Efficient Implementation of ADER Discontinuous Galerkin Schemes for a Scalable Hyperbolic PDE Engine

Michael Dumbser [1],*[ID]**, Francesco Fambri** [1][ID]**, Maurizio Tavelli** [1]**, Michael Bader** [2] **and Tobias Weinzierl** [3]

[1] Department of Civil, Environmental and Mechanical Engineering, University of Trento, I-38123 Trento, Italy; francesco.fambri@unitn.it (F.F.); m.tavelli@unitn.it (M.T.)
[2] Department of Informatics, Technical University of Munich, D-85748 Munich, Germany; bader@in.tum.de
[3] Department of Computer Science, University of Durham, Durham DH1 3LE, UK; tobias.weinzierl@durham.ac.uk
* Correspondence: michael.dumbser@unitn.it

Abstract: In this paper we discuss a new and very efficient implementation of high order accurate arbitrary high order schemes using derivatives discontinuous Galerkin (ADER-DG) finite element schemes on modern massively parallel supercomputers. The numerical methods apply to a very broad class of nonlinear systems of hyperbolic partial differential equations. ADER-DG schemes are by construction communication-avoiding and cache-blocking, and are furthermore very well-suited for vectorization, and so they appear to be a good candidate for the future generation of exascale supercomputers. We introduce the numerical algorithm and show some applications to a set of hyperbolic equations with increasing levels of complexity, ranging from the compressible Euler equations over the equations of linear elasticity and the unified Godunov-Peshkov-Romenski (GPR) model of continuum mechanics to general relativistic magnetohydrodynamics (GRMHD) and the Einstein field equations of general relativity. We present strong scaling results of the new ADER-DG schemes up to 180,000 CPU cores. To our knowledge, these are the largest runs ever carried out with high order ADER-DG schemes for nonlinear hyperbolic PDE systems. We also provide a detailed performance comparison with traditional Runge-Kutta DG schemes.

Keywords: hyperbolic partial differential equations; high order discontinuous Galerkin finite element schemes; shock waves and discontinuities; vectorization and parallelization; high performance computing

1. Introduction

Hyperbolic partial differential equations are omnipresent in the mathematical description of time-dependent processes in fluid and solid mechanics, in engineering and geophysics, as well as in plasma physics, and even in general relativity. Among the most widespread applications nowadays are (i) computational fluid mechanics in mechanical and aerospace engineering, in particular compressible gas dynamics at high Mach numbers; (ii) geophysical and environmental free surface flows in oceans, rivers and lakes, describing natural hazards such as tsunami wave propagation, landslides, storm surges and floods; (iii) seismic, acoustic and electromagnetic wave propagation processes in the time domain are described by systems of hyperbolic partial differential equations, namely the equations of linear elasticity, the acoustic wave equation and the well-known Maxwell equations; (iv) high energy density plasma flows in nuclear fusion reactors as well as astrophysical plasma flows in the solar system and the universe, using either the Newtonian limit or the complete

equations in full general relativity; (v) the Einstein field equations of general relativity, which govern the dynamics of the spacetime around black holes and neutron stars, can be written under the form of a nonlinear system of hyperbolic partial differential equations.

The main challenge of nonlinear hyperbolic PDE arises from the fact that they can contain at the same time smooth solutions (like sound waves) as well as small scale structures (e.g., turbulent vortices), but they can also develop discontinuous solutions (shock waves) after finite time, even when starting from perfectly smooth initial data. These discontinuities were first discovered by Bernhard Riemann in his ground breaking work on the propagation of waves of finite amplitude in air [1,2], where the term *finite* should actually be understood in the sense of *large,* rather than simple sound waves of *infinitesimal* strength that have been considered in the times before Riemann. In the abstract of his work, Riemann stated that his discovery of the shock waves might probably not be of practical use for applied and experimental science, but should be mainly understood as a contribution to the theory of nonlinear partial differential equations. Several decades later, shock waves were also observed experimentally, thus confirming the new and groundbreaking mathematical concept of Riemann.

The connection between symmetries and conservation laws were established in the work of Emmy Noether [3] at the beginning of the 20th century, while the first methods for the *numerical solution* of hyperbolic conservation laws go back to famous mathematicians such as Courant and Friedrichs and co-workers [4–7]. The connection between hyperbolic conservation laws, symmetric hyperbolic systems in the sense of Friedrichs [8] and thermodynamics was established for the first time by Godunov in 1961 [9], and was rediscovered again by Friedrichs and Lax in 1971 [10]. Within this theoretical framework of symmetric hyperbolic and thermodynamically compatible (SHTC) systems, established by Godunov and Romenski [11,12], it is possible to write down the Euler equations of compressible gas dynamics, the magnetohydrodynamics (MHD) equations [13], the equations of nonlinear elasticity [14], as well as a rather wide class of nonlinear hyperbolic conservation laws [15] with very interesting mathematical properties and structure. Very recently, even a novel and unified formulation of continuum physics, including solid and fluid mechanics only as two particular cases of a more general model, have been cast into the form of a single SHTC system [16–19]. In the 1940ies and 1950ies, major steps forward in numerical methods for hyperbolic PDE have been made in the ground-breaking contributions of von Neumann and Richtmyer [20] and Godunov [21]. While the former introduce an artificial viscosity to stabilize the numerical scheme in the presence of discontinuities, the latter constructs his scheme starting from the most elementary problem in hyperbolic conservation laws for which an exact solution is still available, the so-called Riemann problem. The Riemann problem consists in a particular Cauchy problem where the initial data consist of two piecewise constant states, separated by a discontinuity. In the absence of source terms, its solution is self-similar. While provably robust, these schemes are only first order accurate in space and time and thus only applicable to flows with shock waves, but not to those also involving smooth sound waves and turbulent small scale flow structures. In his paper [21], Godunov has also proven that any linear numerical scheme that is required to be monotone can be at most of order one, which is the well-known Godunov barrier theorem. The main goal in the past decades was to find ways how to circumvent it, since it only applies to linear schemes. The first successful nonlinear monotone and higher order accurate schemes were the method of Kolgan [22] and the schemes of van Leer [23,24]. Subsequently, many other higher order nonlinear schemes have been proposed, such as the ENO [25] and WENO schemes [26], and there is a rapidly growing literature on the subject. In this paper we mainly focus on a rather recent family of schemes, which is of the discontinuous finite element type, namely the so-called discontinuous Galerkin (DG) finite element methods, which were systematically introduced for hyperbolic conservation laws in a well-known series of papers by Cockburn and Shu and collaborators [27–31]. For a review on high order DG methods and WENO schemes, the reader is referred to References [32,33]. In this paper we use a particular variant of the DG scheme that is called the ADER discontinuous Galerkin scheme [34–38], where ADER stands for arbitrary high order schemes using derivatives, first developed by Toro et al. in the context of high order finite volume

schemes [39–42]. In comparison to traditional *semi-discrete* DG schemes, which mainly use Runge-Kutta time integration, ADER-DG methods are *fully-discrete* and are based on a predictor-corrector approach that allows the achievement of a naturally cache-blocking and communication-avoiding scheme, which reduces the amount of necessary MPI communications to a minimum. These properties make the method well suitable for high performance computing (HPC).

2. High Order ADER Discontinuous Galerkin Finite Element Schemes

In this paper we consider hyperbolic PDEs with non-conservative products and algebraic source terms of the form (see also References [34,35])

$$\frac{\partial \mathbf{Q}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{Q}) + \mathbf{B}(\mathbf{Q}) \cdot \nabla \mathbf{Q} = \mathbf{S}(\mathbf{Q}), \tag{1}$$

where $t \in \mathbb{R}_0^+$ is the time, $\mathbf{x} \in \Omega \subset \mathbb{R}^d$ is the spatial position vector in d space dimensions, $\mathbf{Q} \in \Omega_Q \subset \mathbb{R}^m$ is the state vector, $\mathbf{F}(\mathbf{Q})$ is the nonlinear flux tensor, $\mathbf{B}(\mathbf{Q}) \cdot \nabla \mathbf{Q}$ is a non-conservative product and $\mathbf{S}(\mathbf{Q})$ is a purely algebraic source term. Introducing the system matrix $\mathbf{A}(\mathbf{Q}) = \partial \mathbf{F}/\partial \mathbf{Q} + \mathbf{B}(\mathbf{Q})$ the above system can also be written in quasi-linear form as

$$\frac{\partial \mathbf{Q}}{\partial t} + \mathbf{A}(\mathbf{Q}) \cdot \nabla \mathbf{Q} = \mathbf{S}(\mathbf{Q}). \tag{2}$$

The system is hyperbolic if for all $\mathbf{n} \neq 0$ and for all $\mathbf{Q} \in \Omega_Q$, the matrix $\mathbf{A}(\mathbf{Q}) \cdot \mathbf{n}$ has m real eigenvalues and a full set of m linearly independent right eigenvectors. The system in Equation (1) is provided with an initial condition $\mathbf{Q}(\mathbf{x}, 0) = \mathbf{Q}_0(\mathbf{x})$ and appropriate boundary conditions on $\partial\Omega$. In some parts of the paper we will also make use of the vector of primitive (physical) variables denoted by $\mathbf{V} = \mathbf{V}(\mathbf{Q})$. For very complex PDE systems, such as the general-relativistic MHD equations, it may be much easier to express the flux tensor $\mathbf{F}$ in terms of $\mathbf{V}$ rather than in terms of $\mathbf{Q}$, however the evaluation of $\mathbf{V} = \mathbf{V}(\mathbf{Q})$ can become very complicated.

