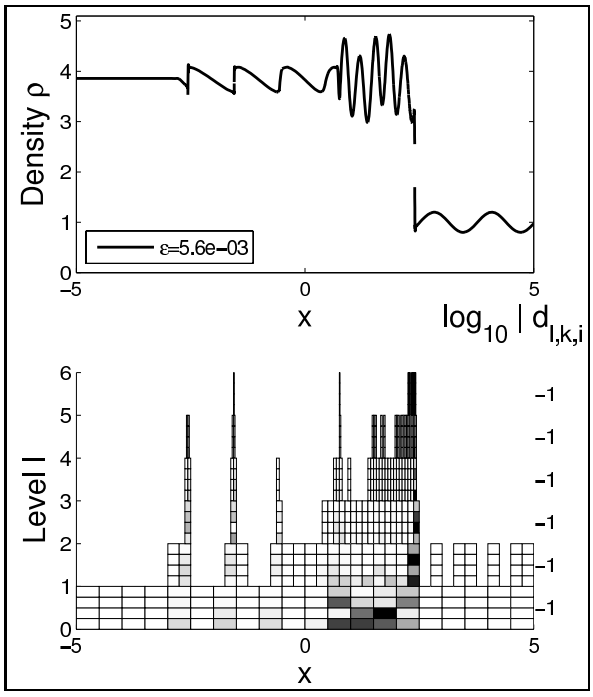
(a) $\varepsilon = 0.18$



(b) $\varepsilon = 0.056$

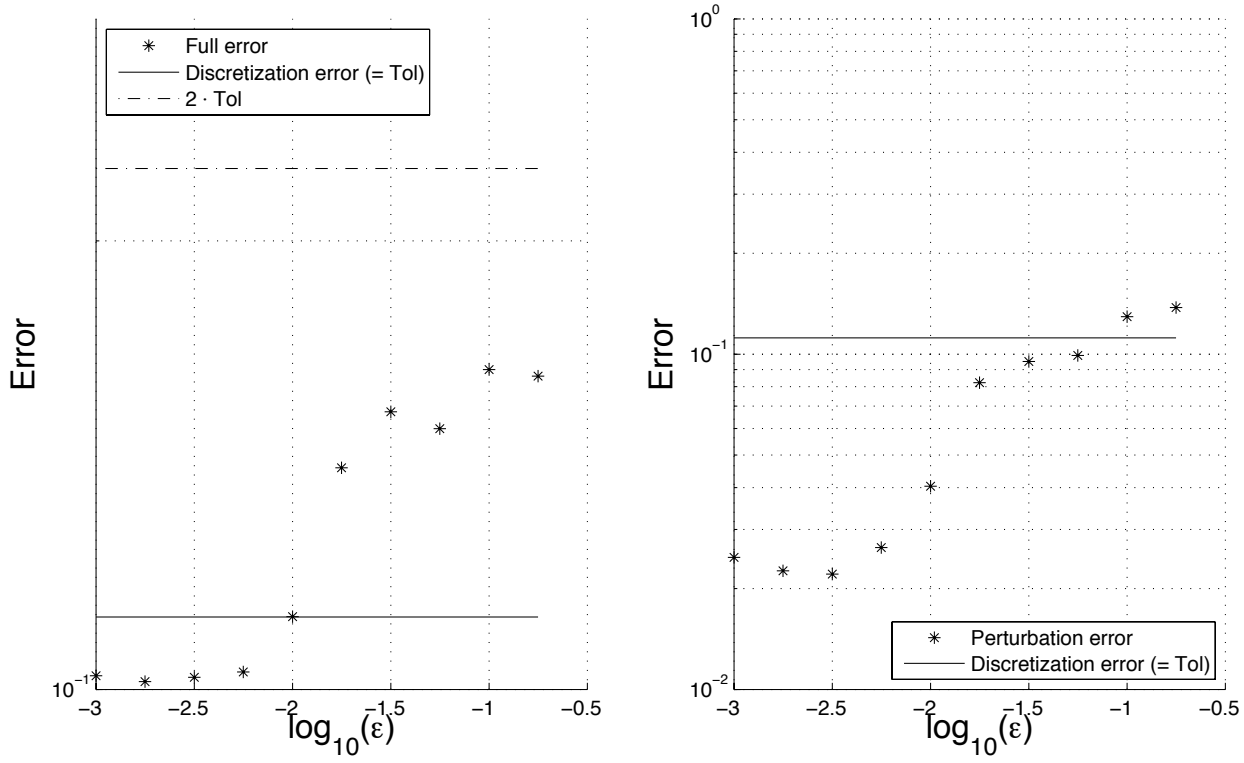


(c) $\varepsilon = 0.018$



(d) $\varepsilon = 0.0056$

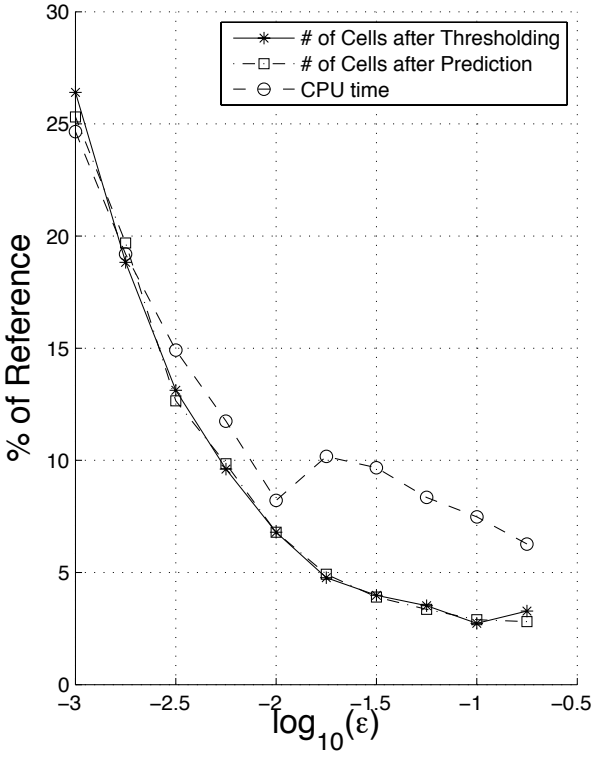
Figure 16: Interaction of a moving shock with density sine waves with $p = 3$ at time $t = 1.8$. Top: density profile. Bottom: representation of the significant details after thresholding ($\mathcal{D}_\varepsilon^u$).



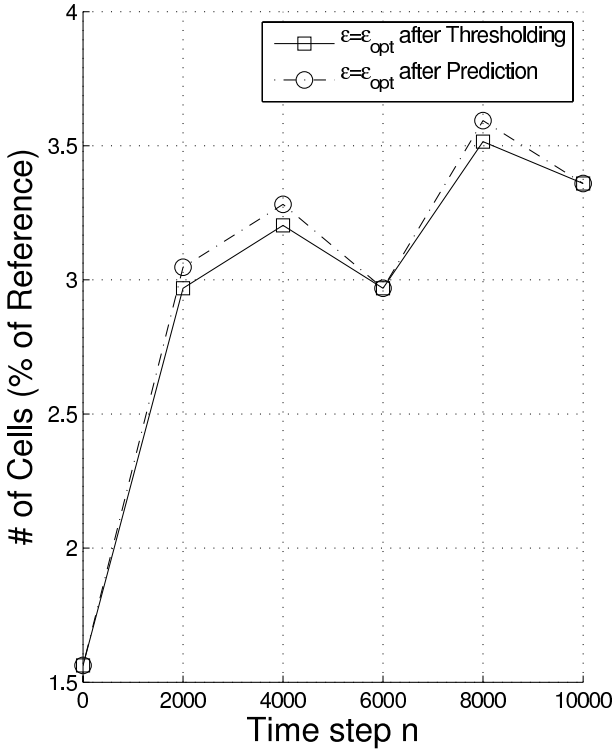
(a) Full error and discretization error vs. tolerance ε

(b) Perturbation and discretization error vs. tolerance ε

Figure 17: Interaction of a moving shock with density sine waves at time $t = 1.8$: error analysis.



(a) Relative number of cells in the final adaptive grid and relative CPU time vs. ε



(b) Number of cells in the adaptive grid vs. number of time steps for $\varepsilon = \varepsilon_{opt}$

Figure 18: Interaction of a moving shock with density sine waves: data compression and speed-up.

11. In other words, the adaptive computation is more than 11 times faster than the reference computation, and yet it results in the same error.

