

ADER Schemes on Adaptive Triangular Meshes for Scalar Conservation Laws

MARTIN KÄSER and ARMIN ISKE

Abstract. ADER schemes are recent finite volume methods for hyperbolic conservation laws, which can be viewed as generalizations of the classical first order Godunov method to arbitrary high orders. In the ADER approach, high order polynomial reconstruction from cell averages is combined with high order flux evaluation, where the latter is done by solving generalized Riemann problems across cell interfaces. Currently available nonlinear ADER schemes are restricted to Cartesian meshes. This paper proposes an *adaptive* nonlinear finite volume ADER method on unstructured triangular meshes for scalar conservation laws, which works with WENO reconstruction. To this end, a customized stencil selection scheme is developed, and the flux evaluation of previous ADER schemes is extended to triangular meshes. Moreover, an a posteriori error indicator is used to design the required adaption rules for the dynamic modification of the triangular mesh during the simulation. The expected convergence orders of the proposed ADER method are confirmed by numerical experiments for linear and nonlinear scalar conservation laws. Finally, the good performance of the adaptive ADER method, in particular its robustness and its enhanced flexibility, is further supported by numerical results concerning Burgers equation.

1 Introduction

Modern approaches for the construction of conservative, high order numerical methods for hyperbolic conservation laws are based on finite volume discretizations (FV), combined with *essentially non-oscillatory* (ENO) or *weighted essentially non-oscillatory* (WENO) reconstruction schemes.

The basic idea of ENO schemes is to first select, for each control volume, a set of stencils comprising neighbouring control volumes. Then, for each stencil a recovery polynomial is computed, which interpolates given cell averages over the control volumes in the stencil. Among the different recovery polynomials, the *smoothest* (i.e. least oscillatory) polynomial is finally selected, which constitutes the numerical solution of the hyperbolic conservation law over its corresponding control volume. In this way, ENO schemes lead to finite volume discretizations of high order space accuracy, provided

that high order reconstruction polynomials are utilized. Moreover, by the selection of smoothest polynomials, spurious oscillations can be avoided.

In the more sophisticated WENO approach, the whole stencil set is used in order to construct, for a corresponding control volume, a *weighted* sum of reconstruction polynomials, each belonging to one stencil. Moreover, the weights are determined by a specific oscillation indicator, which measures the oscillation behaviour of each reconstruction polynomial. WENO schemes show, in comparison with ENO schemes, superior convergence to steady-state solutions and higher order accuracy, especially in smooth regions and around extrema of the solution.

ENO schemes date back to Harten, Engquist, Osher, and Chakravarthy [12], who introduced the concept of ENO schemes for one-dimensional conservation laws. Later, Harten and Chakravarthy [11], Abgrall [1], and Sonar [27] extended their finite volume formulation to unstructured triangular meshes. First WENO schemes were proposed by Liu, Osher, and Chan [19], and by Jiang and Shu [16]. Somewhat later, Friedrich [7], Hu and Shu [13], constructed WENO schemes on unstructured meshes.

In finite volume discretizations, high order accuracy in time is usually obtained by using multi-stage Runge-Kutta methods. In order to avoid oscillatory solutions, the time discretization is required to be *total variation diminishing* (TVD), as observed by Shu [25], Shu and Osher [26]. However, Ruuth and Spiteri [21] showed that the (time) accuracy order of any TVD Runge-Kutta method is essentially limited, which in turn limits the accuracy order of the overall finite volume scheme.

Toro, Millington, and Nejad [31] proposed in 2001 an explicit one-step finite volume scheme, termed **ADER**, which is of **A**rbitrary high order, using high order **DER**ivatives of polynomials. The finite volume discretization of [31] combines high order polynomial reconstruction from cell averages with high order flux evaluation. The latter is done by solving generalized Riemann problems across the cell interfaces, i.e., boundaries of adjacent control volumes. Therefore, the finite volume ADER scheme of the seminal work [31] can be viewed as a generalization of the classical first order Godunov scheme to arbitrary high orders.

ADER schemes have very recently gained considerable popularity in applications from gas and aerodynamics, see e.g. [22, 23], especially for linear advection and linear acoustic problems [6, 24]. Moreover, the application of ADER schemes to nonlinear problems and systems of hyperbolic equations is subject of lively research. But currently available nonlinear ADER schemes are restricted to the one-dimensional case [28, 32], or (for the multi-dimensional case) to Cartesian meshes [22, 23, 29].

This paper proposes a new *adaptive* nonlinear ADER scheme on unstructured triangular meshes for solving Cauchy problems for scalar conservation laws of the form

$$\frac{\partial u}{\partial t} + \nabla f(u) = 0, \quad (1)$$

where for some bounded open domain $\Omega \subset \mathbb{R}^2$, and time interval $I = [0, T]$, $T > 0$, the function $u : I \times \Omega \rightarrow \mathbb{R}$ is the unknown solution of (1), and where $f(u) = (f_1(u), f_2(u))^T$ denotes the flux tensor.

Note that for a *nonlinear* flux, the solution of the hyperbolic equation (1) typically develops discontinuities in the solution u , denoted as shocks. In order to model the propagation of moving discontinuities, it is of primary importance to work with a higher resolution around the discontinuities. This essentially requires adaptive methods in order to effectively combine high order resolution with small computational costs.

The adaptive ADER scheme, proposed in this paper, works with an unstructured triangular mesh, which is modified during the simulation. The required adaption rules are based on a customized a posteriori error indicator, whose construction is based on the ideas in our previous papers [3, 4, 15]. The adaptive ADER scheme of this paper provides an explicit one-step finite volume discretization, whose enhanced flexibility is due to the effective and customized adaption of the triangular mesh. Therefore, the ADER scheme of this paper can be viewed as an extension of previous ADER schemes to adaptive triangular meshes.

The outline of this paper is as follows. In the following Section 2, the basic concepts of high order WENO reconstruction of polynomials from cell averages over triangles is explained. This includes a discussion on an advanced selection strategy for one-sided stencils by using backward sectors. Section 3 is then devoted to high order flux evaluation, where the concept of previous ADER schemes [31, 32, 33] is extended to triangular meshes. In Section 4, the expected convergence orders of the proposed ADER scheme are confirmed by numerical experiments concerning linear and nonlinear scalar conservation laws. The good performance of the adaptive ADER scheme, in particular its robustness and enhanced flexibility, is further supported by using a nonlinear model problem concerning Burgers equation. In order to keep this paper widely self-contained, the required adaption rules, similar to the ones of our previous papers [3, 4, 15], are developed separately in the Appendix.

2 High Order WENO Reconstruction

The reconstruction of high order multivariate polynomials from scattered data is a numerically very critical task. Indeed, already the reconstruction of bivariate polynomials from scattered data requires solving interpolation problems, which are typically ill-conditioned, especially when the reconstruction order is high, or when the scattered data are very unevenly distributed.

This section concerns the reconstruction of high order bivariate polynomials from scattered cell averages on unstructured triangular meshes, as required in the WENO reconstruction of our ADER scheme. To this end, we first formulate the reconstruction problem in the following Subsection 2.1, where critical aspects concerning numerical stability are discussed. Further details on WENO reconstruction are then explained in Subsection 2.2. This is followed by a discussion on the selection of admissible stencils in Subsection 2.3, which is a crucial task for the performance of WENO reconstruction. To this end, an improved scheme for the construction of one-sided stencils by using backward sectors is suggested.

2.1 Reconstruction from Cell Average Values

In order to explain polynomial reconstruction from (scattered) cell averages, let us first fix some required notation. In what follows, we let $\alpha = (\alpha_1, \alpha_2) \in \mathbb{N}_0^2$ denote an index pair, and we use the standard notation $|\alpha| = \alpha_1 + \alpha_2$, $x^\alpha = x_1^{\alpha_1} x_2^{\alpha_2}$ for $x = (x_1, x_2) \in \mathbb{R}^2$, and $D^\alpha = \frac{\partial^{|\alpha|}}{\partial x_1^{\alpha_1} \partial x_2^{\alpha_2}}$.

For any $x_0 \in \mathbb{R}^2$, the set $\{(\cdot - x_0)^\alpha : |\alpha| \leq n\}$ of polynomials is a basis of \mathcal{P}_n , denoting the bivariate polynomials of degree at most n . Therefore, any $p \in \mathcal{P}_n$ can uniquely be expressed by a *monomial expansion* of the form

$$p(x) = \sum_{|\alpha| \leq n} a_\alpha (x - x_0)^\alpha, \quad (2)$$

around x_0 , with coefficients $a_\alpha \in \mathbb{R}$, $|\alpha| \leq n$. We remark at this point, that the representation for p in (2) is usually not suitable for numerical computations (but often quite useful for theoretical purposes). We come back to this important point later in this subsection.

Next, we assume that the computational domain $\Omega \subset \mathbb{R}^2$ in (1) is partitioned by a *conforming triangulation*. Recall that a conforming triangulation $\mathcal{T} = \{T\}_{T \in \mathcal{T}}$ of Ω is a triangular mesh, consisting of pairwise distinct closed nondegenerate triangles, $T \subset \overline{\Omega}$ for $T \in \mathcal{T}$, such that the following two properties are satisfied (see e.g. [20, Section 3.3.1], where the term *primary grid* is used).

- the union of the triangles in \mathcal{T} coincides with the closure $\overline{\Omega}$ of the domain Ω , i.e., $\overline{\Omega} = \bigcup_{T \in \mathcal{T}} T$.
- two different triangles in \mathcal{T} are either disjoint, or they share a common vertex or they share a common edge.

In finite volume methods, each triangle $T \in \mathcal{T}$, also termed (triangular) *cell* or *control volume*, carries, at any fixed time $t \in I$, a *cell average value*

$$\mathcal{A}_T(u) = \frac{1}{|T|} \int_T u(x) dx, \quad \text{for } T \in \mathcal{T}, \quad (3)$$

where $|T|$ is the area of triangle T and $u \equiv u(t, \cdot)$ is the solution of (1) at time t . Note that the cell average $\mathcal{A}_T(u)$ also depends on time t , but for notational simplicity, we omit this here.

Now let us turn to the reconstruction of polynomials in \mathcal{P}_n from N given cell average values $\{\mathcal{A}_{T_{k_\ell}}(u)\}_{1 \leq \ell \leq N}$, with $T_{k_\ell} \in \mathcal{T}$, $1 \leq \ell \leq N$, where $N = (n+1) \times (n+2)/2$ is the dimension of \mathcal{P}_n . This problem requires finding a polynomial $p \in \mathcal{P}_n$, which satisfies the interpolation conditions

$$\mathcal{A}_{T_{k_\ell}}(p) = \mathcal{A}_{T_{k_\ell}}(u), \quad \text{for } 1 \leq \ell \leq N. \quad (4)$$

When using the representation (2), e.g. for $x_0 = 0$, this reconstruction problem leads to a linear equation system, with square coefficient matrix,

$$V = \left(\mathcal{A}_{T_{k_\ell}}(x^\alpha) \right)_{1 \leq \ell \leq N; |\alpha| \leq n} \in \mathbb{R}^{N \times N}, \quad (5)$$

usually referred to as *Vandermonde matrix*. Hence, the reconstruction problem (4) has a unique solution, iff the Vandermonde matrix V in (5) is non-singular, in which case the set $\mathcal{S} = \{T_{k_\ell}\}_{1 \leq \ell \leq N} \subset \mathcal{T}$ of triangles is said to form an *admissible* stencil for \mathcal{P}_n , i.e., the stencil \mathcal{S} is unisolvent w.r.t. the polynomial space \mathcal{P}_n .