2.1. Unlimited ADER-DG Scheme and Riemann Solvers

We cover the computational domain Ω with a set of non-overlapping Cartesian control volumes in space $\Omega_i = [x_i - \frac{1}{2}\Delta x_i, x_i + \frac{1}{2}\Delta x_i] \times [y_i - \frac{1}{2}\Delta y_i, y_i + \frac{1}{2}\Delta y_i] \times [z_i - \frac{1}{2}\Delta z_i, z_i + \frac{1}{2}\Delta z_i]$. Here, $\mathbf{x}_i = (x_i, y_i, z_i)$ denotes the barycenter of cell Ω_i and $\Delta \mathbf{x}_i = (\Delta x_i, \Delta y_i, \Delta z_i)$ is the mesh spacing associated with Ω_i in each spatial dimension. The domain $\Omega = \bigcup \Omega_i$ is the union of all spatial control volumes. A key ingredient of the ExaHyPE engine http://exahype.eu is a cell-by-cell adaptive mesh refinement (AMR), which is built upon the space-tree implementation of Peano [43,44]. For further details about cell-by-cell AMR, see Reference [45]. High order finite volume and finite difference schemes for AMR can be found, e.g., in References [46–52]. For high order AMR with better than second order accurate finite volume and DG schemes in combination with time-accurate local time stepping (LTS), the reader is referred to References [37,53–57]. Since the main focus of this paper is not on AMR, at this point we can only give a very brief summary of existing AMR methods and codes for hyperbolic PDE, without pretending to be complete. The starting point of adaptive mesh refinement for hyperbolic conservation laws was of course the pioneering work of Berger et al. [58–60], who were the first to introduce a patched-based block-structured AMR method. Further developments are reported in References [61–63] based on the second order accurate wave-propagation algorithm of LeVeque. We also would like to draw the attention of the reader to the works of Quirk [64], Coirier and Powell [65] and Deiterding et al. [66,67]. For computational astrophysics, relevant AMR techniques have been documented, e.g., in References [68–76], including the RAMSES, PLUTO, NIRVANA, AMRVAC and BHAC codes. For a recent and more complete survey of high level AMR codes, the reader is referred to the review paper Reference [77].

In the following, the discrete solution of the PDE system in Equation (1) is denoted by $\mathbf{u}_h$ and is defined in terms of tensor products of piecewise polynomials of degree N in each spatial direction.

The discrete solution space is denoted by $\mathcal{U}_h$ in the following. Since we adopt a discontinuous Galerkin (DG) finite element method, the numerical solution $\mathbf{u}_h$ is allowed to *jump* across element interfaces, as in the context of finite volume schemes. Within each spatial control volume Ω_i the discrete solution $\mathbf{u}_h$ restricted to that control volume is written at time t^n in terms of some nodal spatial basis functions $\Phi_l(\mathbf{x})$ and some unknown degrees of freedom $\hat{\mathbf{u}}_{i,l}^n$:

$$\mathbf{u}_h(\mathbf{x}, t^n)|_{\Omega_i} = \sum_l \hat{\mathbf{u}}_{i,l}\Phi_l(\mathbf{x}) := \hat{\mathbf{u}}_{i,l}^n\Phi_l(\mathbf{x}), \tag{3}$$

where $l = (l_1, l_2, l_3)$ is a multi-index and the spatial basis functions $\Phi_l(\mathbf{x}) = \varphi_{l_1}(\xi)\varphi_{l_2}(\eta)\varphi_{l_3}(\zeta)$ are generated via tensor products of one-dimensional nodal basis functions $\varphi_k(\xi)$ on the reference interval $[0, 1]$. The transformation from physical coordinates $\mathbf{x} \in \Omega_i$ to reference coordinates $\boldsymbol{\xi} = (\xi, \eta, \zeta) \in [0, 1]^d$ is given by the linear mapping $\mathbf{x} = \mathbf{x}_i - \frac{1}{2}\Delta\mathbf{x}_i + (\xi\Delta x_i, \eta\Delta y_i, \zeta\Delta z_i)^T$. For the one-dimensional basis functions $\varphi_k(\xi)$ we use the Lagrange interpolation polynomials passing through the Gauss-Legendre quadrature nodes ξ_j of an $N + 1$ point Gauss quadrature formula. Therefore, the nodal basis functions satisfy the interpolation property $\varphi_k(\xi_j) = \delta_{kj}$, where δ_{kj} is the usual Kronecker symbol, and the resulting basis is *orthogonal*. Furthermore, due to this particular choice of a *nodal* tensor-product basis, the entire scheme can be written in a dimension-by-dimension fashion, where all integral operators can be decomposed into a sequence of one-dimensional operators acting only on the $N + 1$ degrees of freedom in the respective dimension. For details on multi-dimensional quadrature, see the well-known book of Stroud [78].

In order to derive the ADER-DG method, we first multiply the governing PDE system in Equation (1) with a test function $\Phi_k \in \mathcal{U}_h$ and integrate over the space-time control volume $\Omega_i \times [t^n; t^{n+1}]$. This leads to

$$\int\limits_{t^n}^{t^{n+1}} \int\limits_{\Omega_i} \Phi_k \frac{\partial \mathbf{Q}}{\partial t} \, d\mathbf{x} \, dt + \int\limits_{t^n}^{t^{n+1}} \int\limits_{\Omega_i} \Phi_k \left(\nabla \cdot \mathbf{F}(\mathbf{Q}) + \mathbf{B}(\mathbf{Q}) \cdot \nabla \mathbf{Q}\right) d\mathbf{x} \, dt = \int\limits_{t^n}^{t^{n+1}} \int\limits_{\Omega_i} \Phi_k \mathbf{S}(\mathbf{Q}) \, d\mathbf{x} \, dt, \tag{4}$$

with $d\mathbf{x} = dx \, dy \, dz$. As already mentioned before, the discrete solution is allowed to jump across element interfaces, which means that the resulting jump terms have to be taken properly into account. In our scheme this is achieved via numerical flux functions (approximate Riemann solvers) and via the path-conservative approach that was developed by Castro and Parés in the finite volume context [79,80]. It has later been also extended to the discontinuous Galerkin finite element framework in References [35,81,82]. In classical Runge-Kutta DG schemes, only a weak form in space of the PDE is obtained, while time is still kept continuous, thus reducing the problem to a nonlinear system of ODE, which is subsequently integrated with standard ODE solvers in time. However, this requires MPI communication in each Runge-Kutta stage. Furthermore, each Runge-Kutta stage requires accesses to the entire discrete solution in memory. In the ADER-DG framework, a completely different paradigm is used. Here, higher order in time is achieved with the use of an element-local space-time predictor, denoted by $\mathbf{q}_h(\mathbf{x}, t)$ in the following, and which will be discussed in more detail later. Using Equation (3), integrating the first term by parts in time and integrating the flux divergence term by parts in space, taking into account the jumps between elements and making use of this local space-time predictor solution $\mathbf{q}_h$ instead of $\mathbf{Q}$, the weak formulation of Equation (4) can be rewritten as

$$\left(\int\limits_{\Omega_i} \Phi_k \Phi_l \, d\mathbf{x}\right)\left(\hat{\mathbf{u}}_{i,l}^{n+1} - \hat{\mathbf{u}}_{i,l}^n\right) + \int\limits_{t^n}^{t^{n+1}} \int\limits_{\partial\Omega_i} \Phi_k \mathcal{D}^-\left(\mathbf{q}_h^-, \mathbf{q}_h^+\right) \cdot \mathbf{n} \, dS \, dt - \int\limits_{t^n}^{t^{n+1}} \int\limits_{\Omega_i^\circ} (\nabla\Phi_k \cdot \mathbf{F}(\mathbf{q}_h)) \, d\mathbf{x} \, dt +$$

$$+ \int\limits_{t^n}^{t^{n+1}} \int\limits_{\Omega_i^\circ} \Phi_k \left(\mathbf{B}(\mathbf{q}_h) \cdot \nabla \mathbf{q}_h\right) d\mathbf{x} \, dt = \int\limits_{t^n}^{t^{n+1}} \int\limits_{\Omega_i} \Phi_k \mathbf{S}(\mathbf{q}_h) \, d\mathbf{x} \, dt, \tag{5}$$