We recall that high-order FV discretizations typically need a much more refined grid for this configuration, since they employ large stencils to reach high order. This is the reason why for high-order discretization the DG discretization has been adopted. If one used high-order FV schemes, the gain given by the multiresolution-based grid coarsening would be lost in large part. In particular, a quantitative comparison between adaptive FV and adaptive DG schemes can be found in [24].

5. Discussion and Conclusion

We have applied the multiwavelet-based grid adaptivity procedure to one-dimensional systems of conservation laws. In Table 1 we summarize the run test cases, reporting the polynomial degree p, the number L of grids, the number $N_E^{(1)}$ of cells on the coarsest grid $\mathcal{G}_1$, the number N_{ref} of cells on the grid for the reference non-adaptive DG scheme, the cell size h_L on the highest refinement level, the optimal threshold value ε_{opt} and the discretization error $\|\tau_L^n\|_{\ell^1}$. Table 2 lists for the optimal value $\varepsilon = \varepsilon_{opt}$, the perturbation error, the relative number of cells in the adaptive grid and the relative CPU time with respect to the reference scheme.

The performed test cases showed that adaptive computations for optimal values of ε have less than 8 percent of the DOFs of the reference non-adaptive scheme. This remains true also for values of ε close enough to ε_{opt}, since the number of cells is not too sensitive to this parameter. We try to quantify the expression 'close enough'. We consider the interaction of two blast waves, which is the test cases where the compression ratio depends more strongly on

Test Case of Section:	p	L	$N_E^{(1)}$	N_{ref}	h_L	$\sqrt{h_L}$	ε_{opt}	$\|\tau_L^n\|_1$
4.1	2	7	16	1,024	~ 0.001	~ 0.03	0.01	0.0005
4.2	2	7	13	832	~ 0.001	~ 0.03	0.056	0.0290
4.3	3	7	20	1,280	~ 0.008	~ 0.09	0.056	0.1118

Table 1: Adaptive DG discretization: summary of the run test cases.

Test Case of Section:	$\|\mathbf{e}_L^n\|_1$	$\frac{N_{adap}(\varepsilon_{opt})}{N_{ref}}$	$\frac{CPU_{adap}(\varepsilon_{opt})}{CPU_{ref}}$
4.1	0.0004	< 8 %	< 9 %
4.2	0.0281	< 8 %	< 13 %
4.3	0.0991	< 4 %	< 9 %

Table 2: Adaptive DG discretization for $\varepsilon = \varepsilon_{opt}$: perturbation error, data compression, and CPU time.

the threshold value ε, see Figure 14(a). Even using a value of ε which is 40 percent larger or smaller than ε_{opt}, the adaptation procedure delivers an adaptive scheme with less than 8 percent of the DOFs of the reference scheme.

The error analysis showed that also in the case of systems of equations the prediction procedure is 'safe', meaning that no significant information is lost. In the scalar case it has been proven analytically that the L^1-norm of the perturbation error can be bounded by a term proportional to h_L^β, where $\beta = \min(1, \alpha)/2$ and α is a constant coming from stability assumptions on the reference scheme [22]. This estimate is, however, very pessimistic. For scalar problems numerous computations have been performed [24], from which it can be concluded that $\varepsilon \sim h_L$ (namely, $\beta = 1$) is a reasonable choice. In our computations the choice of the threshold value as

$$\varepsilon \sim \sqrt{h_L} \tag{46}$$

seems to be more appropriate, and still in agreement with the theoretical result for the scalar case, see Table 1.

Future necessary developments include the implementation of convergence acceleration techniques for steady problems (e.g., local time stepping) and of implicit relaxation techniques, and the extension to viscous systems (Navier-Stokes equations). The compact stencil of the spatial discretization eases the incorporation of p-adaptivity, for instance of the type described in [24]. None of the conceptual ingredients are restricted to the framework of one-dimensional systems. The extension to multidimensional systems of conservation laws would be based on tensor products for hexahedral elements, and on generalizations of the multiwavelets [55] for simplices.

Starting from the same remarks that we collected in Section 1, other adaptive strategies have been developed which share some of the features of our approach [28–33]. One example is given by the so-called *modified wavelet* method [30–32], a method based on the sparse point representation [56]. This technique, limited to structured grids, uses the compressed data set in order to decide in which cells flux evaluations should be performed exactly [24]. At the cells excluded from the adaptive data set the flow variables are just interpolated. Then, time integration is performed. There the prediction strategy is however just empirical: no actual guarantee is given on the retainment of accuracy. The method has been tested on a two-dimensional time-dependent Euler problem (shock-vortex interaction) in [31]: a compression ratio up to 8 is obtained, which corresponds to at least 12 percent of cells with respect to the uniform reference scheme. The whole procedure is rather expensive, since this translates into a computational time which is at least 50 percent of the reference time. One reason is that time integration is performed for all DOFs.