Abgrall shows in [1], that the condition number of the Vandermonde matrix V in (5) is $\mathcal{O}(h^{-n})$, where h is a measure for the local mesh width of the triangles in \mathcal{S} , see [1] for details. So for large degree n and small mesh width h the corresponding linear equation system is ill-conditioned. But the condition number of the linear system depends on the choice of the basis for the polynomial expansion. Therefore, for the sake of numerical stability, Abgrall suggests in [1] to replace the representation in (2) by a polynomial expansion, of the form (7), based on *barycentric coordinates*.

In order to briefly explain this standard stabilization technique, let the set $\mathcal{S}_n = \{T_1, T_2, \dots, T_N\} \subset \mathcal{T}$, $N \geq 3$, denote an admissible stencil for \mathcal{P}_n ,

$n \geq 1$. Then, there is a substencil $\mathcal{S}_1 \subset \mathcal{S}_n$ containing three triangles from \mathcal{S}_n , say $\mathcal{S}_1 = \{T_1, T_2, T_3\}$, such that \mathcal{S}_1 constitutes an admissible stencil for \mathcal{P}_1 . In this case, there are unique linear polynomials $\Lambda_1, \Lambda_2, \Lambda_3 \in \mathcal{P}_1$ satisfying

$$\mathcal{A}_{T_j}(\Lambda_i) = \delta_{ij}, \quad 1 \leq i, j \leq 3, \quad \text{with} \quad \sum_{i=1}^3 \Lambda_i(x) \equiv 1. \quad (6)$$

The polynomials $\Lambda_1, \Lambda_2, \Lambda_3$ in (6) are said to be the *barycentric coordinates* of the stencil \mathcal{S}_1 . Now any polynomial $p \in \mathcal{P}_n$ can uniquely be expressed as a linear combination of the form

$$p(x) = \sum_{|\alpha| \leq n} b_\alpha \Lambda^\alpha(x), \quad \text{where } \Lambda^\alpha = \Lambda_1^{\alpha_1} \Lambda_2^{\alpha_2}. \quad (7)$$

Due to the scale-invariance of the barycentric coordinates $\Lambda_1, \Lambda_2, \Lambda_3$, the condition number of the matrix

$$B = \left(\mathcal{A}_{T_{k_\ell}}(\Lambda^\alpha) \right)_{1 \leq \ell \leq N; |\alpha| \leq n} \in \mathbb{R}^{N \times N},$$

is independent of the local mesh width h , see [1]. Therefore, the representation (7) is, due to its robustness, particularly suited for adaptive mesh refinement, even for strongly distorted meshes.

2.2 WENO Reconstruction

During the last decade, WENO reconstruction methods have extensively been used for one-dimensional problems, and they have also gained popularity for problems on multi-dimensional Cartesian meshes, where the latter basically boils down to solving several one-dimensional problems separately. The basic idea of truly two-dimensional WENO reconstruction on triangulations is to first select, for each triangular cell $T \in \mathcal{T}$, k admissible stencils \mathcal{S}_i , $i = 1, \dots, k$, before a set of reconstruction polynomials $p_i \in \mathcal{P}_n$, each corresponding to one stencil \mathcal{S}_i , is computed.

For the reconstruction polynomial $p \in \mathcal{P}_n$ on triangle T , the WENO method uses a weighted sum

$$p(x) = \sum_{i=1}^k \omega_i p_i(x), \quad \text{with} \quad \sum_{i=1}^k \omega_i = 1, \quad (8)$$

of the reconstruction polynomials p_i , where the normalized weights ω_i are positive and data-dependent. The weights ω_i in (8) are determined by using

an *oscillation indicator*, which measures, for any stencil $\mathcal{S}_i \subset \mathcal{S}_n$, the oscillation behaviour of the corresponding reconstruction polynomial $p_i \in \mathcal{P}_n$ on triangle T , $1 \leq i \leq k$. As supported by numerical results in [7, 13], the oscillation indicator

$$\mathcal{I}_T(p) = \sum_{1 \leq |\alpha| \leq n} \int_T |T|^{|\alpha|-1} |D^\alpha p(x)|^2 dx, \quad \text{for } p \in \mathcal{P}_n \text{ and } T \in \mathcal{T}, \quad (9)$$

is very suitable. Furthermore, the weights ω_i in (8) are then given by

$$\omega_i = \frac{\tilde{\omega}_i}{\sum_{i=1}^k \tilde{\omega}_i} \quad \text{with} \quad \tilde{\omega}_i = (\epsilon + \mathcal{I}_T(p_i))^{-r}, \quad \text{for } i = 1, \dots, k. \quad (10)$$

The parameter ϵ in (10) is a small positive number to avoid division by zero. We remark that numerical results are usually not sensitive to the choice of ϵ . In general, large values ϵ are suitable for smooth problems. However, a large value ϵ may lead to small (undesired) oscillations near shocks. Therefore, smaller values ϵ are preferably used for discontinuous problems. In our numerical examples, we let $\epsilon = 10^{-5}$.

The positive integer r in (10) serves to control the sensitivity of the weights with respect to the oscillation indicator (9). Note that in the limit, when r tends to infinity, the resulting WENO scheme becomes a classical ENO scheme, where only one stencil, corresponding to one *smoothest* (i.e. least oscillatory) reconstruction polynomial, is taken. In contrast, when r tends to zero, this leads to a WENO scheme with equal weights $\omega_i \equiv 1/k$, $1 \leq i \leq k$, in which case this “WENO” reconstruction may become oscillatory or even unstable. In our implementation we let $r = 4$, which turns out to be large enough to (essentially) avoid undesired oscillations near discontinuities, but small enough to improve upon the classical ENO scheme.

2.3 Stencil Selection

This subsection proposes a customized stencil selection technique for WENO reconstruction by high order polynomials from scattered cell averages. This in particular leads to an improvement over previous stencil selection strategies, especially in the construction of one-sided stencils near discontinuities.

Let us first remark that the selection of admissible stencils from unstructured triangular meshes is a critical task, especially for large polynomial degree n . In fact, the *quality* of the utilized stencils, to be selected among many admissible stencils, has a strong impact on the performance of the resulting WENO reconstruction. The following aspects are crucial for the selection of k suitable stencils \mathcal{S}_i , $i = 1, \dots, k$, around a “center” cell $T \in \mathcal{T}$.

- every stencil should be local (relative to its corresponding center T);
- the number of stencils, k , should be small in order to keep the required computational costs small;
- in *smooth* regions of the solution the stencils should, for the sake of good approximation quality, be *well-centered* (i.e. isotropic);
- in *non-smooth* (i.e. discontinuous) regions of the solution, *one-sided* (i.e. anisotropic) stencils should be preferred in order to avoid interpolation across discontinuities, which would lead to undesired oscillations.

In order to construct suitable (local) stencils on unstructured triangulations, we work with various concepts of triangle neighbourhoods, as some of these were already utilized in [11, 27]. Let us first recall some relevant ideas from [11, 27], before we propose an extension for the construction of one-sided stencils of [11] later in this subsection.

Definition 1 *Let \mathcal{T} be a conforming triangulation. For any triangle $T \in \mathcal{T}$ the set*

$$\mathcal{N}^0(T) = \left\{ \tilde{T} \in \mathcal{T} \setminus \{T\} : \tilde{T} \cap T \text{ is an edge of } T \right\}$$

is called level-0 von Neumann neighbourhood of triangle T . Any triangle in $\mathcal{N}^0(T)$ is called a level-0 von Neumann neighbour of T .

A straightforward extension to level-1 von Neumann neighbourhoods (and level-1 von Neumann neighbours) can be accomplished by merging level-0 von Neumann neighbourhoods, so that the level-1 von Neumann neighbourhood of any triangle $T \in \mathcal{T}$ is given by

$$\mathcal{N}^1(T) = \left(\bigcup_{\tilde{T} \in \mathcal{N}^0(T)} \mathcal{N}^0(\tilde{T}) \right) \setminus \{T\}.$$

Figure 1 shows an example for level- i von Neumann neighbourhoods, $i = 0, 1$, of a triangle (dark-shaded), along with its (light-shaded) level-0 von Neumann neighbours (Figure 1 (a)), and its (light-shaded) level-1 von Neumann neighbours (Figure 1 (b)).

We further extend von Neumann neighbourhoods to higher level- p von Neumann neighbourhoods by the recursive definition

$$\mathcal{N}^p(T) = \left(\bigcup_{\tilde{T} \in \mathcal{N}^{p-1}(T)} \mathcal{N}^{p-1}(\tilde{T}) \right) \setminus \{T\}, \quad \text{for } p \geq 1,$$

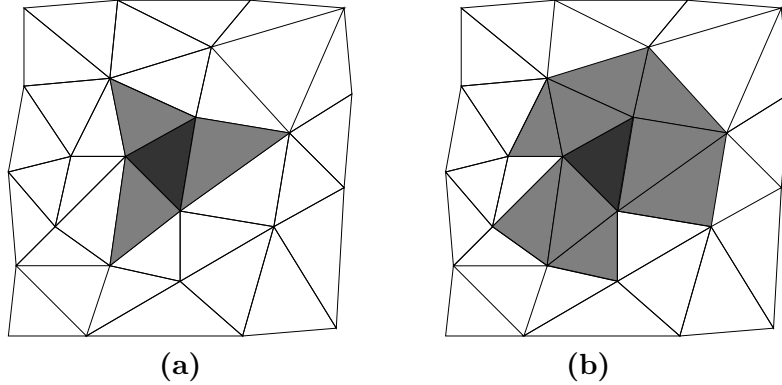


Figure 1: A triangle (dark-shaded) and its **(a)** (light-shaded) level-0 von Neumann neighbours; **(b)** (light-shaded) level-1 von Neumann neighbours.

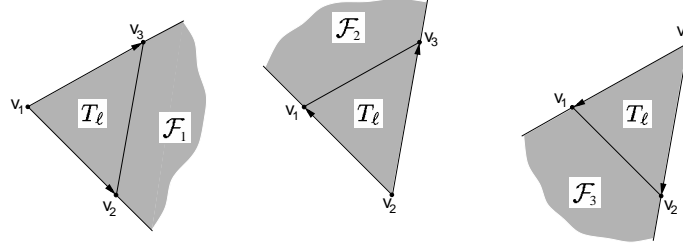
in order to obtain a richer set of admissible *well-centered* (i.e. isotropic) stencils, which are used in the WENO reconstruction of (higher order) polynomials in smooth regions of the solution.