where the first integral leads to the element mass matrix, which is diagonal since our basis is orthogonal. The boundary integral contains the approximate Riemann solver and accounts for the jumps across element interfaces, also in the presence of non-conservative products. The third and fourth integral account for the smooth part of the flux and the non-conservative product, while the right hand side takes into account the presence of the algebraic source term. According to the framework of path-conservative schemes [35,79,80,82], the jump terms are defined via a path-integral in phase space between the boundary extrapolated states at the left $\mathbf{q}_h^-$ and at the right $\mathbf{q}_h^+$ of the interface as follows:

$$\mathcal{D}^- \left(\mathbf{q}_h^-, \mathbf{q}_h^+\right) \cdot \mathbf{n} = \frac{1}{2} \left(\mathbf{F}(\mathbf{q}_h^+) + \mathbf{F}(\mathbf{q}_h^-)\right) \cdot \mathbf{n} + \frac{1}{2} \left(\int_0^1 \mathbf{B}(\boldsymbol{\psi}) \cdot \mathbf{n} \, ds - \boldsymbol{\Theta}\right) \left(\mathbf{q}_h^+ - \mathbf{q}_h^-\right), \qquad (6)$$

with $\mathbf{B} \cdot \mathbf{n} = \mathbf{B}_1 n_1 + \mathbf{B}_2 n_2 + \mathbf{B}_3 n_3$. Throughout this paper, we use the simple straight-line segment path

$$\boldsymbol{\psi} = \boldsymbol{\psi}(\mathbf{q}_h^-, \mathbf{q}_h^+, s) = \mathbf{q}_h^- + s \left(\mathbf{q}_h^+ - \mathbf{q}_h^-\right), \qquad 0 \le s \le 1. \qquad (7)$$

In order to achieve exactly well-balanced schemes for certain classes of hyperbolic equations with non-conservative products and source terms, the segment path is not sufficient and a more elaborate choice of the path becomes necessary, see e.g., References [83–86]. In Equation (6) above the symbol $\boldsymbol{\Theta} > 0$ denotes an appropriate numerical dissipation matrix. Following References [35,87,88], the path integral that appears in Equation (6) can be simply evaluated via some sufficiently accurate numerical quadrature formulae. We typically use a three-point Gauss-Legendre rule in order to approximate the path-integral. For a simple path-conservative Rusanov-type method [35,89], the numerical dissipation matrix reads

$$\boldsymbol{\Theta}_{\text{Rus}} = s_{\max} \mathbf{I}, \qquad \text{with} \qquad s_{\max} = \max \left(\left|\lambda(\mathbf{q}_h^-)\right|, \left|\lambda(\mathbf{q}_h^+)\right|\right), \qquad (8)$$

where $\mathbf{I}$ denotes the identity matrix and $s_{\max}$ is the maximum wave speed (eigenvalue λ of matrix $\mathbf{A} \cdot \mathbf{n}$) at the element interface. In order to reduce numerical dissipation, one can use better Riemann solvers, such as the Osher-type schemes proposed in References [88,90], or the recent extension of the original HLLEM method of Einfeldt and Munz [91] to general conservative and non-conservative hyperbolic systems recently put forward in Reference [92]. The choice of the approximate Riemann solver and therefore of the viscosity matrix $\boldsymbol{\Theta}$ completes the numerical scheme in Equation (5). In the next subsection, we shortly discuss the computation of the element-local space-time predictor $\mathbf{q}_h$, which is a key ingredient of our high order accurate and communication-avoiding ADER-DG schemes.

2.2. Space-Time Predictor and Suitable Initial Guess

As already mentioned previously, the element-local space–time predictor is an important *key feature* of ADER-DG schemes and is briefly discussed in this section. The computation of the predictor solution $\mathbf{q}_h(\mathbf{x}, t)$ is based on a weak formulation of the governing PDE system in space–time and was first introduced in References [34,93,94]. Starting from the known solution $\mathbf{u}_h(\mathbf{x}, t^n)$ at time t^n and following the terminology of Harten et al. [95], we solve a so-called Cauchy problem *in the small*, i.e., without considering the interaction with the neighbor elements. In the ENO scheme of Harten et al. [95] and in the original ADER approach of Toro and Titarev [40–42] the strong differential form of the PDE was used, together with a combination of Taylor series expansions and the so-called Cauchy-Kovalewskaya procedure. The latter is very cumbersome, or becomes even unfeasible for very complicated nonlinear hyperbolic PDE systems, since it requires a lot of analytic manipulations of the governing PDE system in order to replace time derivatives with known space derivatives at time t^n. This is achieved by successively differentiating the governing PDE system with respect to space and time and inserting the resulting terms into the Taylor series. For an explicit example of the Cauchy–Kovalewskaya procedure applied to the three-dimensional Euler equations of compressible gas dynamics and the MHD equations, see References [96,97]. Instead, the local space–time discontinuous Galerkin predictor, introduced in References [34,93,94], requires only

pointwise evaluations of the fluxes, source terms and non-conservative products, for element Ω_i the predictor solution $\mathbf{q}_h$ is now expanded in terms of a local space–time basis

$$\mathbf{q}_h(\mathbf{x}, t)|_{\Omega_i^{st}} = \sum_l \theta_l(\mathbf{x}, t)\hat{\mathbf{q}}_l^i := \theta_l(\mathbf{x}, t)\hat{\mathbf{q}}_l^i, \tag{9}$$

with the multi-index $l = (l_0, l_1, l_2, l_3)$ and where the space–time basis functions $\theta_l(\mathbf{x}, t) = \varphi_{l_0}(\tau)\varphi_{l_1}(\xi)\varphi_{l_2}(\eta)\varphi_{l_3}(\zeta)$ are again generated from the same one-dimensional nodal basis functions $\varphi_k(\xi)$ as before, i.e., the Lagrange interpolation polynomials of degree N passing through $N + 1$ Gauss-Legendre quadrature nodes. The spatial mapping $\mathbf{x} = \mathbf{x}(\boldsymbol{\xi})$ is also the same as before and the coordinate time is mapped to the reference time $\tau \in [0, 1]$ via $t = t^n + \tau\Delta t$. Multiplication of the PDE system in Equation (1) with a test function θ_k and integration over the space–time control volume $\Omega_i^{st} = \Omega_i \times [t^n, t^{n+1}]$ yields the following weak form of the governing PDE, which is *different* from Equation (4), because now the test and basis functions are both time dependent:

$$\int_{t^n}^{t^{n+1}} \int_{\Omega_i} \theta_k(\mathbf{x}, t)\frac{\partial \mathbf{q}_h}{\partial t} \, d\mathbf{x} \, dt + \int_{t^n}^{t^{n+1}} \int_{\Omega_i} \theta_k(\mathbf{x}, t)\left(\nabla \cdot \mathbf{F}(\mathbf{Q}) + \mathbf{B}(\mathbf{q}_h) \cdot \nabla \mathbf{q}_h\right) d\mathbf{x} \, dt = \int_{t^n}^{t^{n+1}} \int_{\Omega_i} \theta_k(\mathbf{x}, t)\mathbf{S}(\mathbf{q}_h) \, d\mathbf{x} \, dt. \tag{10}$$

Since we are only interested in an element local predictor solution, i.e., without considering interactions with the neighbor elements we do not yet take into account the jumps in $\mathbf{q}_h$ across the element interfaces, because this will be done in the final corrector step of the ADER-DG scheme in Equation (5). Instead, we introduce the known discrete solution $u_h(\mathbf{x}, t^n)$ at time t^n. For this purpose, the first term is integrated by parts in time. This leads to

$$\int_{\Omega_i} \theta_k(\mathbf{x}, t^{n+1})\mathbf{q}_h(\mathbf{x}, t^{n+1}) \, d\mathbf{x} - \int_{t^n}^{t^{n+1}} \int_{\Omega_i} \frac{\partial}{\partial t}\theta_k(\mathbf{x}, t)\mathbf{q}_h(\mathbf{x}, t) \, d\mathbf{x} \, dt - \int_{\Omega_i} \theta_k(\mathbf{x}, t^n)\mathbf{u}_h(\mathbf{x}, t^n) \, d\mathbf{x} =$$
$$\int_{t^n}^{t^{n+1}} \int_{\Omega_i^{\circ}} \theta_k(\mathbf{x}, t)\nabla \cdot \mathbf{F}(\mathbf{q}_h) \, d\mathbf{x} \, dt + \int_{t^n}^{t^{n+1}} \int_{\Omega_i^{\circ}} \theta_k(\mathbf{x}, t)\left(\mathbf{S}(\mathbf{q}_h) - \mathbf{B}(\mathbf{q}_h) \cdot \nabla \mathbf{q}_h\right) d\mathbf{x} \, dt. \tag{11}$$

Using the local space–time ansatz (9), Equation (11) becomes an element-local nonlinear system for the unknown degrees of freedom $\hat{\mathbf{q}}_{i,l}$ of the space–time polynomials $\mathbf{q}_h$. The solution of Equation (11) can be found via a simple and fast converging fixed point iteration (a discrete Picard iteration) as detailed e.g., in References [34,98]. For linear homogeneous systems, the discrete Picard iteration converges in a finite number of at most $N + 1$ steps, since the involved iteration matrix is nilpotent, see Reference [99].