It is worth considering yet another approach for its close similarity to ours, pointing out analogies and differences. In [29] Shelton presents a multiresolution DG method for unsteady compressible flows. There but the same Alpert's multiwavelets are adopted in the multiresolution analysis. As regards thresholding the same test as in (32) is performed, though using the ℓ^2-norm. Besides, therein a different level-dependent threshold value is used. All the cells with significant details and their direct neighbors on the same level are retained in the adaptive grid. We suspect

that this inflation of the set of significant details is needed to take into account time advancement: in fact, no prediction procedure is mentioned. Shelton's 'reliable' choice for the threshold value is $\varepsilon \sim h_L^{p-\frac{1}{2}}$, though no error control analysis is given. This choice would be far too pessimistic in our context, but is probably necessary in his setting in order not to miss important features of the solution (due to lack of prediction procedure). An interesting aspect of this approach is the use of scale transition algorithms to transfer matrix operators of the reference scheme onto the multiresolution-based DG scheme. This is however not necessary in our context, since the underlying scheme is a classical DG method, which acts on the single-scale coefficients and not on the details.

Shelton tested his approach on a few one- and two-dimensional inviscid flows, obtaining compression ratios N_{adap}/N_{ref} varying between 5 and 25 percent. In particular, for Sod's test case he obtained a compression ratio of about 30 percent, including 4 times more cells than we do with our approach for optimal ε_{opt}. Regarding computational time, though the initial rate of reduction in the wall clock time is roughly the same for different values of p, it deteriorates faster for high orders of consistency, probably due to the overhead of the operators and the encoding/decoding algorithms. An important aspect to keep in mind is that no direct comparison can be done with our approach, since in [29] no guarantee is given on the accuracy of the solution and only qualitative comparisons were presented. This seems to confirm the strength of our approach, which yields consistent gain in terms both of compression and speed-up with a viable heuristic to estimate an appropriate threshold value.

Appendix A.

Algorithm Appendix A.1 describes the limiting strategy, which operates in *two successive passes*. In the first pass ($\theta = 1$), if condition (A.3) is satisfied, then all the coefficients of the approximation of order larger than 1 are put to zero, as expressed by (A.5). In the second pass ($\theta = 2$), condition (A.3) is again verified and, if it is true, then the coefficient of order 1 is scaled with a factor $c_k < 1$, as expressed by (A.6). In condition (A.3), $\delta\mathbf{v}_{k,\pm}$ are defined as

$$\delta\mathbf{v}_{k,\pm} := \lim_{\eta\to 0}\left[\mathbf{u}_h^{(k)}(x_k \pm \eta, \cdot)\right] - \overline{\mathbf{v}}_k, \tag{A.1}$$

where $\overline{\mathbf{v}}_k$ is the cell average of $\mathbf{u}_h^{(k)}$ at time t^n:

$$\overline{\mathbf{v}}_k := \frac{1}{\mathrm{vol}\left(V^{(k)}\right)} \int_{V^{(k)}} \mathbf{u}_h(x, t^n)\, dx. \tag{A.2}$$

See [57] for the choice of constant $M > 0$. In condition (A.3) the order of magnitude of each conserved variable is taken into account separately. In equation (A.4), Δ_+ is the forward difference operator $\Delta_+\overline{\mathbf{v}}_k := \overline{\mathbf{v}}_{k+1} - \overline{\mathbf{v}}_k$, and m is the minmod function [58] with 5 arguments. Note that the limiting process is conservative, since the coefficients $\mathbf{u}_1^{(k)}$ of order 1 are not affected by the limiting.