As to the stencil selection in non-smooth regions of the solution, so-called *one-sided* stencils are preferred. One-sided stencils are required to capture preference directions of the solution, and so the construction of such anisotropic stencils requires particular care. According to Harten and Chakravarthy [11], the construction of suitable one-sided stencils can be accomplished by employing a sectoral search algorithm.

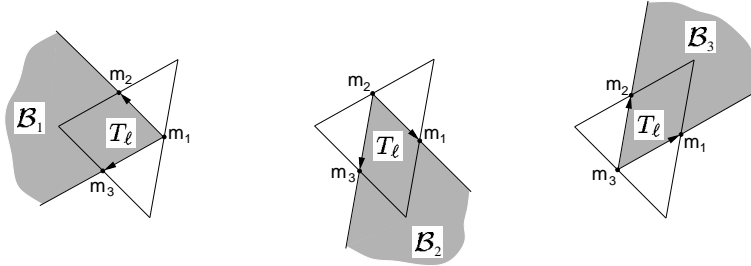
The basic idea in [11] for this sectoral search is to merely include von Neumann neighbours of a triangle $T \in \mathcal{T}$, whose barycenters lie in one of the three *forward sectors* \mathcal{F}_j , $j = 1, 2, 3$, of T . Recall that each forward sector of T is spanned by a corresponding edge pair of T , such that the resulting sector contains T . For the purpose of illustration, Figure 2 **(a)** shows the three forward sectors $\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3$ of a triangle $T = T_\ell$.

Here we further improve the construction of one-sided stencils by including additional sectors, called *backward sectors*. For any triangle T , its three backward sectors \mathcal{B}_j , $j = 1, 2, 3$, are defined by the three midpoints m_1, m_2, m_3 of the edges of T , where each backward sector has its origin at one midpoint and its two boundary edges pass through the other two midpoints. Figure 2 **(b)** shows the three backward sectors $\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3$ of a triangle $T = T_\ell$.

The basic idea for also including backward sectors is to enlarge the sample of directions, on which the subsequent construction of one-sided stencils



(a) the three forward sectors $\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3$ of T_ℓ



(b) the three backward sectors $\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3$ of T_ℓ

Figure 2: Forward sectors and backward sectors of a triangle T_ℓ .

relies. Note that for any triangle $T \in \mathcal{T}$, each of its three backward sectors, \mathcal{B}_j , corresponds to an opposite forward sector \mathcal{F}_j , $j = 1, 2, 3$. Due to the geometry of the complementary six sectors, \mathcal{B}_j and \mathcal{F}_j , $j = 1, 2, 3$, this allows us to better capture preference directions of the solution around triangle T , which in turn improves the quality of the WENO reconstruction at T . Indeed, this is supported by our numerical tests.

Let us finally remark that the shape of a stencil depends on the local geometry of the mesh. Especially for high order reconstruction, and for highly distorted meshes, this may lead to non-admissible stencils. In the implementation of our ADER method, such non-admissible stencils are detected and ignored. This in turn leads to a very robust WENO reconstruction, as only admissible stencils are considered.

3 High Order Flux Evaluation on Triangulations

In this section, we explain our extension of the ADER flux evaluation scheme in [28, 31, 32, 33] to unstructured triangular meshes. To this end, we first recall some relevant background on finite volume methods in Subsection 3.1, before details on the required ADER flux evaluation across cell interfaces are discussed in Subsection 3.2. The latter relies on the solution to generalized Riemann problems, explained in Subsection 3.3.

In combination with high order WENO reconstruction of Section 2, this yields an explicit one-step finite volume method on unstructured triangular meshes, of arbitrary high order m , referred to as ADER m . A corresponding CFL stability condition for ADER m schemes is developed in Subsection 3.4, before the algorithmic formulation of the method ADER m is finally provided in Subsection 3.5.

3.1 Finite Volume Formulation

In order to explain some relevant concepts of finite volume methods, let us consider the two-dimensional scalar conservation law (1) with solution $u(t, x)$. According to the finite volume method, discrete values of the solution u are taken as cell averages over a partitioning $\mathcal{T} = \{T\}_{T \in \mathcal{T}}$ of the domain Ω into finitely many control volumes. We remark that in the general formulation of finite volume schemes, the partitioning \mathcal{T} is not necessarily required to be a triangular mesh.

In the finite volume method of this paper we work with conforming Delaunay triangulations (see Section 2), in which case the partitioning \mathcal{T} of the domain Ω is a triangular mesh. In order to somewhat simplify our notation of the previous section, let $\bar{u}_T^n = \mathcal{A}_T(u)$ denote, for any triangle $T \in \mathcal{T}$, the cell average of u over T at time $t = t^n$, see (3). Moreover, let $\tau = t^{n+1} - t^n$ denote a current time step length, from time t^n to t^{n+1} .

The formulation of any finite volume scheme (see [18, Chapter 23]) usually results in an explicit numerical method of the form

$$\bar{u}_T^{n+1} = \bar{u}_T^n - \frac{\tau}{|T|} \sum_{j=1}^3 \hat{F}_{T,j}^n, \quad (11)$$

where $\hat{F}_{T,j}^n$ is the *numerical flux* across the edge $(\partial T)_j$, $j = 1, 2, 3$, of the triangular cell T during the time interval $[t^n, t^{n+1}]$. For a more comprehensive treatment of finite volume methods, we refer to the textbooks [18, 30].

3.2 Flux Evaluation Across Triangular Cells

With assuming polynomial representation for the numerical solution u over the triangular cells $T \in \mathcal{T}$, the numerical flux $\hat{F}_{T,j}^n$ in (11) can be computed *exactly* by using Gauss quadrature. In this case, the numerical flux is given by a weighted sum of the form

$$\hat{F}_{T,j}^n = \sum_{k=1}^{N_t} \alpha_k |(\partial T)_j| \sum_{h=1}^{N_x} \beta_h \vec{F}(u(t_{G_k}, x_{G_h})) \cdot \vec{n}_{T,j}, \quad (12)$$

whose weights α_k , β_h , and integration points (t_{G_k}, x_{G_h}) of its time and space discretization are determined by the utilized Gaussian quadrature rule. Moreover, $\vec{n}_{T,j}$ in (12) is the outer normal vector of the edge $(\partial T)_j$, whose length is denoted as $|(\partial T)_j|$, $j = 1, 2, 3$.

To evaluate the flux function \vec{F} in (12) at the Gaussian integration points (t_{G_k}, x_{G_h}) , we essentially need to determine the function values $u(t_{G_k}, x_{G_h})$, $1 \leq k \leq N_t$, $1 \leq h \leq N_x$, also referred to as the *states* of the solution at the cell interface. This is accomplished by solving a *generalized Riemann problem* (GRP) at the integration points (t_{G_k}, x_{G_h}) , respectively.

Let us first formulate this GRP, before we discuss further details concerning flux evaluation. In order to extend the previous ADER scheme [28, 29, 32, 33] to triangular meshes, we express the arising multi-dimensional GRP as a sequence of (simpler) *one-dimensional* GRPs normal to the cell interfaces, where each (one-dimensional) GRP corresponds to one Gaussian integration point. In order to further explain this, let $T \in \mathcal{T}$ denote a triangular cell, and let $x_{G_h} \in T$ denote a Gaussian integration point in (12), located at one cell interface of T . Then, the corresponding one-dimensional GRP across this cell interface at x_{G_h} has local (spatial) coordinate $x \equiv x_{\mathbf{n}}$, whose origin is x_{G_h} and whose orientation is along the corresponding outer normal $\vec{\mathbf{n}}$ of T , see Figure 3.

Any such one-dimensional GRP is described by the governing partial differential equation (PDE) and the initial condition (IC) for $u(t, x)$ at *local* time $t = 0$ (i.e., corresponding to current time $t \equiv t^n$) by

$$\text{PDE:} \quad \frac{\partial u}{\partial t} + \nabla f(u) = 0, \quad (13)$$

$$\text{IC:} \quad u(0, x) = \begin{cases} p_{\text{in}}(x), & \text{for } x < 0, \\ p_{\text{out}}(x), & \text{for } x > 0, \end{cases} \quad (14)$$

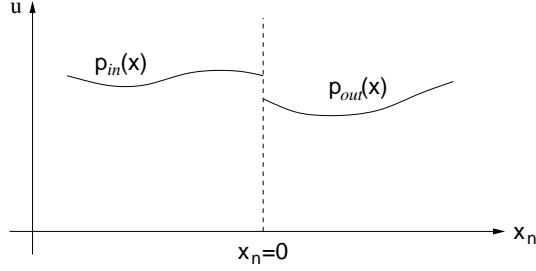


Figure 3: Generalized Riemann problem along the outer unit normal with reconstruction polynomials $p_{\text{in}}(x)$ and $p_{\text{out}}(x)$.

where the polynomial belonging to the triangular cell T is denoted as p_{in} , and the polynomial belonging to the adjacent triangle (at this cell interface) is denoted as p_{out} . The solution of the GRP (13),(14) is discussed in the following subsection.

3.3 Solving the Generalized Riemann Problem

Recalling equation (12), we wish to evaluate the solution $u(t, \cdot)$ of the one-dimensional GRP (13),(14) at any Gaussian integration point x_{G_h} for intermediate time $t_{G_k} \in [t^n, t^{n+1}]$. This leads us to one of the central ideas of the ADER approach: the solution u is approximated at m -th order time accuracy at the cell interface $x = 0$ by using its Taylor series expansion around (local) time $t = 0$, so that

$$u(t, 0) \approx u(0^+, 0) + \sum_{k=1}^{m-1} \frac{t^k}{k!} \frac{\partial^k}{\partial t^k} u(0^+, 0), \quad (15)$$

where we let $0^+ = \lim_{t \searrow 0} t$.

So on given accuracy order m , this requires solving a sequence of one-dimensional GRPs, one for each Gaussian integration point, across the cell interfaces at accuracy order m (for the time discretization). We refer to this generalized Riemann problem as GRP_{m-1} in order to indicate its dependence on m . For order $m = 1$, for instance, this leads us to the conventional Riemann Problem (RP), GRP_0 , where the initial condition is given by two constant functions, separated by the corresponding cell interface. Therefore, ADER schemes can be viewed as generalization of the classical first order Godunov scheme [9] to arbitrary high order.