However, we emphasize that the choice of an appropriate *initial guess* $\mathbf{q}_h^0(\mathbf{x}, t)$ for $\mathbf{q}_h(\mathbf{x}, t)$ is of fundamental importance to obtain a faster convergence and thus a computationally more efficient scheme. For this purpose, one can either use an extrapolation of $\mathbf{q}_h$ from the previous time interval $[t^{n-1}, t^n]$, as suggested e.g., in Reference [100], or one can employ a second-order accurate MUSCL-Hancock-type approach, as proposed in Reference [98], which is based on discrete derivatives computed at time t^n. The initial guess is most conveniently written in terms of a Taylor series expansion of the solution in time, where then suitable approximations of the time derivatives are computed. In the following we introduce the operator

$$\mathcal{L}(\mathbf{u}_h(\mathbf{x}, t^n)) = \mathbf{S}(\mathbf{u}_h(\mathbf{x}, t^n)) - \nabla \cdot \mathbf{F}(\mathbf{u}_h(\mathbf{x}, t^n)) - \mathbf{B}(\mathbf{u}_h(\mathbf{x}, t^n)) \cdot \nabla \mathbf{u}_h(\mathbf{x}, t^n), \tag{12}$$

which is an approximation of the time derivative of the solution. The second-order accurate MUSCL-type initial guess [98] then reads

$$\mathbf{q}_h^0(\mathbf{x}, t) = \mathbf{u}_h(\mathbf{x}, t^n) + (t - t^n)\,\mathcal{L}(\mathbf{u}_h(\mathbf{x}, t^n)), \tag{13}$$

while a third-order accurate initial guess for $\mathbf{q}_h(\mathbf{x}, t)$ is given by

$$\mathbf{q}_h^0(\mathbf{x}, t) = \mathbf{u}_h(\mathbf{x}, t^n) + (t - t^n) \, k_1 + \frac{1}{2} \, (t - t^n)^2 \, \frac{(k_2 - k_1)}{\Delta t}. \tag{14}$$

Here, we have used the abbreviations $k_1 := \mathcal{L} \, (\mathbf{u}_h(\mathbf{x}, t^n))$ and $k_2 := \mathcal{L} \, (\mathbf{u}_h(\mathbf{x}, t^n) + \Delta t k_1)$. For an initial guess of even higher order of accuracy, it is possible to use the so-called continuous extension Runge-Kutta (CERK) schemes of Owren and Zennaro [101]; see also Reference [102] for the use of CERK time integrators in the context of high-order discontinuous Galerkin finite element methods. If an initial guess with polynomial degree $N - 1$ in time is chosen, it is sufficient to use *one single* Picard iteration to solve Equation (11) to the desired accuracy.

At this point, we make some comments about a suitable data-layout for high order ADER-DG schemes. In order to compute the discrete derivative operators needed in the predictor Equation (11), especially for the computation of the discrete gradient $\nabla \mathbf{q}_h$, it is very convenient to use an array-of-struct (AoS) data structure. In this way, the first or fastest-running unit-stride index is the one associated with the m quantities contained in the vector $\mathbf{Q}$, while the other indices are associated with the space–time degrees of freedom, i.e., we arrange the data contained in the set of degrees of freedom $\hat{\mathbf{q}}_l^i$ as $\hat{\mathbf{q}}_{v,l_1,l_2,l_3,l_0}^i$, with $1 \leq v \leq m$ and $1 \leq l_k \leq N + 1$. The discrete derivatives in space and time direction can then be simply computed by the multiplication of a subset of the degrees of freedom with the transpose of a small $(N + 1) \times (N + 1)$ matrix D_{kl} from the right, which reads

$$D_{kl} = \frac{1}{h} \left(\int_0^1 \phi_k(\xi) \phi_m(\xi) d\xi \right)^{-1} \left(\int_0^1 \phi_m(\xi) \frac{\partial \phi_l(\xi)}{\partial \xi} d\xi \right), \tag{15}$$

where h is the respective spatial or temporal step size in the corresponding coordinate direction, i.e., either Δx_i, Δy_i, Δz_i or Δt. For this purpose, the optimized library for small matrix multiplications `libxsmm` can be employed on Intel machines, see References [103–105] for more details. However, the AoS data layout is *not* convenient for *vectorization* of the PDE evaluation in ADER-DG schemes, since vectorization of the fluxes, source terms and non-conservative products should preferably be done over the integration points l. For this purpose, we convert the AoS data layout *on the fly* into a struct-of-array (SoA) data layout via appropriate transposition of the data and then call the physical flux function $\mathbf{F}(\mathbf{q}_h)$ as well as the combined algebraic source term and non-conservative product contained in the expression $\mathbf{S}(\mathbf{q}_h) - \mathbf{B}(\mathbf{q}_h) \cdot \nabla \mathbf{q}_h$ simultaneously for a subset of VECTORLENGTH space–time degrees of freedom, where VECTORLENGTH is the length of the AVX registers of modern Intel Xeon CPUs, i.e., 4 for those with the old 256 bit AVX and AVX2 instruction sets (Sandy Bridge, Haswell, Broadwell) and 8 for the latest Intel Xeon Scalable CPUs with 512 bit AVX instructions (Skylake). The result of the vectorized evaluation of the PDE, which is still in SoA format, is then converted back to the AoS data layout using appropriate vectorized shuffle commands.

The element-local space–time predictor is arithmetically very intensive, but at the same time it is also by construction cache-blocking. While in traditional RKDG schemes, each Runge-Kutta stage requires touching all spatial degrees of freedom of the entire domain once per Runge-Kutta stage, in our ADER-DG approach the spatial degrees of freedom $\mathbf{u}_h$ need to be loaded only once per element and time step, and from those all space–time degrees of freedom of $\mathbf{q}_h$ are computed. Ideally, this procedure fits entirely into the L3 cache or even into the L2 cache of the CPU, at least up to a certain critical polynomial degree $N_c = N_c(m)$, which is a function of the available L3 or L2 cache size, but also of the number of quantities m to be evolved in the PDE system.

Last but not least, it is important to note that it is possible to hide the entire MPI communication that is inevitably needed on distributed memory supercomputers behind the space–time predictor. For this purpose, the predictor is first invoked on the MPI boundaries of each CPU, which then immediately sends the boundary-extrapolated data $\mathbf{q}_h^-$ and $\mathbf{q}_h^+$ to the neighbor CPUs. While the messages containing the data of these non-blocking MPI send and receive commands are sent around,

each CPU can compute the space–time predictor of purely interior elements that do not need any MPI communication.

For an efficient task-based formalism used within ExaHyPE in the context of shared memory parallelism, see Reference [106]. This completes the description of the efficient implementation of the unlimited ADER-DG schemes used within the ExaHyPE engine.

2.3. A Posteriori Subcell Finite Volume Limiter

In regions where the discrete solution is smooth, there is indeed no need for using nonlinear limiters. However, in the presence of shock waves, discontinuities or strong gradients, and taking into account the fact that even a *smooth signal* may become *non-smooth* on the discrete level if it is *underresolved* on the grid, we have to supplement our high order unlimited ADER-DG scheme described above with a nonlinear limiter.

In order to build a simple, robust and accurate limiter, we follow the ideas outlined in References [36–38,107], where a novel *a posteriori* limiting strategy for ADER-DG schemes was developed, based on the ideas of the MOOD paradigm introduced in References [108–111] in the finite volume context. In a first run, the unlimited ADER-DG scheme is used and produces a so-called *candidate solution*, denoted by $\mathbf{u}_h^*(\mathbf{x}, t^{n+1})$ in the following. This candidate solution is then checked *a posteriori* against several physical and numerical detection criteria. For example, we require some relevant physical quantities of the solution to be positive (e.g., pressure and density), we require the absence of floating point errors (NaN) and we impose a relaxed discrete maximum principle (DMP) in the sense of polynomials, see Reference [36]. As soon as one of these detection criteria is not satisfied, a cell is marked as troubled zone and is scheduled for limiting.