Algorithm Appendix A.1. *Limiter*

1: ***for*** $\theta = 1, 2$ ***do***
2: ***if*** *there exists* $j \in \{1, \ldots, r\}$ *such that:*

$$\frac{\max\left(\left|(\delta\mathbf{v}_{k+1,-})_j\right|, \left|(\delta\mathbf{v}_{k,+})_j\right|\right)}{\max\left(\max_{k\in\mathcal{I}}\left|(\overline{\mathbf{v}}_k)_j\right|, 1\right)} > M\Delta x^2 \tag{A.3}$$

then
3: *compute* $c_k := \min\left(1, c_{k,1}, \ldots, c_{k,r}\right)$ *with*

$$c_{k,j} := m\left(1, \frac{(\Delta_+\overline{\mathbf{v}}_{k-1})_j}{(\delta\mathbf{v}_{k,+})_j}, \frac{(\Delta_+\overline{\mathbf{v}}_k)_j}{(\delta\mathbf{v}_{k,+})_j}, \frac{(\Delta_+\overline{\mathbf{v}}_{k-1})_j}{(\delta\mathbf{v}_{k,-})_j}, \frac{(\Delta_+\overline{\mathbf{v}}_k)_j}{(\delta\mathbf{v}_{k,-})_j}\right) \tag{A.4}$$

4: ***if*** $c_k < 1$ ***then***
5: ***if*** $\theta = 1$ ***then***

6: *update the coefficients of the approximation as*

$$\mathbf{u}_i^{(k)} := \begin{cases} \mathbf{u}_i^{(k)}, & \text{if } i \in \{1,2\}, \\ \mathbf{0}, & \text{if } i > 2 \end{cases} \tag{A.5}$$

7: ***else***
8: *update the coefficients of the approximation as*

$$\mathbf{u}_i^{(k)} := \begin{cases} \mathbf{u}_1^{(k)}, & \text{if } i = 1, \\ c_k\, \mathbf{u}_i^{(k)}, & \text{if } i > 1 \end{cases} \tag{A.6}$$

9: ***end if***
10: ***end if***
11: ***end if***
12: ***end for***

Acknowledgements

Financial support from the Deutsche Forschungsgemeinschaft (German Research Association) through grant GSC 111, and by the Air Force Office of Scientific Research, Air Force Materiel Command, USAF, under grant number FA8655-08-1-3060, is gratefully acknowledged.

References

[1] M. J. Berger, J. Oliger, Adaptive Mesh Refinement for Hyperbolic Partial Differential Equations, J. Comput. Phys. 53 (1984) 484–512.
[2] M. J. Berger, P. Colella, Local Adaptive Mesh Refinement for Shock Hydrodynamics, J. Comput. Phys. 82 (1989) 64–84.
[3] J.-C. Jouhaud, M. Montagnac, L. Tourrette, A Multigrid Adaptive Mesh Refinement Strategy for 3D Aerodynamic Design, Int. J. Numer. Meth. Fluids 47 (2005) 367–385.
[4] R. H. Nochetto, K. G. Siebert, A. Veeser, Theory of Adaptive Finite Element Methods: an Introduction, in: Multiscale, Nonlinear and Adaptive Approximation, Springer, 2009, pp. 409–539.
[5] Z. J. Wang (Ed.), Adaptive High-Order Methods in Computational Fluid Dynamics, World Scientific Publishing Company, 2011.
[6] F. Bramkamp, P. Lamby, S. Müller, An Adaptive Multiscale Finite Volume Solver for Unsteady and Steady State Flow Computations, J. Comput. Phys. 197 (2004) 460–490.
[7] A. Harten, J. M. Hyman, Self-Adjusting Grid Methods for One-Dimensional Hyperbolic Conservation Laws, J. Comput. Phys. 50 (1983) 235–269.
[8] S. G. Mallat, A Theory for Multiresolution Signal Decomposition: The Wavelet Representation, IEEE Trans. Pattern Anal. Machine Intell. 11 (1989) 674–693.
[9] I. Daubechies, Ten Lectures on Wavelets, SIAM, Philadelphia, Pennsylvania, 1992.
[10] A. Harten, Discrete Multi-Resolution Analysis and Generalized Wavelets, Applied Numer. Math. 12 (1993) 153–192.
[11] A. Harten, Multiresolution Representation of Data: A General Framework, SIAM J. Numer. Anal. 33 (1996).
[12] A. Harten, Multiresolution Algorithms for the Numerical Solution of Hyperbolic Conservation Laws, Comm. Pure Appl. Math. 48 (1995) 1305–1342.
[13] E. Bacry, S. Mallat, G. Papanicolaou, A Wavelet Based Space-Time Adaptive Numerical Method for Partial Differential Equations, Math. Model. Numer. Anal. 26 (1992) 793–834.
[14] A. Cohen, I. Daubechies, J.-C. Feauveau, Biorthogonal Bases of Compactly Supported Wavelets, Comm. Pure Appl. Math. 45 (1992) 485–560.
[15] A. Harten, Adaptive Multiresolution Schemes for Shock Computations, J. Comput. Phys. 115 (1994) 319–338.
[16] A. Cohen, Numerical Analysis of Wavelet Methods, Elsevier, 2003.
[17] B. Gottschlich-Müller, S. Müller, Adaptive Finite Volume Schemes for Conservation Laws based on Local Multiresolution Techniques, in: M. Fey, R. Jeltsch (Eds.), Hyperbolic Problems: Theory, Numerics, applications, Birkhäuser, 1999, pp. 385–394.
[18] A. Cohen, S. M. Kaber, S. Müller, M. Postel, Fully Adaptive Multiresolution Finite Volume Schemes for Conservation Laws, Math. Comput. 72 (2003) 183–225.
[19] S. Müller, Adaptive Multiscale Schemes for Conservation Laws, Springer, 2003.
[20] S. Müller, Multiresolution schemes for conservation laws, in: R. DeVore, A. Kunoth (Eds.), Multiscale, Nonlinear and Adaptive Approximation, Springer, 2009, pp. 379–408.
[21] W. Dahmen, Wavelet Methods for PDEs - Some Recent Developments, J. Comput. Appl. Math. 128 (2001) 133–185.