Let us now address the evaluation of the terms on the right hand side of (15) in detail. Its leading term $u(0^+, 0)$ accounts for the first-instant interaction of the left and right data states at the cell interface, corresponding to a Gaussian quadrature point x_{G_h} . In order to determine the state $u(0^+, 0)$ at x_{G_h} , we follow along the lines of Toro and Titarev [32]. According to [32], the two reconstruction polynomials, p_{in} and p_{out} , which are belonging to the two adjacent cells of the interface at x_{G_h} , are first evaluated at x_{G_h} in order to obtain *boundary extrapolated values*, u_ℓ and u_r (ℓ = left; r = right). The leading term $u(0^+, 0)$ in (15) is then determined by the solution of a conventional Riemann problem, GRP_0 , of the form

$$\text{PDE:} \quad \frac{\partial u}{\partial t} + \nabla f(u) = 0, \quad (16)$$

$$\text{IC:} \quad u(0, x) = \begin{cases} u_\ell \equiv \lim_{x \rightarrow x_{G_h}^-} p_{\text{in}}(x), & \text{for } x < 0, \\ u_r \equiv \lim_{x \rightarrow x_{G_h}^+} p_{\text{out}}(x), & \text{for } x > 0, \end{cases} \quad (17)$$

where the solution is evaluated along the t -axis. For further details, we refer to [32].

Now let us turn to the evaluation of the remaining $m - 1$ terms in (15), which include the time derivatives $\frac{\partial^k}{\partial t^k} u(0^+, 0)$ of the solution at the corresponding Gaussian integration point x_{G_h} . In order to compute these required time derivatives, we employ the *Cauchy-Kowalewski* method, being a recursive procedure to express any time derivative in (15) as a function of available space derivatives. In fact, by applying the Cauchy-Kowalewski procedure, any time derivative of $u(t, x)$ can at any point (t, x) be expressed as a function of the form

$$\frac{\partial^k}{\partial t^k} u(t, x) = \mathbf{G}^k \left(\partial_x^{(0)} u(t, x), \dots, \partial_x^{(k)} u(t, x) \right), \quad 1 \leq k \leq m - 1, \quad (18)$$

where we let $u_x^{(j)} = \frac{\partial^j}{\partial x^j} u$, $0 \leq j \leq k$, for the space derivatives.

Now in order to evaluate the required space derivatives $\partial_x^{(j)} u(t, x)$ in (18) at the Gaussian integration point x_{G_h} , and at time $t = 0^+$, we work with *boundary extrapolated derivatives*,

$$\begin{aligned} \partial_x^{(k)} u_\ell &= \lim_{x \rightarrow x_{G_h}^-} \partial_x^{(k)} p_{\text{in}}(x), \\ \partial_x^{(k)} u_r &= \lim_{x \rightarrow x_{G_h}^+} \partial_x^{(k)} p_{\text{out}}(x), \end{aligned} \quad k = 1, \dots, m - 1,$$

given by the derivatives of the two polynomials, p_{in} and p_{out} , which are belonging to the two adjacent cells of the interface at x_{G_h} . These extrapolated derivatives can be viewed as constant states for further $m - 1$ conventional Riemann problems of space derivatives.

According to [32], an evolution equation can be constructed for each space derivative $\partial_x^{(j)} u$ in (18). This is done by differentiation of the governing equation (13) with respect to x . Indeed, as shown in [32], each space derivative then satisfies the inhomogeneous evolution equation

$$\frac{\partial}{\partial t} \left(\partial_x^{(k)} u \right) + \lambda(u) \nabla \left(\partial_x^{(k)} u \right) = S^k, \quad (19)$$

where $\lambda(u) = \frac{\partial f(u)}{\partial u}$ denotes the characteristic speed of the flux, and where

$$S^k(t, x) \equiv S^k \left(\partial_x^{(0)} u(t, x), \dots, \partial_x^{(k)} u(t, x) \right) \quad (20)$$

is a source term, being an algebraic function of the spatial derivatives $\partial_x^{(j)} u(t, x)$, $0 \leq j \leq k$. We remark that the source term S^k in (20) vanishes for the simple (linear) case, where the characteristic speed λ is constant. The solution of the resulting generalized Riemann problem for nonlinear systems with source term was first treated in [33].

Unlike the more general setting in [33], we are merely interested in first-instant interactions of left and right states, i.e., at time $t = 0^+$. Therefore, it is reasonable to work with the following simplifications. Firstly, we neglect the source term in (19). Secondly, we linearize the equation (19) about the leading term $u(0^+, 0)$, which is readily available by the solution of the conventional Riemann problem (16), (17). As shown in [32], this linearization does not affect the accuracy of the utilized flux evaluation scheme.

Therefore, in order to determine the required higher order space derivatives, we solve a set of $m - 1$ *homogeneous* and *linearized* conventional Riemann problems of the form

$$\text{PDE:} \quad \frac{\partial}{\partial t} \left(\partial_x^{(k)} u \right) + \lambda(u(0^+, 0)) \nabla \left(\partial_x^{(k)} u \right) = 0, \quad (21)$$

$$\text{IC:} \quad \partial_x^{(k)} u(0, x) = \begin{cases} \partial_x^{(k)} u_\ell, & \text{for } x < 0, \\ \partial_x^{(k)} u_r, & \text{for } x > 0, \end{cases} \quad (22)$$

where the constant $\lambda(u(0^+, 0))$ in (21) is the same for all $m - 1$ Riemann problems (21), (22), and thus it needs to be determined only once beforehand by using the leading term $u(0^+, 0)$.

Altogether, the solution of the generalized Riemann problem, GRP_{m-1} , requires solving a set of m conventional Riemann problems, namely the (possibly *nonlinear*) Riemann problem (16),(17) for the leading state $u(0^+, 0)$, and the $m - 1$ *linear* Riemann problems (21),(22) for the higher order space derivatives $\partial_x^{(k)} u(0^+, 0)$, $1 \leq k \leq m - 1$. These space derivatives are then used in the Cauchy-Kowalewski procedure (18) to compute the time derivatives $\partial_t^{(k)} u(0^+, 0)$, $1 \leq k \leq m - 1$, which in turn are required for the evaluation of the Taylor expansion (15). In this way, the value $u(t, 0)$ is computed via (15) at m -th order time accuracy, at any Gaussian integration point (t_{G_k}, x_{G_k}) , where $t_{G_k} \in [t^n, t^{n+1}]$.

3.4 CFL Condition

Recall that explicit finite volume schemes, such as the proposed ADER scheme, are usually required to satisfy a Courant-Friedrichs-Lewy (CFL) stability condition, which gives a restriction for the time step size.

In order to derive a corresponding CFL condition for our ADER scheme, let ρ_T be the radius of the inscribed circle of a triangular cell $T \in \mathcal{T}$. Moreover, let

$$\lambda_T^{(\max)} = \max_{1 \leq j \leq 3N_x} |\lambda_{1,j}(u) \cdot n_{1,j} + \lambda_{2,j}(u) \cdot n_{2,j}|$$

denote the maximum normal characteristic speed at the $3N_x$ Gaussian integration points of the three cell edges $(\partial T)_j$, $j = 1, 2, 3$.

Similar to the CFL condition in [20, Subsection 3.4.1], we decided to restrict the time step size τ in the implementation of our ADER scheme on (unstructured) triangular meshes \mathcal{T} by the CFL condition

$$\tau \leq \min_{T \in \mathcal{T}} \frac{\rho_T}{\lambda_T^{(\max)}}. \quad (23)$$

3.5 Algorithmic Formulation of the Method ADER m

Let us combine the computational steps of the WENO reconstruction in Section 2 and the ADER flux evaluation scheme of this section in order to provide an algorithmic formulation of the resulting finite volume method ADER m . Any time step $t^n \rightarrow t^{n+1}$ of ADER m is accomplished by the following algorithm.

Algorithm 1 (ADERm).

INPUT: *Triangulation \mathcal{T} , cell averages $\{\bar{u}_T^n \equiv \bar{u}_T(t^n) : T \in \mathcal{T}\}$, positive time step size $\tau = t^{n+1} - t^n$ satisfying (23), and order m .*

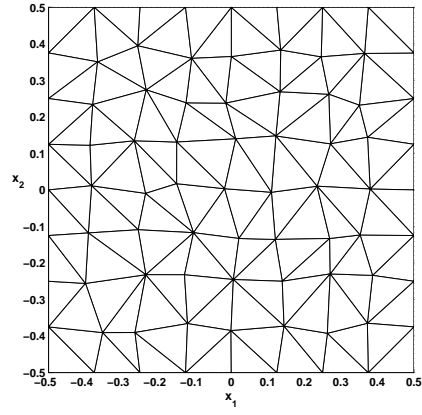
- **FOR** each $T \in \mathcal{T}$ **DO**
 - (1) *Compute reconstruction polynomial p_T of order m satisfying (4) from given cell averages by using WENO reconstruction (8)–(10).*
- **FOR** each $T \in \mathcal{T}$ **DO**
 - (2a) *Solve the GRP $_{m-1}$, given by the RP (16),(17) and the sequence of linear RPs (21),(22), at each Gaussian integration point x_{G_h} .*
 - (2b) *Evaluate $u(\cdot, x_{G_h})$ at each Gaussian integration point t_{G_k} via (15).*
 - (2c) *Compute numerical fluxes $\hat{F}_{T,j}$, $j = 1, 2, 3$, via (12).*
 - (2d) *Update each cell average $\bar{u}_T^{n+1} \equiv \bar{u}_T(t^{n+1})$ by using (11).*

OUTPUT: *Updated cell averages $\{\bar{u}_T^{n+1} \equiv \bar{u}_T(t^{n+1}) : T \in \mathcal{T}\}$.*

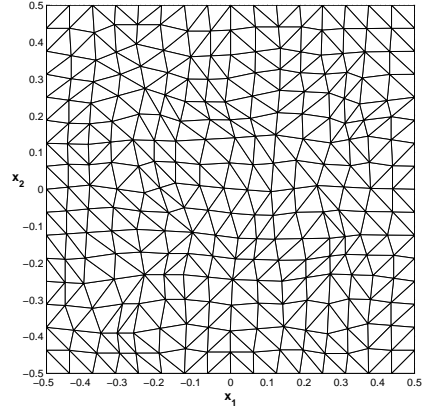
We remark that step (2b) of Algorithm 1 requires the application of the Cauchy-Kowalewski procedure (18) in order to replace the time derivatives in (15) by space derivatives.

4 Convergence Order of ADERm Methods

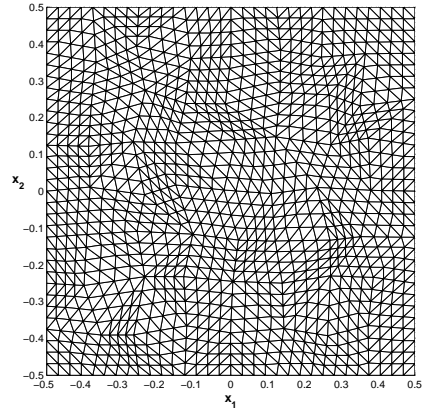
In this section we show that the proposed ADER m scheme attains the expected convergence order m . This is done by numerical experiments, where the schemes ADER2, ADER3, and ADER4 are applied to two different model problems, one linear and one nonlinear advection problem. The numerical experiments are performed by using two sequences, **A** and **B**, of *non-adaptive* triangular meshes, where each mesh sequence consists of five distorted triangular meshes of decreasing mesh width. The triangular meshes of sequence **A** are mildly distorted, whereas the meshes of sequence **B** are highly distorted. The first four meshes, **A**₀–**A**₃, of the sequence **A** are shown in Figure 4, and the corresponding ones of the mesh sequence **B**, **B**₀–**B**₃, are shown in Figure 5.