A cell Ω_i that has been marked for limiting is now split into $(2N + 1)^d$ finite volume subcells, which are denoted by $\Omega_{i,s}$. They satisfy $\Omega_i = \bigcup_s \Omega_{i,s}$. Note that this very fine division of a DG element into finite volume subcells does *not* reduce the time step of the overall ADER-DG scheme, since the CFL number of explicit DG schemes scales with $1/(2N + 1)$, while the CFL number of finite volume schemes (used on the subgrid) is of the order of unity. The discrete solution in the subcells $\Omega_{i,s}$ is represented at time t^n in terms of *piecewise constant* subcell averages $\bar{\mathbf{u}}_{i,s}^n$, i.e.,

$$\bar{\mathbf{u}}_{i,s}^n = \frac{1}{|\Omega_{i,s}|} \int_{\Omega_{i,s}} \mathbf{Q}(\mathbf{x}, t^n)\, d\mathbf{x}. \tag{16}$$

These subcell averages are now evolved in time with a second or third order accurate finite volume scheme, which actually looks very similar to the previous ADER-DG scheme in Equation (5), with the difference that now the test function is unity and the spatial control volumes Ω_i are replaced by the sub-volumes $\Omega_{i,s}$:

$$\left(\bar{\mathbf{u}}_{i,s}^{n+1} - \bar{\mathbf{u}}_{i,s}^n\right) + \int_{t^n}^{t^{n+1}} \int_{\partial\Omega_{i,s}} \mathcal{D}^-\left(\mathbf{q}_h^-, \mathbf{q}_h^+\right) \cdot \mathbf{n}\, dS\, dt + \int_{t^n}^{t^{n+1}} \int_{\Omega_{i,s}^\circ} \left(\mathbf{B}(\mathbf{q}_h) \cdot \nabla\mathbf{q}_h\right) d\mathbf{x}\, dt = \int_{t^n}^{t^{n+1}} \int_{\Omega_{i,s}} \mathbf{S}(\mathbf{q}_h)\, d\mathbf{x}\, dt. \tag{17}$$

Here we use again a space–time predictor solution $\mathbf{q}_h$, but which is now computed from an initial condition given by a second order TVD reconstruction polynomial or from a WENO [26] or CWENO reconstruction [51,112–114] polynomial $\mathbf{w}_h(\mathbf{x}, t^n)$ computed from the cell averages $\bar{\mathbf{u}}_{i,s}^n$ via an appropriate reconstruction operator. The predictor is either computed via a standard second order MUSCL–Hancock-type strategy, or via the space–time DG approach of Equation (11), but where the initial data $\mathbf{u}_h(\mathbf{x}, t^n)$ are now replaced by $\mathbf{w}_h(\mathbf{x}, t^n)$ and the spatial control volumes Ω_i are replaced by the subcells $\Omega_{i,s}$.

Once all subcell averages $\bar{\mathbf{u}}^{n+1}_{i,s}$ inside a cell Ω_i have been computed according to Equation (17), the limited DG polynomial $\mathbf{u}'_h(\mathbf{x}, t^{n+1})$ at the next time level is obtained again via a classical constrained least squares reconstruction procedure requiring

$$\frac{1}{|\Omega_{i,s}|} \int_{\Omega_{i,s}} \mathbf{u}'_h(\mathbf{x}, t^{n+1})\, d\mathbf{x} = \bar{\mathbf{u}}^{n+1}_{i,s} \qquad \forall \Omega_{i,s} \in \Omega_i, \qquad \text{and} \qquad \int_{\Omega_i} \mathbf{u}'_h(\mathbf{x}, t^{n+1})\, d\mathbf{x} = \sum_{\Omega_{i,s} \in \Omega_i} |\Omega_{i,s}| \bar{\mathbf{u}}^{n+1} i, s\,. \tag{18}$$

Here, the second relation is a constraint and means conservation at the level of the control volume Ω_i. This completes the brief description of the subcell finite volume limiter used here.

3. Some Examples of Typical PDE Systems Solved With the ExaHyPE Engine

The great advantage of ExaHyPE over many existing PDE solvers is its great flexibility and versatility for the solution of a very wide class of hyperbolic PDE systems in Equation (1). The implementation of the numerical method and the definition of the PDE system to be solved are completely independent of each other. The compute kernels are provided either as generic or as an optimized implementation for the general PDE system given by Equation (1), while the user only needs to provide particular implementations of the functions $\mathbf{F}(\mathbf{Q})$, $\mathbf{B}(\mathbf{Q})$ and $\mathbf{S}(\mathbf{Q})$. It is obviously also possible to drop terms that are not needed. This allows to solve all the PDE systems listed below in one single software package. In all numerical examples shown below, we have used a CFL condition of the type

$$\Delta t \leq \frac{\alpha}{\frac{|\lambda^x_{\max}|}{\Delta x} + \frac{|\lambda^y_{\max}|}{\Delta y} + \frac{|\lambda^z_{\max}|}{\Delta z}}, \tag{19}$$

where Δx, Δy and Δz are the mesh spacings and $|\lambda^x_{\max}|$, $|\lambda^x_{\max}|$ and $|\lambda^x_{\max}|$ are the maximal absolute values of the eigenvalues (wave speeds) of the matrix $\mathbf{A} \cdot \mathbf{n}$ in x, y and z direction, respectively. The coefficient $\alpha < 1/(2N+1)$ can be obtained via a numerical von Neumann stability analysis and is reported for some relevant N in Reference [34].

3.1. The Euler Equations of Compressible Gas Dynamics

The Euler equations of compressible gas dynamics are among the simplest nonlinear systems of hyperbolic conservation laws. They only involve a conservative flux $\mathbf{F}(\mathbf{Q})$ and read

$$\frac{\partial}{\partial t} \begin{pmatrix} \rho \\ \rho\mathbf{v} \\ \rho E \end{pmatrix} + \nabla \cdot \begin{pmatrix} \rho\mathbf{v} \\ \rho\mathbf{v} \otimes \mathbf{v} + p\mathbf{I} \\ \mathbf{v}\,(\rho E + p) \end{pmatrix} = 0. \tag{20}$$

Here, ρ is the mass density, $\mathbf{v}$ is the fluid velocity, ρE is the total energy density and p is the fluid pressure, which is related to ρ, ρE and $\mathbf{v}$ via the so-called equation of state (EOS). In the following we show the computational results for two test problems. The first one is the smooth isentropic vortex test case first proposed in Reference [115] and also used in Reference [36], which has an exact solution and is therefore suitable for a numerical convergence study. Some results of Reference [36] are summarized in Table 1 below, where N_x denotes the number of cells per space dimensions. From the results one can conclude that the high order ADER-DG schemes converge with the designed order of accuracy in both space and time. In order to give a quantitative assessment for the cost of the scheme, we define and provide the TDU metric, which is the cost per degree of freedom update per CPU core, see also Reference [34]. The TDU metric is easily computed by dividing the measured wall clock time (WCT) of a simulation by the number of elements per CPU core and time steps carried out, and by the number of spatial degrees of freedom per element, i.e., $(N+1)^d$. With the appropriate initial guess and AVX 512 vectorization of the code discussed in the previous section, the cost for updating one single degree of freedom for a fourth order ADER-DG scheme ($N = 3$) for the 3D compressible Euler equations is as low as TDU = 0.25 µs when using one single CPU core of a new Intel i9-7900X Skylake test workstation

with 3.3 GHz nominal clock speed, 32 GB of RAM and a total number of 10 CPU cores. This cost metric can be directly compared with the cost to update one single point or control volume of existing finite difference and finite volume schemes.

Table 1. L^1, L^2 and L^∞ errors and numerical convergence rates obtained for the two-dimensional isentropic vortex test problem using different unlimited ADER-DG schemes, see Reference [36].

	N_x	L^1 Error	L^2 Error	L^∞ Error	L^1 Order	L^2 Order	L^∞ Order	Theor.
$N = 3$	25	5.77×10^{-4}	9.42×10^{-5}	7.84×10^{-5}	—	—	—	4
	50	2.75×10^{-5}	4.52×10^{-6}	4.09×10^{-6}	4.39	4.38	4.26	
	75	4.36×10^{-6}	7.89×10^{-7}	7.55×10^{-7}	4.55	4.30	4.17	
	100	1.21×10^{-6}	2.37×10^{-7}	2.38×10^{-7}	4.46	4.17	4.01	
$N = 4$	20	1.54×10^{-4}	2.18×10^{-5}	2.20×10^{-5}	—	—	—	5
	30	1.79×10^{-5}	2.46×10^{-6}	2.13×10^{-6}	5.32	5.37	5.75	
	40	3.79×10^{-6}	5.35×10^{-7}	5.18×10^{-7}	5.39	5.31	4.92	
	50	1.11×10^{-6}	1.61×10^{-7}	1.46×10^{-7}	5.50	5.39	5.69	
$N = 5$	10	9.72×10^{-4}	1.59×10^{-4}	2.00×10^{-4}	—	—	—	6
	20	1.56×10^{-5}	2.13×10^{-6}	2.14×10^{-6}	5.96	6.22	6.55	
	30	1.14×10^{-6}	1.64×10^{-7}	1.91×10^{-7}	6.45	6.33	5.96	
	40	2.17×10^{-7}	2.97×10^{-8}	3.59×10^{-8}	5.77	5.93	5.82	