[22] N. Hovhannisyan, S. Müller, R. Schäfer, Adaptive Multiresolution Discontinuous Galerkin Schemes for Conservation Laws, Technical Report 311, Institut für Geometrie und Praktische Mathematik, RWTH Aachen University, 2010.
[23] V. Strela, Multiwavelets: Theory and Applications, Ph.D. thesis, Massachussettes Institute of Technology, 1996.
[24] R. Schäfer, Adaptive Multiresolution Discontinuous Galerkin Schemes for Conservation Laws, Ph.D. thesis, RWTH Aachen University, 2011.
[25] F. Iacono, G. May, S. Müller, R. Schäfer, An Adaptive Multiwavelet-Based DG Discretization for Compressible Fluid Flow, in: A. Kuzmin (Ed.), ICCFD 2010, Proceedings of 6th Int. Conference on Computational Fluid Dynamics, Springer, 2011, pp. 813–822.
[26] F. Iacono, G. May, S. Müller, R. Schäfer, A Discontinuous Galerkin Discretization with Multiwavelet-Based Grid Adaptation for Compressible Flows, in: 49th AIAA Aerospace Sciences Meeting, 2011-0200.
[27] F. Iacono, High-Order Methods for Convection-Dominated Nonlinear Problems Using Multilevel Techniques, Ph.D. thesis, RWTH Aachen University, 2011 (in preparation).
[28] J. L. D. Calle, P. R. B. Devloo, S. M. Gomes, Wavelets and Adaptive Grids for the Discontinuous Galerkin Method, Numerical Algorithms 39 (2005) 143–154.
[29] A. B. Shelton, A Multi-Resolution Discontinuous Galerkin Method for Unsteady Compressible Flows, Ph.D. thesis, Georgia Institute of Technology, 2008.
[30] H. M. Kang, K. H. Kim, D. H. Lee, D. H. Lee, Improvement in Computational Efficiency of Euler Equations via a Modified Sparse Point Representation Method, Computers & Fluids 37 (2008) 65–280.
[31] H. M. Kang, D. H. Lee, K. H. Kim, D. H. Lee, Improved Computational Efficiency of Unsteady Flow Problems via the Modified Wavelet Method, AIAA Journal 46 (2008) 1191–1203.
[32] D. Lee, H. Kang, D. Lee, Convergence Acceleration for Euler Equation based on SPR, in: H. Deconinck, E. Dick (Eds.), Computational Fluid Dynamics 2006, Springer Berlin Heidelberg, 2009, pp. 297–302.
[33] R. Archibald, G. Fann, W. Shelton, Adaptive Discontinuous Galerkin Methods in Multiwavelets Bases, Applied Numer. Math. 61 (2011) 879–890.
[34] G. E. Karniadakis, S. J. Sherwin, Spectral/hp Element Methods for Computational Fluid Dynamics, Oxford University press, 2005.
[35] S. J. Ruuth, Global Optimization of Explicit Strong-Stability-Preserving Runge-Kutta Methods, Math. Comput. 75 (2006) 183–207.
[36] C.-W. Shu, TVB Uniformly High-Order Schemes for Conservation Laws, Math. Comput. 49 (1987) 105–121.
[37] B. V. Leer, Towards the Ultimate Conservative Difference Scheme. II. Monotonicity and Conservation Combined in a Second-Order Scheme, J. Comput. Phys. 14 (1974) 361–370.
[38] B. Alpert, G. Beylkin, D. Gines, L. Vozovoi, Adaptive Solution of Partial Differential Equations in Multiwavelet Bases, J. Comput. Phys. 182 (2002) 149–190.