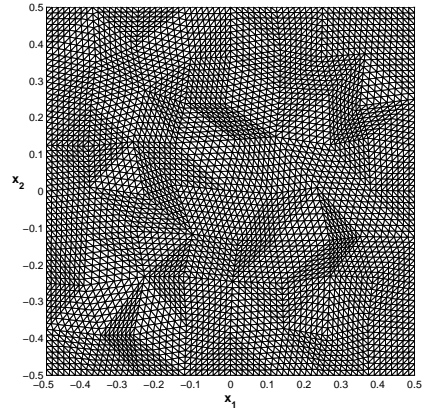
\mathbf{A}_0 ($h = 1/8$)



\mathbf{A}_1 ($h = 1/16$)

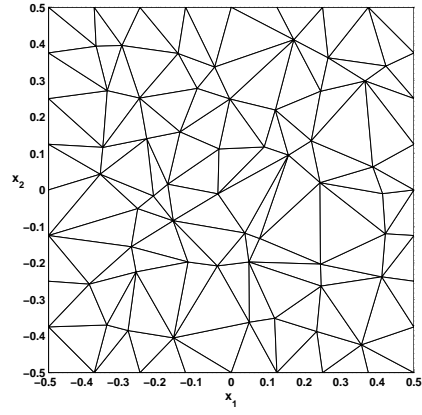


\mathbf{A}_2 ($h = 1/32$)

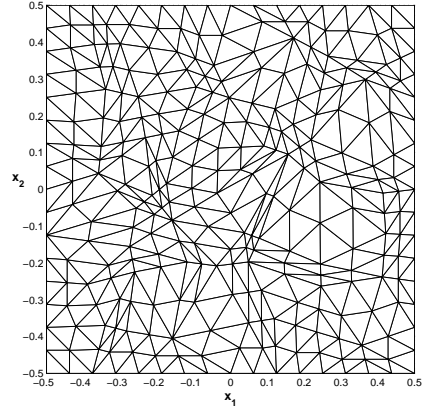


\mathbf{A}_3 ($h = 1/64$)

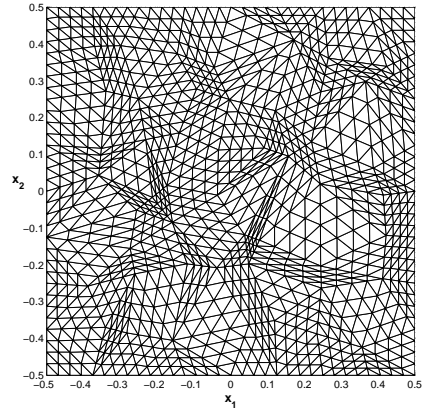
Figure 4: Mesh sequence \mathbf{A}_0 – \mathbf{A}_3 comprising four mildly distorted meshes.



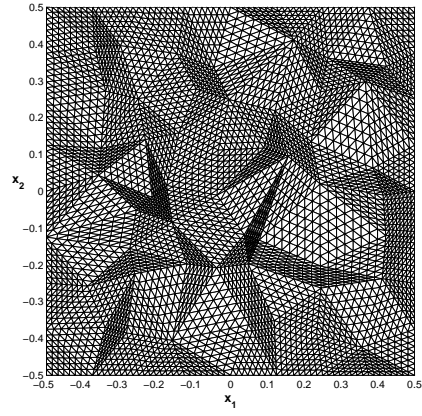
\mathbf{B}_0 ($h = 1/8$)



\mathbf{B}_1 ($h = 1/16$)



\mathbf{B}_2 ($h = 1/32$)



\mathbf{B}_3 ($h = 1/64$)

Figure 5: Mesh sequence \mathbf{B}_0 – \mathbf{B}_3 comprising four highly distorted meshes.

4.1 Linear Advection

In the first model problem, we consider solving the two-dimensional linear advection equation

$$u_t + u_{x_1} + u_{x_2} = 0, \quad (24)$$

with initial condition

$$u_0(x) = u(0, x) = \sin\left(2\pi(x_1 + x_2)\right), \quad (25)$$

on the computational domain $\Omega = [-0.5, 0.5] \times [-0.5, 0.5]$. The computations are carried out for the time interval $I = [0, 1]$. We use periodic boundary conditions, so that the reference solution $\tilde{u}(1, x)$ at final time $t = 1$, coincides with the initial condition (25), i.e., $u_0(x) \equiv \tilde{u}(1, x)$.

In order to study the influence of the mesh irregularity on the accuracy, we compute the solution of (24), (25) on the two mesh sequences **A** (Figure 4) and **B** (Figure 5). The mesh widths h , displayed in Figures 4 and 5, are given by the (constant) length of the edges along the boundary of Ω . Therefore, h is only a rough indicator for the mesh width. But at each refinement level i , the number of cells in the mesh **A** _{i} coincides with the number of cells in the corresponding mesh **B** _{i} , $i = 0, \dots, 4$.

The computations are performed by using the methods ADER2, ADER3, and ADER4. We use nine stencils in the WENO reconstruction, namely three *centered* stencils, three stencils in forward sectors \mathcal{F}_j , and three stencils in backward sectors \mathcal{B}_j .

For each mesh, **A** _{i} and **B** _{i} , $i = 0, \dots, 4$, we determine the time step size τ according to the CFL condition (23). This is done as follows. Due to unit normal characteristic speed in (24), we have $\lambda_T^{(\max)} \equiv 1$. Therefore, for any triangular mesh, the resulting time step τ is bounded above by the smallest radius ρ_{\min} of an inscribed circle of a triangular cell in the mesh, i.e., $\tau \leq \rho_{\min}$ according to (23). This leads us to $\tau = 0.025$ for the time step size in the computations on the coarse mesh **A**₀, and $\tau = 0.0125$ for the coarse mesh **B**₀. For the next finer meshes, **A** _{i} and **B** _{i} , their smallest inscribed circles' radii, $\rho_{\min}(\mathbf{A}_i)$ and $\rho_{\min}(\mathbf{B}_i)$, are half the size of their coarser predecessor, i.e., $\rho_{\min}(\mathbf{A}_i) = \rho_{\min}(\mathbf{A}_{i-1})/2$ and $\rho_{\min}(\mathbf{B}_i) = \rho_{\min}(\mathbf{B}_{i-1})/2$ for $i = 1, \dots, 4$. Therefore, we halve the time step size τ for the simulations on **A** _{i} and **B** _{i} accordingly.

We have recorded the errors between the cell averages of the numerical solution u_h , output by each method ADER m , and a reference solution \tilde{u} , which is computed by using a 7-point quadrature rule on triangles, being exact for polynomials of order up to 6. The numerical results obtained by

ADER2, ADER3, and ADER4 are displayed in Table 1 (for mesh sequence **A**) and in Table 2 (for mesh sequence **B**), where the errors and the corresponding convergence orders,

$$E_p(h) = \|u_h - \tilde{u}\|_p \quad \text{and} \quad k_p = \frac{\log(E_p(h) / E_p(h/2))}{\log(2)}, \quad (26)$$

are shown for the norms $\|\cdot\|_1$, $\|\cdot\|_2$, and $\|\cdot\|_\infty$.

Note that each method ADER m attains its expected convergence order $m \approx k_p$ in (26) for each of the three norms and on either mesh sequence. But the errors $E_p(h)$ on the mildly distorted meshes of sequence **A** (see Table 1) are smaller than those on the sequence **B** (see Table 2) of highly distorted meshes. This is because the triangles of the sequence **A** are closer to being equilateral than those in the mesh sequence **B**. This complies with corresponding results in [2, 17], where it is shown that simulations on meshes with equilateral triangles lead to higher accuracy compared with simulations on meshes with non-equilateral triangles.

Nevertheless, it is quite remarkable that even for the sequence **B** of highly distorted meshes, reasonable numerical results are obtained by each method ADER m , which shows that the proposed ADER scheme, in combination with the stencil selection algorithm in the WENO reconstruction, is robust, even for very anisotropic stencils.

h	$E_1(h)$	k_1	$E_2(h)$	k_2	$E_\infty(h)$	k_∞
1/8	$1.1265 \cdot 10^{-1}$	—	$1.2826 \cdot 10^{-1}$	—	$2.7656 \cdot 10^{-1}$	—
1/16	$4.2780 \cdot 10^{-2}$	1.40	$4.8948 \cdot 10^{-2}$	1.39	$1.0326 \cdot 10^{-1}$	1.42
1/32	$1.1288 \cdot 10^{-2}$	1.92	$1.2915 \cdot 10^{-2}$	1.92	$2.6589 \cdot 10^{-2}$	1.96
1/64	$2.6513 \cdot 10^{-3}$	2.42	$3.0153 \cdot 10^{-3}$	2.43	$1.1444 \cdot 10^{-2}$	1.41
1/128	$6.3234 \cdot 10^{-4}$	2.13	$7.1838 \cdot 10^{-4}$	2.14	$3.7882 \cdot 10^{-3}$	1.65
1/8	$1.4226 \cdot 10^{-1}$	—	$1.6078 \cdot 10^{-1}$	—	$2.7919 \cdot 10^{-1}$	—
1/16	$1.6160 \cdot 10^{-2}$	3.14	$1.8617 \cdot 10^{-2}$	3.11	$3.9276 \cdot 10^{-2}$	2.83
1/32	$1.5446 \cdot 10^{-3}$	3.39	$1.8346 \cdot 10^{-3}$	3.34	$4.2469 \cdot 10^{-3}$	3.21
1/64	$2.0259 \cdot 10^{-4}$	3.40	$2.2524 \cdot 10^{-4}$	3.51	$4.2128 \cdot 10^{-4}$	3.87
1/128	$2.4139 \cdot 10^{-5}$	3.17	$2.6835 \cdot 10^{-5}$	3.17	$5.1008 \cdot 10^{-5}$	3.14
1/8	$2.9912 \cdot 10^{-2}$	—	$3.4907 \cdot 10^{-2}$	—	$7.2935 \cdot 10^{-2}$	—
1/16	$1.1801 \cdot 10^{-3}$	4.66	$1.5787 \cdot 10^{-3}$	4.47	$5.2470 \cdot 10^{-3}$	3.80
1/32	$6.9519 \cdot 10^{-5}$	4.09	$8.9930 \cdot 10^{-5}$	4.13	$3.2150 \cdot 10^{-4}$	4.03
1/64	$6.4714 \cdot 10^{-6}$	3.97	$8.0984 \cdot 10^{-6}$	4.03	$3.1137 \cdot 10^{-5}$	3.91
1/128	$4.4070 \cdot 10^{-7}$	4.00	$5.5669 \cdot 10^{-7}$	3.99	$2.2974 \cdot 10^{-6}$	3.88

Table 1: Linear case. Results by ADER2, ADER3, ADER4 on sequence **A**.