In the following we show the results obtained with an ADER-DG scheme using piecewise polynomials of degree $N = 9$ for a very stringent test case, which is the so-called Sedov blast wave problem detailed in References [100,107,116,117]. It consists in an explosion propagating in a zero pressure gas, leading to an infinitely strong shock wave. In our setup, the outer pressure is set to 10^{-14}, i.e., close to machine zero. In order to get a robust numerical scheme, it is useful to perform the reconstruction step in the subcell finite volume limiter as well as the space–time predictor of the ADER-DG scheme in primitive variables, see Reference [100]. The computational results obtained are shown in Figure 1, where we can observe a very good agreement with the reference solution. One furthermore can see that the discrete solution respects the circular symmetry of the problem and the *a posteriori* subcell limiter is only acting in the vicinity of the shock wave.
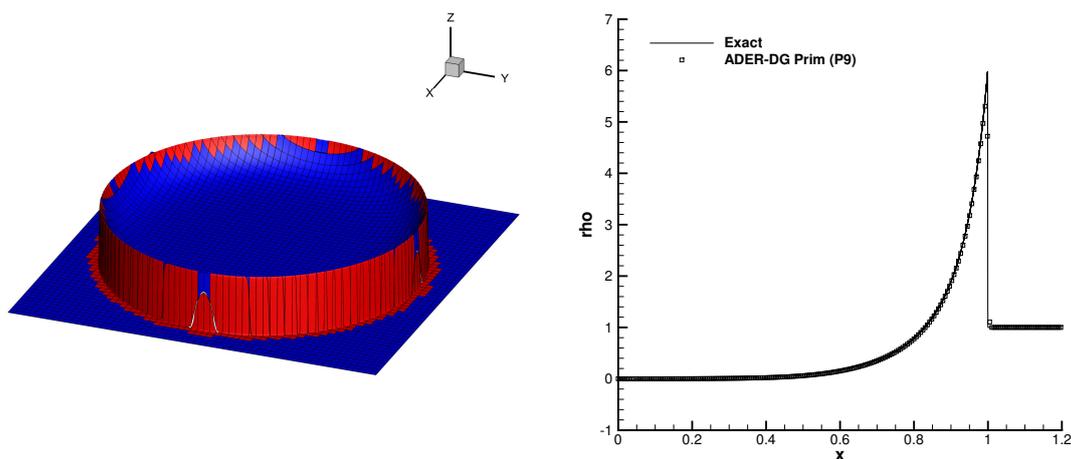

Figure 1. Sedov blast wave problem using an ADER-DG *P*9 scheme with *a posteriori* subcell finite volume limiter using predictor and limiter in primitive variables, see Reference [100]. Unlimited cells are depicted in blue, while limited cells are highlighted in red (**left**). 1D cut through the numerical solution and comparison with the exact solution (**right**).

3.2. A Novel Diffuse Interface Approach for Linear Seismic Wave Propagation in Complex Geometries

　　Seismic wave propagation problems in complex 3D geometries are often very challenging due to the geometric complexity. Standard approaches either use regular curvilinear boundary-fitted meshes, or unstructured tetrahedral or hexahedral meshes. In all cases, a certain amount of user interaction for grid generation is required. Furthermore, the geometric complexity can have a negative impact on the admissible time step size due to the CFL condition, since the mesh generator may create elements with very bad aspect ratio, so-called sliver elements. In the case of regular curvilinear grids, the Jacobian of the mapping may become ill-conditioned and thus reduce the admissible time step size. In Reference [118] a novel diffuse interface approach has been forwarded, where only the definition of a scalar volume fraction function α is required, where $\alpha = 1$ is set inside the solid medium, and $\alpha = 0$ in the surrounding gas or vacuum. The governing PDE system proposed in Reference [118] reads

$$\frac{\partial \sigma}{\partial t} - \mathbf{E}(\lambda, \mu) \cdot \frac{1}{\alpha} \nabla(\alpha \mathbf{v}) + \frac{1}{\alpha} \mathbf{E}(\lambda, \mu) \cdot \mathbf{v} \otimes \nabla \alpha = \mathbf{S}_\sigma, \tag{21}$$

$$\frac{\partial \alpha \mathbf{v}}{\partial t} - \frac{\alpha}{\rho} \nabla \cdot \sigma - \frac{1}{\rho} \sigma \nabla \alpha = \mathbf{S}_v, \tag{22}$$

$$\frac{\partial \alpha}{\partial t} = 0, \qquad \frac{\partial \lambda}{\partial t} = 0, \qquad \frac{\partial \mu}{\partial t} = 0, \qquad \frac{\partial \rho}{\partial t} = 0, \tag{23}$$

and clearly falls into the class of PDE systems described by Equation (1). Here, σ denotes the symmetric stress tensor, $\mathbf{v}$ is the velocity vector, $\alpha \in [0, 1]$ is the volume fraction, λ and μ are the Lamé constants and ρ is the density of the solid medium. The elasticity tensor $\mathbf{E}$ is a function of λ and μ and relates stress and strain via the Hooke law. The last four quantities obey trivial evolution equations, which state that these parameters remain constant in time. However, they still need to be properly included in the evolution system, since they have an influence on the solution of the Riemann problem. An analysis of the eigenstructure of Equations (21)–(23) shows that the eigenvalues are all real and are *independent* of the volume fraction function α. Furthermore, the exact solution of a generic Riemann problem with $\alpha = 1$ on the left and $\alpha = 0$ on the right yields the free surface boundary condition $\sigma \cdot \mathbf{n} = 0$ at the interface, see Reference [118] for details. In this new approach, the mesh generation problem can be fully avoided, since all that is needed is the specification of the scalar volume fraction function α, which is set to unity inside the solid and to zero outside. A realistic 3D wave propagation example based on real DTM data of the Mont Blanc region is shown in Figures 2 and 3, where the 3D contour colors of the wave field as well as a set of seismogram recordings in two receiver points are reported. For this simulation, a uniform Cartesian base-grid of 80^3 elements was used, together with one level of AMR refinement close to the free surface boundary determined by the DTM model. A fourth order ADER-DG scheme ($N = 3$) has been used in this simulation. We stress that the entire setup of the computational model in the diffuse interface approach is completely automatic, and no manual user interaction was required. The reference solution was obtained with a high order ADER-DG scheme of the same polynomial degree $N = 3$ using an unstructured boundary-fitted tetrahedral mesh [119] of similar spatial resolution, containing a total of 1,267,717 elements. We observe an excellent agreement between the two simulations, which were obtained with two completely different PDE systems on two different grid topologies.
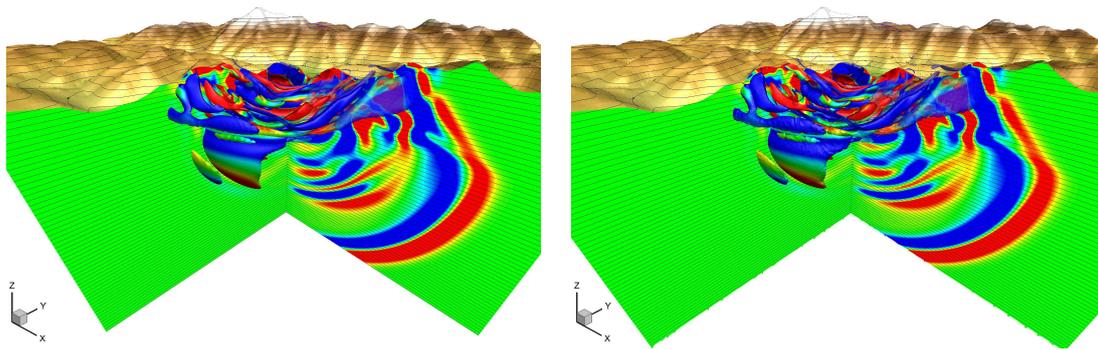
Figure 2. Wave field of a seismic wave propagation problem with the novel diffuse interface approach on adaptive Cartesian grids developed in Reference [118] (**left**) compared with the reference solution obtained on a classical boundary-fitted unstructured tetrahedral mesh [119] (**right**).
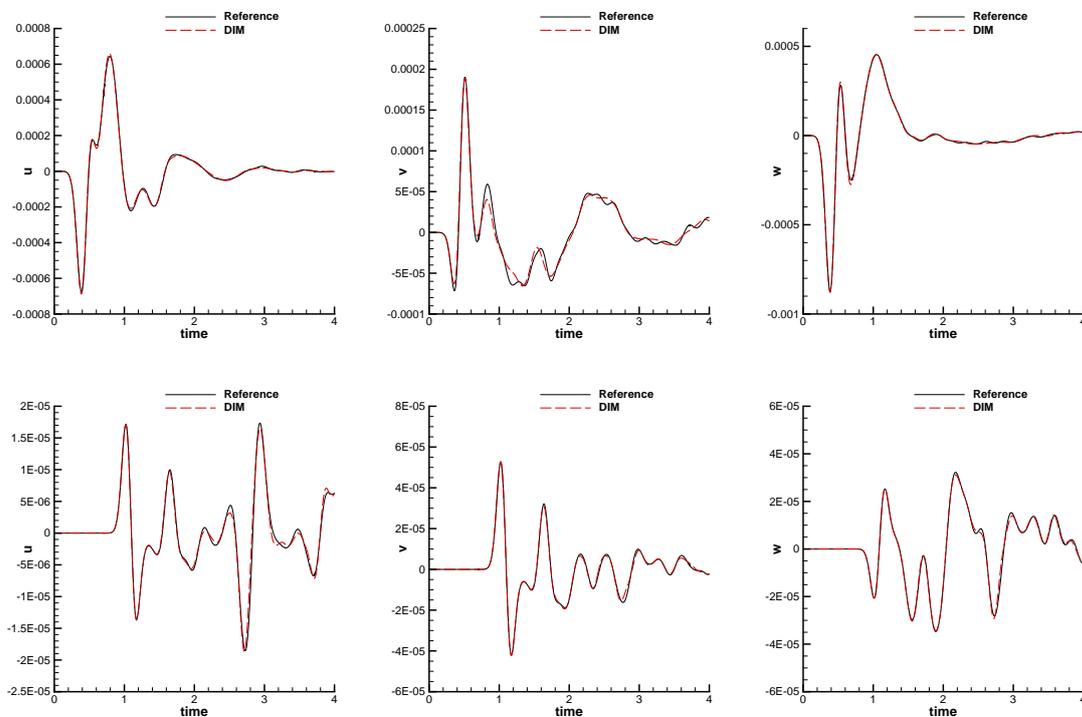