[39] B. Alpert, A Class of Bases in L^2 for the Sparse Representation of Integral Operators, SIAM J. Math. Anal. 24 (1993) 246–262.
[40] B. Cockburn, C.-W. Shu, TVB Runge-Kutta Local Projection Discontinuous Galerkin Finite Element Method for Conservation Laws II: General Framework, Math. Comput. 52 (1989) 411–435.
[41] D. Levy, G. Puppo, G. Russo, A Fourth Order Central WENO Scheme for Multi-Dimensional Hyperbolic Systems of Conservation Laws, SIAM J. Sci. Comput. 24 (2002) 480–506.
[42] J. Chen, Z. Shi, Application of a Fourth-Order Relaxation Scheme to Hyperbolic Systems of Conservation Laws, Acta Mech. Sinica 22 (2006) 84–92.
[43] B. Cockburn, S.-Y. Lin, C.-W. Shu, TVB Runge-Kutta Local Projection Discontinuous Galerkin Finite Element Method for Conservation Laws III: One-Dimensional Systems, J. Comput. Phys. 84 (1989) 90–113.
[44] J. D. Anderson, Fundamentals of Aerodynamics, McGraw-Hill, 2007.
[45] G. A. Sod, A Survey of Several Finite Difference Methods for Systems of Nonlinear Hyperbolic Conservation Laws, J. Comput. Phys. 27 (1978) 1–31.
[46] P. Woodward, Trade-Offs in Designing Explicit Hydrodynamical Schemes for Vector Computers, in: G. Rodrigue (Ed.), Parallel Computations, Academic Press, 1982, pp. 153–172.
[47] P. Woodward, P. Colella, The Numerical Simulation of Two-Dimensional Fluid Flow with Strong Shocks, J. Comput. Phys. 54 (1984) 115–173.
[48] E. F. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics: a Practical Introduction, Springer, 1999.
[49] E. F. Toro, M. Spruce, W. Speares, Restoration of the Contact Surface in the HLL-Riemann Solver, Shock Waves 4 (1994) 25–34.
[50] A. Harten, P. D. Lax, B. Van Leer, On Upstream Differencing of Godunov-Type Schemes for Hyperbolic Conservation Laws, SIAM Rev. 25 (1983) 35–61.
[51] M. Arora, P. L. Roe, On Postshock Oscillations Due to Shock Capturing Schemes in Unsteady Flows, J. Comput. Phys. 130 (1997) 25–40.
[52] S. Karni, S. Canic, Computations of Slowly Moving Shocks, J. Comput. Phys. 136 (1997) 132–139.
[53] E. Johnsen, S. K. Lele, J. Larsson, Analysis and Correction of Errors Generated by Slowly Moving Shocks, in: 49th AIAA Aerospace Sciences Meeting, 2011-657.
[54] C.-W. Shu, S. Osher, Efficient Implementation of Essentially Non-Oscillatory Shock-Capturing Schemes II, J. Comput. Phys. 83 (1989) 32–78.
[55] T. P. Y. Yu, K. Kolarov, W. Lynch, Barysymmetric Multiwavelets on Triangle, Technical Report 1997-006, Interval Research Corporation, 1997.
[56] M. Holmström, Solving Hyperbolic PDEs Using Interpolating Wavelets, SIAM J. Sci. Comput. 21 (1999) 405–420.
[57] C.-W. Shu, TVB Boundary Treatment for Numerical Solutions of Conservation Laws, Math. Comput. 49 (1987) 123–134.
[58] A. Harten, On a Class of High Resolution Total-Variation-Stable Finite-Difference Schemes, SIAM J. Numer. Anal. 21 (1984) 1–23.