h	$E_1(h)$	k_1	$E_2(h)$	k_2	$E_\infty(h)$	k_∞
1/8	$1.3924 \cdot 10^{-1}$	—	$1.6233 \cdot 10^{-1}$	—	$3.9986 \cdot 10^{-1}$	—
1/16	$3.2158 \cdot 10^{-2}$	2.11	$3.8800 \cdot 10^{-2}$	2.06	$1.4476 \cdot 10^{-1}$	1.47
1/32	$6.8809 \cdot 10^{-3}$	2.22	$8.3858 \cdot 10^{-3}$	2.21	$3.9424 \cdot 10^{-2}$	1.88
1/64	$1.6080 \cdot 10^{-3}$	2.10	$1.9787 \cdot 10^{-3}$	2.08	$1.0345 \cdot 10^{-2}$	1.93
1/128	$3.8924 \cdot 10^{-4}$	2.05	$4.8469 \cdot 10^{-4}$	2.03	$3.1769 \cdot 10^{-3}$	1.70
1/8	$2.7500 \cdot 10^{-1}$	—	$3.0955 \cdot 10^{-1}$	—	$4.9177 \cdot 10^{-1}$	—
1/16	$3.8493 \cdot 10^{-2}$	2.84	$4.4821 \cdot 10^{-2}$	2.79	$9.5172 \cdot 10^{-2}$	2.37
1/32	$4.5424 \cdot 10^{-3}$	3.08	$5.3011 \cdot 10^{-3}$	3.08	$1.1456 \cdot 10^{-2}$	3.05
1/64	$5.2333 \cdot 10^{-4}$	3.12	$6.0649 \cdot 10^{-4}$	3.13	$1.2106 \cdot 10^{-3}$	3.24
1/128	$6.1609 \cdot 10^{-5}$	3.09	$7.1088 \cdot 10^{-5}$	3.09	$1.4629 \cdot 10^{-4}$	3.05
1/8	$6.6326 \cdot 10^{-2}$	—	$7.9679 \cdot 10^{-2}$	—	$1.5932 \cdot 10^{-1}$	—
1/16	$3.9170 \cdot 10^{-3}$	4.08	$5.2793 \cdot 10^{-3}$	3.92	$1.3527 \cdot 10^{-2}$	3.56
1/32	$2.0676 \cdot 10^{-4}$	4.24	$2.7034 \cdot 10^{-4}$	4.29	$8.8686 \cdot 10^{-4}$	3.93
1/64	$1.3002 \cdot 10^{-5}$	3.99	$1.5726 \cdot 10^{-5}$	4.10	$5.3229 \cdot 10^{-5}$	4.06
1/128	$7.7907 \cdot 10^{-7}$	4.06	$9.5160 \cdot 10^{-7}$	4.05	$3.7559 \cdot 10^{-6}$	3.82

Table 2: Linear case. Results by ADER2, ADER3, ADER4 on sequence **B**.

4.2 Nonlinear Advection

As regards the nonlinear case, we consider solving *Burgers equation* [5]

$$u_t + \left(\frac{1}{2}u^2\right)_{x_1} + \left(\frac{1}{2}u^2\right)_{x_2} = 0, \quad (27)$$

with initial condition

$$u_0(x) = u(0, x) = 0.3 + 0.7 \sin\left(2\pi(x_1 + x_2)\right), \quad (28)$$

on the computational domain $\Omega = [-0.5, 0.5] \times [-0.5, 0.5]$. The computations are carried out for the *short* time interval $I = [0, \frac{1}{4\pi}]$, so that during the entire simulation the solution u of the Cauchy problem (27),(28) is smooth. As in the linear case, we work with periodic boundary conditions. Note that the initial condition (28) leads to a transonic rarefaction.

The cell averages of a reference solution \tilde{u} are calculated via a 7-point quadrature rule for triangles, where the value at each quadrature point is calculated by using Newton's method. Our numerical results are reflected by Tables 3 (concerning mesh sequence **A**) and 4 (mesh sequence **B**). The errors $E_p(h)$ in (26), obtained after the final time step of the simulation, are shown along with the experimental convergence orders k_p in (26).

As for the linear model problem of the previous subsection, each method $\text{ADER}m$ attains its expected convergence order m , except for $\text{ADER}4$, which seems to not quite attain the expected order $m = 4$ on the highly distorted mesh sequence, **B**₀–**B**₄, see Table 4.

We can explain this behaviour of $\text{ADER}4$ as follows. It is well-known that the occurrence of long and thin triangles may lead to reconstruction polynomials of rather poor approximation quality, due to almost degenerate forward and backward sectors. This leads to very elongated one-sided stencils, which are covering only a small range of preference directions. The resulting reconstruction quality, especially when measured in the $\|\cdot\|_\infty$ -norm, is in this case rather poor.

Note that this effect is not observed in the linear case. This is because the solution u of the linear model problem (24),(25) is sufficiently smooth during the entire simulation, whereas the solution $u(T, \cdot)$ of the nonlinear model problem (27),(28) exhibits steep gradients at final time $T = \frac{1}{4\pi}$. The steep gradients of $u(T, \cdot)$ are not reconstructed sufficiently accurate, in particular when working with the highly distorted mesh sequence **B**.

Nevertheless, the approximation behaviour of $\text{ADER}m$ can significantly be improved by working with adaptive triangular meshes. This is supported by the numerical results of the following section.

h	$E_1(h)$	k_1	$E_2(h)$	k_2	$E_\infty(h)$	k_∞
1/8	$1.4816 \cdot 10^{-2}$	—	$2.1592 \cdot 10^{-2}$	—	$8.9534 \cdot 10^{-2}$	—
1/16	$5.0152 \cdot 10^{-3}$	1.56	$6.8720 \cdot 10^{-3}$	1.65	$3.2865 \cdot 10^{-2}$	1.45
1/32	$1.3421 \cdot 10^{-3}$	1.90	$1.8877 \cdot 10^{-3}$	1.86	$1.0561 \cdot 10^{-2}$	1.64
1/64	$3.4067 \cdot 10^{-4}$	1.98	$4.8618 \cdot 10^{-4}$	1.96	$2.7014 \cdot 10^{-3}$	1.97
1/128	$8.3667 \cdot 10^{-5}$	2.03	$1.2018 \cdot 10^{-4}$	2.02	$7.0141 \cdot 10^{-4}$	1.95
1/8	$1.2429 \cdot 10^{-2}$	—	$1.5481 \cdot 10^{-2}$	—	$4.7784 \cdot 10^{-2}$	—
1/16	$1.6329 \cdot 10^{-3}$	2.93	$2.2922 \cdot 10^{-3}$	2.76	$1.0174 \cdot 10^{-2}$	2.23
1/32	$1.9838 \cdot 10^{-4}$	3.04	$3.0528 \cdot 10^{-4}$	2.91	$2.1328 \cdot 10^{-3}$	2.25
1/64	$2.7484 \cdot 10^{-5}$	3.31	$4.0679 \cdot 10^{-5}$	3.37	$2.8764 \cdot 10^{-4}$	3.35
1/128	$3.5762 \cdot 10^{-6}$	3.04	$5.1999 \cdot 10^{-6}$	3.06	$4.9262 \cdot 10^{-5}$	2.63
1/8	$2.9430 \cdot 10^{-3}$	—	$3.9772 \cdot 10^{-3}$	—	$1.6612 \cdot 10^{-2}$	—
1/16	$2.2322 \cdot 10^{-4}$	3.72	$3.5916 \cdot 10^{-4}$	3.47	$1.5177 \cdot 10^{-3}$	3.45
1/32	$1.9599 \cdot 10^{-5}$	3.51	$3.7513 \cdot 10^{-5}$	3.26	$2.7872 \cdot 10^{-4}$	2.44
1/64	$1.7003 \cdot 10^{-6}$	4.09	$2.9834 \cdot 10^{-6}$	4.24	$2.9170 \cdot 10^{-5}$	3.78
1/128	$1.3478 \cdot 10^{-7}$	3.78	$2.4466 \cdot 10^{-7}$	3.72	$2.6691 \cdot 10^{-6}$	3.56

Table 3: Burgers. Results by ADER2, ADER3, ADER4 on sequence **A**.

h	$E_1(h)$	k_1	$E_2(h)$	k_2	$E_\infty(h)$	k_∞
1/8	$2.4789 \cdot 10^{-2}$	—	$3.5598 \cdot 10^{-2}$	—	$1.2987 \cdot 10^{-1}$	—
1/16	$8.1998 \cdot 10^{-3}$	1.60	$1.1486 \cdot 10^{-2}$	1.63	$6.5593 \cdot 10^{-2}$	0.99
1/32	$2.2506 \cdot 10^{-3}$	1.87	$3.2835 \cdot 10^{-3}$	1.81	$2.7181 \cdot 10^{-2}$	1.27
1/64	$5.5952 \cdot 10^{-4}$	2.01	$8.4517 \cdot 10^{-4}$	1.96	$9.1484 \cdot 10^{-3}$	1.57
1/128	$1.3480 \cdot 10^{-4}$	2.05	$2.0520 \cdot 10^{-4}$	2.04	$2.5284 \cdot 10^{-3}$	1.86
1/8	$2.1345 \cdot 10^{-2}$	—	$2.7487 \cdot 10^{-2}$	—	$8.6973 \cdot 10^{-2}$	—
1/16	$3.0335 \cdot 10^{-3}$	2.81	$4.4508 \cdot 10^{-3}$	2.63	$1.9878 \cdot 10^{-2}$	2.13
1/32	$3.8506 \cdot 10^{-4}$	2.98	$6.4792 \cdot 10^{-4}$	2.78	$5.4981 \cdot 10^{-3}$	1.85
1/64	$4.5916 \cdot 10^{-5}$	3.07	$7.6192 \cdot 10^{-5}$	3.09	$6.2541 \cdot 10^{-4}$	3.14
1/128	$5.5909 \cdot 10^{-6}$	3.04	$9.2328 \cdot 10^{-6}$	3.04	$8.5906 \cdot 10^{-5}$	2.86
1/8	$5.6973 \cdot 10^{-3}$	—	$8.1636 \cdot 10^{-3}$	—	$4.3144 \cdot 10^{-2}$	—
1/16	$5.1513 \cdot 10^{-4}$	3.47	$9.2607 \cdot 10^{-4}$	3.14	$5.0294 \cdot 10^{-3}$	3.10
1/32	$3.9238 \cdot 10^{-5}$	3.71	$7.8427 \cdot 10^{-5}$	3.56	$6.1687 \cdot 10^{-4}$	3.03
1/64	$2.7966 \cdot 10^{-6}$	3.81	$6.0176 \cdot 10^{-6}$	3.70	$5.2142 \cdot 10^{-5}$	3.56
1/128	$1.8851 \cdot 10^{-7}$	3.89	$4.4105 \cdot 10^{-7}$	3.77	$5.6319 \cdot 10^{-6}$	3.21

Table 4: Burgers. Results by ADER2, ADER3, ADER4 on sequence **B**.