Figure 3. Seismogram recordings in two observation points obtained with the diffuse interface approach on adaptive Cartesian meshes [118] and with a reference solution obtained with high order ADER-DG schemes on boundary-fitted unstructured meshes [119].

3.3. The Unified Godunov-Peshkov-Romenski Model of Continuum Mechanics (GPR)

A major achievement of ExaHyPE was the first successful numerical solution of the unified first order symmetric hyperbolic and thermodynamically compatible Godunov–Peshkov–Romenski (GPR) model of continuum mechanics, see References [17,18]. The GPR model is based on the seminal papers by Godunov and Romenski [14,15,120] on inviscid symmetric hyperbolic systems. The dissipative mechanisms, which allow to model both plastic solids as well as viscous fluids within one single set of equations were added later in the groundbreaking work of Peshkov and Romenski in

Reference [16]. The GPR model is briefly outlined below, while for all details the interested reader is referred to References [16–18]. The governing equations read

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_k}\left(\rho u_k\right) = 0, \tag{24}$$

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial}{\partial x_k}\left(\partial \rho u_i u_k + p\delta_{ik} - \sigma_{ik}\right) = 0, \tag{25}$$

$$\frac{\partial A_{ik}}{\partial t} + \frac{\partial\left(A_{im}u_m\right)}{\partial x_k} + u_j\left(\frac{\partial A_{ik}}{\partial x_j} - \frac{\partial A_{ij}}{\partial x_k}\right) = -\frac{\psi_{ik}}{\theta_1(\tau_1)}, \tag{26}$$

$$\frac{\partial \rho J_i}{\partial t} + \frac{\partial}{\partial x_k}\left(\rho J_i u_k + T\delta_{ik}\right) = -\frac{1}{\theta_2(\tau_2)}\rho H_i, \tag{27}$$

$$\frac{\partial \rho E}{\partial t} + \frac{\partial}{\partial x_k}\left(u_k \rho E + u_i\left(p\delta_{ik} - \sigma_{ik}\right) + q_k\right) = 0. \tag{28}$$

Furthermore, the system is also endowed with an entropy inequality, see Reference [17]. Here, ρ is the mass density, $[u_i] = \mathbf{v} = (u, v, w)$ is the velocity vector, p is a non-equilibrium pressure, $[A_{ik}] = \mathbf{A}$ is the distorsion field, $[J_i] = \mathbf{J}$ is the thermal impulse vector, T is the temperature and ρE is the total energy density that is defined according to Reference [17] as

$$\rho E = \rho e + \frac{1}{2}\rho \mathbf{v}^2 + \frac{1}{4}\rho c_s^2 \operatorname{tr}\left((\operatorname{dev}\mathbf{G})^T(\operatorname{dev}\mathbf{G})\right) + \frac{1}{2}\rho \alpha^2 \mathbf{J}^2 \tag{29}$$

in terms of the specific internal energy $e = e(p, \rho)$ given by the usual equation of state (EOS), the kinetic energy, the energy stored in the medium due to deformations and in the thermal impulse. Furthermore, $\mathbf{G} = \mathbf{A}^T \mathbf{A}$ is a metric tensor induced by the distorsion field $\mathbf{A}$, which allows to measure distances and thus deformations in the medium, c_s is the shear sound speed and α is a heat wave propagation speed; the symbol $\operatorname{dev}\mathbf{G} = \mathbf{G} - \frac{1}{3}\operatorname{tr}\mathbf{G}$ indicates the trace-free part of the metric tensor $\mathbf{G}$. From the definition of the total energy Equation (29) and the relations $H_i = E_{J_i}$, $\psi_{ik} = E_{A_{ij}}$, $\sigma_{ik} = -\rho A_{mi} E_{A_{mk}}$, $T = E_S$ and $q_k = E_S E_{J_k}$ the shear stress tensor and the heat flux read $\sigma = -\rho c_s^2 \mathbf{G}\operatorname{dev}\mathbf{G}$ and $\mathbf{q} = \alpha^2 T \mathbf{J}$. It can furthermore be shown via formal asymptotic expansion [17] that via an appropriate choice of θ_1 and θ_2 in the stiff relaxation limit $\tau_1 \to 0$ and $\tau_2 \to 0$, the stress tensor and the heat flux tend to those of the compressible Navier-Stokes equations

$$\sigma \to \mu\left(\nabla \mathbf{v} + \nabla \mathbf{v}^T - \frac{2}{3}\left(\nabla \cdot \mathbf{v}\right)\mathbf{I}\right) \qquad \text{and} \qquad \mathbf{q} \to -\lambda \nabla T, \tag{30}$$

with transport coefficients $\mu = \mu(\tau_1, c_s)$ and $\lambda = \lambda(\tau_2, \alpha)$ related to the relaxation times τ_1 and τ_2 and to the propagation speeds c_s and α, respectively. For a complete derivation, see References [17,18]. In the opposite limit $\tau_1 \to \infty$ the model describes an ideal elastic solid with large deformations. This means that elastic solids as well as viscous fluids can be described with the aid of the same mathematical model. At this point we stress that numerically we always solve the unified *first order hyperbolic* PDE system in Equations (24)–(28), even in the stiff relaxation limit in Equation (30), when the compressible Navier-Stokes-Fourier system is retrieved asymptotically. We emphasize that we never need to discretize any parabolic terms, since the hyperbolic system in Equations (24)–(28) with algebraic relaxation source terms fits perfectly into the framework of Equation (1).