5 ADER4 on Adaptive Triangular Meshes

In this section, we apply the proposed *adaptive* ADER4 method to a Cauchy problem for Burgers equation (27). Moreover, we provide a numerical comparison between various adaptive and non-adaptive variants of ADER4. The latter is done in Subsection 5.2.

5.1 Burgers Equation

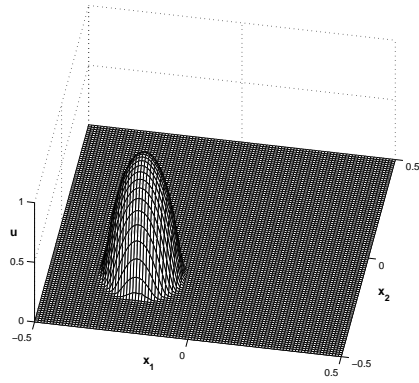
Burgers equation (27) constitutes a popular standard test case concerning nonlinear conservation laws, mainly due to its shock wave behaviour. Even for smooth initial data, the solution of Burgers equation typically develops discontinuities, corresponding to shocks. We consider solving Burgers equation (27) in combination with the initial condition

$$u_0(x) = \begin{cases} \exp\left(\frac{\|x-c\|^2}{\|x-c\|^2 - R^2}\right), & \text{for } \|x-c\| < R, \\ 0, & \text{otherwise,} \end{cases} \quad (29)$$

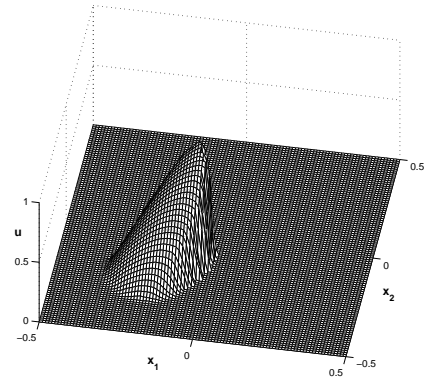
with $R = 0.15$, $c = (-0.2, -0.2)^T$ on the two-dimensional computational domain $\Omega = [-0.5, 0.5]^2 \subset \mathbb{R}^2$. This test case is also used in [8].

A 3D view on the numerical solution u , obtained by ADER4, is shown at four different times, $t_0 = 0$ (initial time), $t_{100} = 0.21427$ (100 time steps), $t_{300} = 0.64146$ (300 time steps), and $t_{700} = 1.49514$ (700 time steps), in Figure 6. The corresponding adaptive triangular meshes are shown in Figure 7. Recall that the time step size is subject to the CFL condition (23), see Algorithm 1.

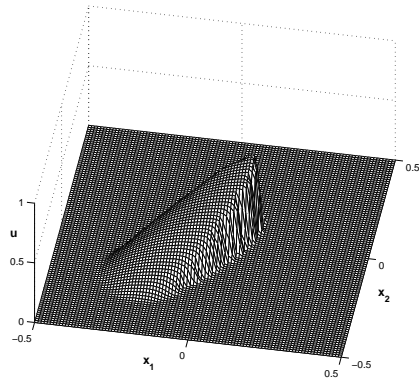
Note that already for the initial condition u_0 , its support is effectively localized by the adaptive refinement of the triangular mesh. The adaptive triangular mesh continues to capture the support of the solution u very well. In particular, the propagation of the shock front is well-resolved during the entire simulation, see Figure 7. Moreover, in regions, where the solution u is rather smooth, the triangular mesh is rather coarse. The latter helps to reduce the required computational costs, which supports the utility of the customized adaption rules (discussed in the Appendix). This is further supported by the numerical comparison in the following subsection.



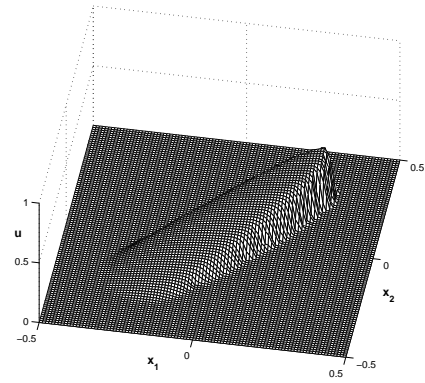
$t_0 = 0$



$t_{100} = 0.21427$



$t_{300} = 0.64146$



$t_{700} = 1.49514$

Figure 6: Burgers equation. 3D view on the numerical solution u obtained by ADER4 at four different times.

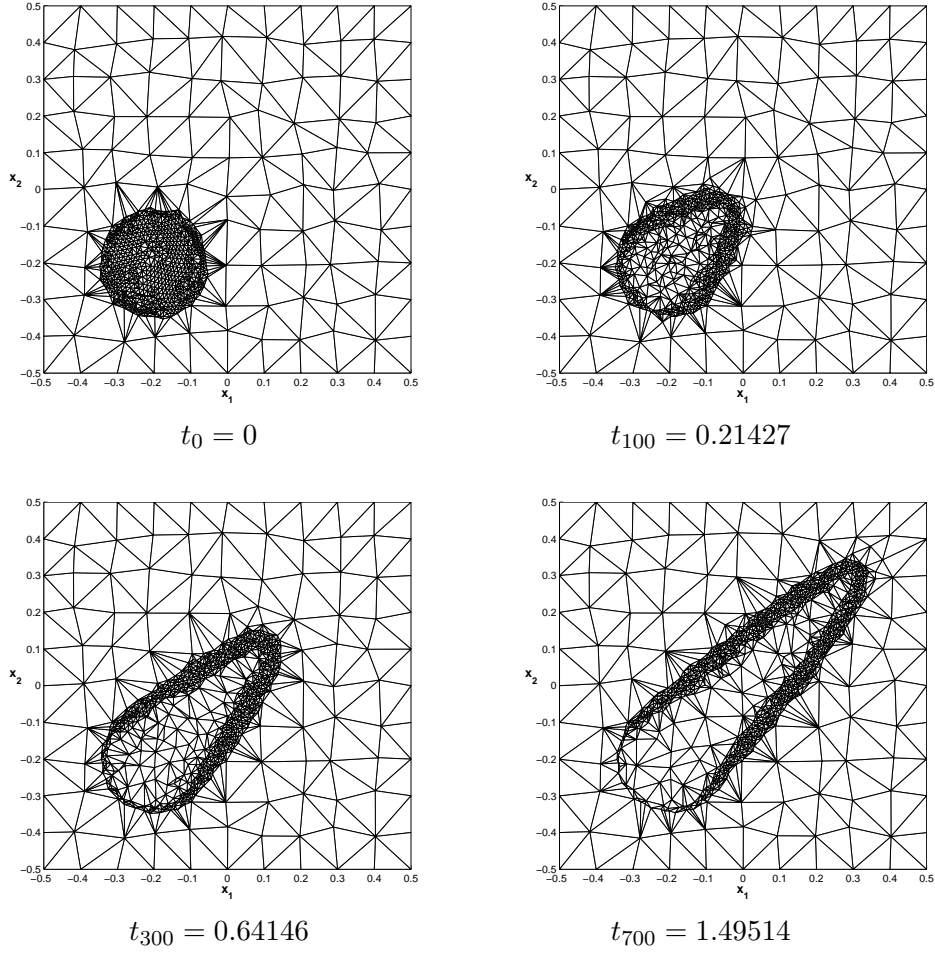


Figure 7: Burgers equation. Adaptive triangulation during the simulation by ADER4 at four different times.

5.2 Comparison with Non-Adaptive Triangular Meshes

Not surprisingly, in all our numerical experiments we found that the performance of the method ADER m over an *adaptive* triangular mesh is, in terms of its enhanced accuracy and smaller complexity, always superior to any (comparable) method ADER m over a *non-adaptive* triangular mesh. To be more precise, at fixed computational costs, the adaptive method ADER m reduces the approximation error of its non-adaptive counterpart quite sig-

nificantly. Likewise, at fixed approximation error, the adaptive variant of ADER m requires much less computational time than any comparable non-adaptive variant of ADER m .

To conclude this section we provide a numerical comparison between various adaptive and non-adaptive variants of the method ADER4. This comparison does not only support our general statements from above, but it also serves to quantify the gain in performance when working with adaptive triangular meshes rather than with non-adaptive ones.

For the purpose of comparison, we consider solving the Cauchy problem for the linear advection equation (24) in combination with the initial condition (29). We evaluate the performance of the method ADER4 for a sequence \mathbf{AD}_2 – \mathbf{AD}_4 of three different *adaptive* triangular meshes, where \mathbf{AD}_2 is the “coarsest” and \mathbf{AD}_4 is the “finest” adaptive mesh by their minimal edge length. We compare the numerical results with those obtained by the method ADER4 on the sequence of *non-adaptive* meshes \mathbf{A}_2 – \mathbf{A}_4 from Section 4, see also Figure 4, where the minimal edge length of the non-adaptive mesh \mathbf{A}_i coincides with the minimal edge length of the adaptive mesh \mathbf{AD}_i , $i = 2, 3, 4$.

Mesh	CPU[sec]	E_1	E_2	E_∞
\mathbf{A}_2	343	0.0061	0.0202	0.1520
\mathbf{A}_3	2880	0.0025	0.0089	0.0844
\mathbf{A}_4	24874	0.0011	0.0045	0.0432
\mathbf{AD}_2	245	0.0067	0.0218	0.1604
\mathbf{AD}_3	1876	0.0028	0.0097	0.0882
\mathbf{AD}_4	14231	0.0013	0.0049	0.0469

Table 5: Numerical results obtained by ADER4 on non-adaptive meshes \mathbf{A}_i and adaptive meshes \mathbf{AD}_i , $i = 2, 3, 4$, for the Cauchy problem (24),(29).

We recorded the resulting approximation errors E_1 , E_2 , and E_∞ in (26) between each numerical solution $u_h(t, x)$ and the analytic solution $u(t, x) \equiv u_0(x - t)$ at time $t = 0.5$. The numerical results are shown in Table 5, where also the elapsed CPU times are included, respectively. The comparison in Table 5 shows that the approximation errors E_1 , E_2 , and E_∞ obtained by ADER4 on the adaptive mesh \mathbf{AD}_i are almost equal to the corresponding errors for the non-adaptive mesh \mathbf{A}_i , $i = 2, 3, 4$. The computational costs required for ADER4 on the non-adaptive mesh \mathbf{A}_i , however, are reduced by about 30%–40% when using the adaptive mesh \mathbf{AD}_i ,

$i = 2, 3, 4$. In conclusion, at comparable accuracy, the adaptive method ADER4 requires significantly smaller computational costs than the non-adaptive method ADER4, see Table 5. This complies with previous observations in all our numerical experiments.

Acknowledgement

We gratefully appreciate useful comments and suggestions from Professor E. F. Toro, which helped to improve a previous version of this paper. This work was partly supported by the European Union through the project NetAGES (Network for Automated Geometry Extraction from Seismic), contract no. IST-1999-29034.

References

- [1] R. Abgrall. On essentially non-oscillatory schemes on unstructured meshes: Analysis and implementation. *J. Comput. Phys.*, 144:45–58, 1994.
- [2] M. Batdorf, L. A. Freitag, and C. F. Ollivier-Gooch. Computational study of the effect of unstructured mesh quality on solution efficiency. In *Proc. 13th AIAA Computational Fluid Dynamics Conference*, Snowmass Village, Colorado, 1997.
- [3] J. Behrens, A. Iske, and M. Käser. Adaptive meshfree method of backward characteristics for nonlinear transport equations. In M. Griebel and M. A. Schweitzer, editors, *Meshfree Methods for Partial Differential Equations*, pages 21–36. Springer, Berlin, 2002.
- [4] J. Behrens, A. Iske, and S. Pöhn. Effective node adaption for grid-free semi-lagrangian advection. In T. Sonar and I. Thomas, editors, *Discrete Modelling and Discrete Algorithms in Continuum Mechanics*, pages 110–119. Logos, Berlin, 2001.
- [5] J. M. Burgers. Application of a model system to illustrate some points of the statistical theory of free turbulence. In *Proc. Acad. Sci. Amsterdam*, volume 43, pages 2–12, 1940.
- [6] M. Dumbser. ADER discontinuous Galerkin schemes for aeroacoustics. In *Proceedings of the Euromech Colloquium 449*, Chamonix, France, December 2003.

- [7] O. Friedrich. Weighted essentially non-oscillatory schemes for the interpolation of mean values on unstructured grids. *J. Comput. Phys.*, 144:194–212, 1998.
- [8] J. Fürst and T. Sonar. On meshless collocation approximations of conservation laws: positive schemes and dissipation models. *ZAMM*, 81:403–415, 2001.
- [9] S. K. Godunov. A finite difference method for the computation of discontinuous solutions of the equations of fluid dynamics. *Mat. Sb.*, 47:271–306, 1959.
- [10] T. Gutzmer and A. Iske. Detection of discontinuities in scattered data approximation. *Numerical Algorithms*, 16(2):155–170, 1997.
- [11] A. Harten and S. Chakravarthy. Multi-dimensional ENO schemes for general geometries. Technical Report 91-76, ICASE, 1991.
- [12] A. Harten, B. Engquist, S. Osher, and S. Chakravarthy. Uniformly high order essentially non-oscillatory schemes, III. *J. Comput. Phys.*, 71:231–303, 1987.
- [13] C. Hu and C. W. Shu. Weighted essentially non-oscillatory schemes on triangular meshes. *J. Comput. Phys.*, 150:97–127, 1999.
- [14] A. Iske. Scattered data modelling using radial basis functions. In A. Iske, E. Quak, and M. Floater, editors, *Tutorials on Multiresolution in Geometric Modelling*, pages 205–242. Springer, Heidelberg, 2002.
- [15] A. Iske and M. Käser. Conservative semi-Lagrangian advection on adaptive unstructured meshes. *Numer. Meth. Part. Diff. Eq.* **20**(3), 2004, 388–411.
- [16] G. S. Jiang and C. W. Shu. Efficient implementation of weighted ENO schemes. *J. Comput. Phys.*, 126:202–228, 1996.
- [17] M. Käser, H. Igel, M. Sambridge, and J. Brown. A comparative study of explicit differential operators on arbitrary grids. *J. Comput. Acoustics*, 9:1111–1125, 2001.
- [18] R. L. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, Cambridge, 2002.
- [19] X. Liu, S. Osher, and T. Chan. Weighted essentially non-oscillatory schemes. *J. Comput. Phys.*, 115:200–212, 1994.

- [20] A. Meister and J. Struckmeier. *Hyperbolic Partial Differential Equations: Theory, Numerics and Applications*. Vieweg, Braunschweig, 2002.
- [21] T. J. Ruuth and R. J. Spiteri. Two barriers on strong-stability-preserving time discretization methods. *J. Sci. Comput.*, 17:211–220, 2002.
- [22] T. Schwartzkopff, C.-D. Munz, and E. F. Toro. ADER: A high-order approach for linear hyperbolic systems in 2D. *J. Sci. Comput.*, 17:231–240, 2002.
- [23] T. Schwartzkopff, C.-D. Munz, E. F. Toro, and R. C. Millington. The ADER approach in 2D. In T. Sonar and I. Thomas, editors, *Discrete Modelling and Discrete Algorithms on Continuum Mechanics*, pages 207–216. Logos, Berlin, 2001.
- [24] T. Schwartzkopff, M. Dumbser, and C.-D. Munz. Fast high order ADER schemes for linear hyperbolic equations and their numerical dissipation and dispersion. To appear in *J. Comput. Phys.*
- [25] C. W. Shu. Total-variation-diminishing time discretizations. *SIAM J. Sci. Stat. Comput.*, 9:1073–1084, 1988.
- [26] C. W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *J. Comput. Phys.*, 77:439–471, 1988.
- [27] T. Sonar. On the construction of essentially non-oscillatory finite volume approximations to hyperbolic conservation laws on general triangulations: Polynomial recovery, accuracy and stencil selection. *Comput. Methods Appl. Mech. Engrg.*, 140:157–181, 1997.
- [28] V. A. Titarev and E. F. Toro. ADER: Arbitrary high order Godunov approach. *J. Sci. Comput.*, 17:609–618, 2002.
- [29] V. A. Titarev and E. F. Toro. ADER schemes for three-dimensional nonlinear hyperbolic systems. Preprint NIO4007-NPA, Isaac Newton Institute for Mathematical Sciences, University of Cambridge, UK, 2004.
- [30] E. F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Springer, 2nd edition, 1999.

- [31] E. F. Toro, R. C. Millington, and L. A. M. Nejad. Towards very high order Godunov schemes. In E. F. Toro, editor, *Godunov methods; Theory and applications*, pages 907–940, Oxford, 2001. Kluwer Academic Plenum Publishers. International Conference.
- [32] E. F. Toro and V. A. Titarev. Very high order Godunov-type schemes for nonlinear scalar conservation laws. In *ECCOMAS Computational Fluid Dynamics Conference*, Swansea, Wales, September 2001. European Congress on Computational Methods in Applied Sciences and Engineering.
- [33] E. F. Toro and V. A. Titarev. Solution of the generalised Riemann problem for advection-reaction equations. *Proc. Royal Society of London A*, 458:271–281, 2002.

Authors’ addresses:

Martin Käser
 Civil & Environmental Engineering
 University of Trento
 I-38050 Trento, ITALY
martin.kaeser@ing.unitn.it

Armin Iske
 Department of Mathematics
 University of Leicester
 Leicester LE1 7RH, UK
iske@mcs.le.ac.uk

Appendix: Adaption Rules

This appendix briefly explains the utilized adaption rules, which are similar to the ones of our previous papers [3, 4, 15]. The adaption rules rely on an a posteriori error indicator, which is combined with refinement and coarsening strategies for the triangular cells.

5.3 Error Indication

A customized error indicator is used in order to adaptively modify the triangles of the current triangulation \mathcal{T} . A *significance* value η_T , assigned to each $T \in \mathcal{T}$, reflects the local approximation quality of the cell average \bar{u}_T over triangle T . The significances η_T , $T \in \mathcal{T}$, are used to flag single triangles as “to be refined” or “to be coarsened”.

Definition 2 Let $\eta^* = \max_{T \in \mathcal{T}} \eta_T$, and let $\theta_{\text{crs}}, \theta_{\text{ref}}$ be two tolerance values satisfying $0 < \theta_{\text{crs}} < \theta_{\text{ref}} < 1$. We say that a cell $T \in \mathcal{T}$ is **to be refined**, iff $\eta_T > \theta_{\text{ref}} \cdot \eta^*$, and T is **to be coarsened**, iff $\eta_T < \theta_{\text{crs}} \cdot \eta^*$.

In our numerical experiments, we let $\theta_{\text{crs}} = 0.01$ and $\theta_{\text{ref}} = 0.05$. Note that a triangle T cannot be refined and be coarsened at the same time; in fact, it may neither be refined nor be coarsened. In order to define the error indicator η_T , we first need to introduce another concept for triangle neighbourhoods, which leads us to Moore neighbourhoods, see Figure 8.

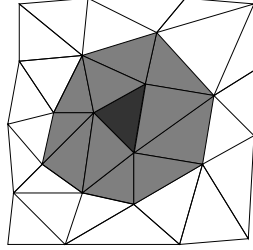


Figure 8: A triangle T (dark shaded) and its Moore neighbours.

Definition 3 Let \mathcal{T} be a conforming triangulation. For any triangle $T \in \mathcal{T}$, the set

$$\mathcal{M}(T) = \left\{ \tilde{T} \in \mathcal{T} \setminus \{T\} : \tilde{T} \cap T \neq \emptyset \right\}$$

is called **Moore neighbourhood** of T . Any triangle in $\mathcal{M}(T)$ is called a **Moore neighbour** of T .

Following along the lines of [10], and with assuming that for each triangle $T \in \mathcal{T}$ its cell average is assigned to its barycenter ξ_T , i.e., $\bar{u}_T \equiv \bar{u}(\xi_T)$, we define the error indicator for any triangle $T \in \mathcal{T}$ by

$$\eta_T = |\bar{u}(\xi_T) - s(\xi_T)|, \quad (30)$$

where for the Moore neighbourhood $\mathcal{M}(T)$ of T the function $s \equiv s_{\mathcal{M}(T)}$ in (30) denotes the *thin plate spline interpolant* [14] satisfying the interpolation conditions

$$s(\xi_{\tilde{T}}) = \bar{u}(\xi_{\tilde{T}}), \quad \text{for all } \tilde{T} \in \mathcal{M}(T).$$

Now, for any triangular cell $T \in \mathcal{T}$, the error indication η_T is small, whenever the approximation quality of \bar{u} by s around T is good, whereas a high value η_T indicates that \bar{u} is subject to strong variation locally around T . This way, the error indicator allows us to locate discontinuities of the solution u quite effectively. For further details, we refer to our previous papers [3, 4, 15], where similar adaption rules are employed.

5.4 Coarsening and Refinement

The adaptive insertion and removal of current triangles $T \in \mathcal{T}$ is accomplished by the following operations.

5.4.1 Coarsening.

A triangular cell $T \in \mathcal{T}$ is *coarsened* by the removal of its three vertices (nodes) from the current Delaunay triangulation \mathcal{T} . But the coarsening of a triangle T is only performed, if *all* triangular cells of its Moore neighbourhood $\mathcal{M}(T)$, and T itself, are flagged as *to be coarsened*. After the removal of T , the Delaunay triangulation \mathcal{T} is updated by a local retriangulation according to the Delaunay criterion.

5.4.2 Refinement.

A triangular cell $T \in \mathcal{T}$ is *refined* by the insertion of its barycenter ξ_T into \mathcal{T} , followed by a subsequent local Delaunay retriangulation.