In the Figure 4 we show numerical results obtained in Reference [17] for a viscous heat conducting shock wave and the comparison with the exact solution of the compressible Navier-Stokes equations.
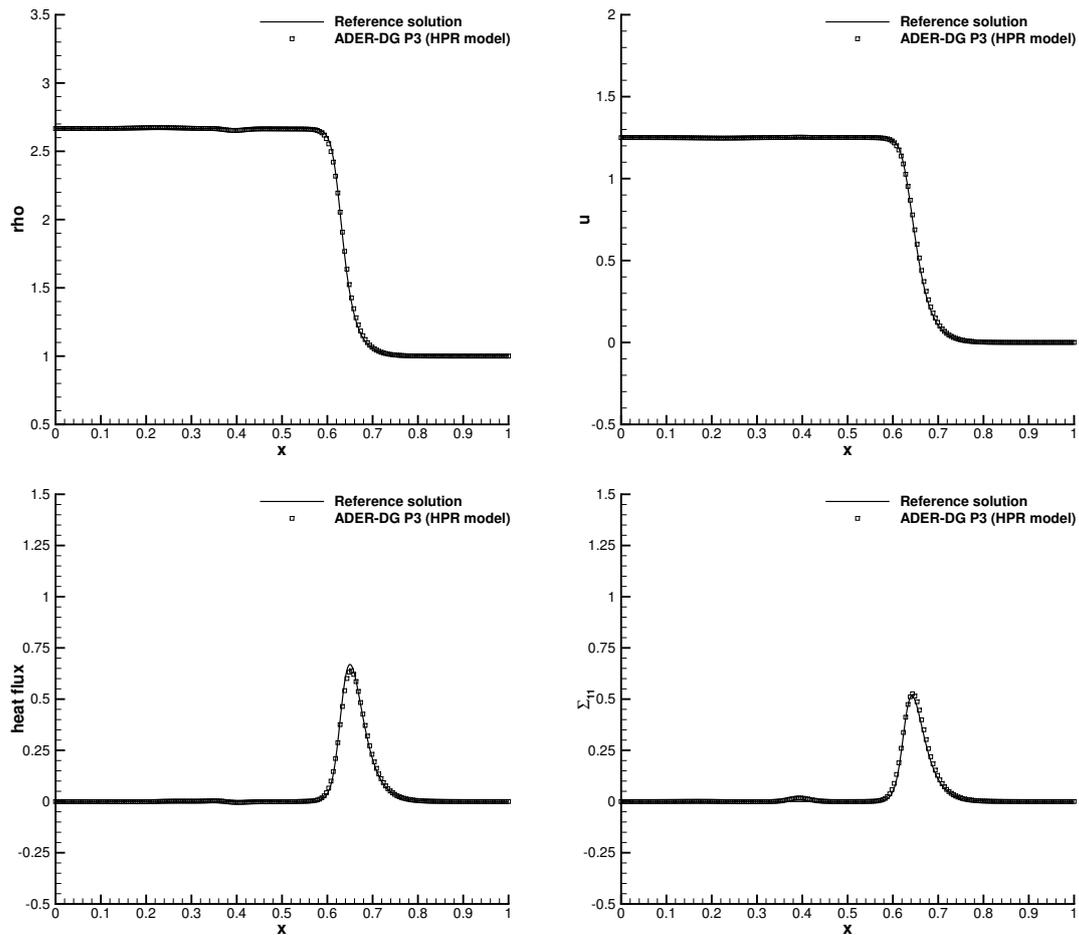
Figure 4. Viscous heat conducting shock. Comparison of the exact solution of the compressible Navier-Stokes equations with the numerical solution of the GPR model based on ADER-DG *P*3 schemes. Density profile (**top left**), velocity profile (**top right**), heat flux (**bottom left**) and stress σ_{11} (**bottom right**).

3.4. The Equations of Ideal General Relativistic Magnetohydrodynamics (GRMHD)

A very challenging PDE system is given by the equations of ideal general relativistic magnetohydrodynamics (GRMHD). The governing PDE are a result of the Einstein field equations and can be written in compact covariant notation as follows:

$$\nabla_\mu T^{\mu\nu} = 0, \qquad \text{and} \qquad \nabla_\mu {}^* F^{\mu\nu} = 0 \qquad \text{and} \qquad \nabla_\mu(\rho u^\mu) = 0, \tag{31}$$

where ∇_μ is the usual covariant derivative operator, $T^{\mu\nu}$ is the energy-momentum tensor, ${}^* F^{\mu\nu}$ is the Faraday tensor and u^μ is the four-velocity. The compact equations above can be expanded into a so-called 3+1 formalism, which can be cast into the form of Equation (1), see References [57,121,122] for more details. The final evolution system involves nine field variables plus the 10 quantities of the background space–time, which is supposed to be stationary here. A numerical convergence study for the large amplitude Alfvén wave test problem described in Reference [122] solved in the domain $\Omega = [0, 2\pi]^3$ up to $t = 1$ and carried out with high order ADER-DG schemes in Reference [57] is reported in the Table 2 below, where we also show a direct comparison with high order Runge-Kutta discontinuous Galerkin schemes. We observe that the ADER-DG schemes are competitive with RKDG methods, even for this very complex system of hyperbolic PDE. The results reported in Table 2 refer to the non-vectorized version of the code. Further significant performance improvements are expected

from a carefully vectorized implementation of the GRMHD equations, in particular concerning the vectorization of the cumbersome conversion of the vector of conservative variables to the vector of primitive variables, i.e., the function $\mathbf{V} = \mathbf{V}(\mathbf{Q})$. For the GRMHD system $\mathbf{V}$ cannot be computed analytically in terms of $\mathbf{Q}$, but requires the iterative solution of one nonlinear scalar algebraic equation together with the computation of the roots of a third order polynomial, see Reference [122] for details. In our vectorized implementation of the PDE, we have therefore in particular vectorized the primitive variable recovery via a direct implementation in AVX intrinsics. We have furthermore made use of careful auto-vectorization via the compiler for the evaluation of the physical flux function and for the non-conservative product. Thanks to this vectorization effort, on one single CPU core of an Intel i9-7900X Skylake test workstation with 3.3 GHz nominal clock frequency and using AVX 512 the CPU time necessary for a single degree of freedom update (TDU) for a fourth order ADER-DG scheme ($N = 3$) could be reduced to TDU = 2.3 μs for the GRMHD equations in three space dimensions.

Table 2. Accuracy and cost comparison between ADER-DG and RKDG schemes of different orders for the GRMHD equations in three space dimensions. The errors refer to the variable B_y. The table also contains total wall clock times (WCT) measured in seconds using 512 MPI ranks of the SuperMUC phase 1 system at the LRZ in Garching, Germany.

N_x	L_2 Error	L_2 Order	WCT [s]	N_x	L_2 Error	L_2 Order	WCT [s]
	ADER-DG ($N = 3$)				RKDG ($N = 3$)		
8	7.6396×10^{-4}		0.093	8	8.0909×10^{-4}		0.107
16	1.7575×10^{-5}	5.44	1.371	16	2.2921×10^{-5}	5.14	1.394
24	6.7968×10^{-6}	2.34	6.854	24	7.3453×10^{-6}	2.81	6.894
32	1.0537×10^{-6}	6.48	21.642	32	1.3793×10^{-6}	5.81	21.116
	ADER-DG ($N = 4$)				RKDG ($N = 4$)		
8	6.6955×10^{-5}		0.363	8	6.8104×10^{-5}		0.456
16	2.2712×10^{-6}	4.88	5.696	16	2.3475×10^{-6}	4.86	6.666
24	3.3023×10^{-7}	4.76	28.036	24	3.3731×10^{-7}	4.78	29.186
32	7.4728×10^{-8}	5.17	89.271	32	7.7084×10^{-8}	5.13	87.115
	ADER-DG ($N = 5$)				RKDG ($N = 5$)		
8	5.2967×10^{-7}		1.090	8	5.7398×10^{-7}		1.219
16	7.4886×10^{-9}	6.14	16.710	16	8.1461×10^{-9}	6.14	17.310
24	7.1879×10^{-10}	5.78	84.425	24	7.7634×10^{-10}	5.80	83.777
32	1.2738×10^{-10}	6.01	263.021	32	1.3924×10^{-10}	5.97	260.859

As second test problem we present the results obtained for the Orszag-Tang vortex system in flat Minkowski spacetime, where the GRMHD equations reduce to the special relativistic MHD equations. The initial condition is given by

$$\left(\rho, u, v, w, p, B_x, B_y, B_z \right) = \left(1, -\frac{3}{4\sqrt{2}} \sin y, \frac{3}{4\sqrt{2}} \sin x, 0, 1, -\sin y, \sin 2x, 0 \right),$$

and we set the adiabatic index to $\Gamma = 4/3$. The computational domain is $\Omega = [0, 2\pi]^2$ and is discretized with a dynamically adaptive AMR grid. For this test we chose the P_5 version of the ADER-DG scheme with FV subcell limiter and the rest mass density as indicator function for AMR, i.e., $\varphi(\mathbf{Q}) = \rho$. Figure 5 shows 1D cuts through the numerical solution at time $t = 2$ and at $y = 0.01$, while Figure 6 shows the numerical results for the AMR-grid with limiter-status map (blue cells are unlimited, while limited cells are highlighted in red), together with Schlieren images for the rest-mass density at time $t = 2$. The same simulation has been repeated with different refinement estimator functions χ that tell the AMR algorithm where and when to refine and to coarsen the mesh: (i) A simple first order derivative estimator χ_1 based on discrete gradients of the indicator function $\varphi(\mathbf{Q})$, (ii) the classical second order derivative estimator χ_2 based on Reference [123], (iii) a novel estimator χ_3 based on the action of the *a posteriori* subcell finite volume limiter, i.e., the mesh is refined where the limiter is active (iv) a multi-resolution estimator χ_4 based on the difference in L_∞ norm of the discrete solution

on two different refinement levels ℓ and $\ell - 1$. The reference solution is obtained on a uniform fine grid corresponding to the finest refinement level, i.e., a uniform composed of 270×270 elements. The results shown in Figure 6 clearly show that the numerical results obtained by means of different refinement estimator functions are comparable with each other and thus the proposed AMR algorithm is robust with respect to the particular choice of the mesh.
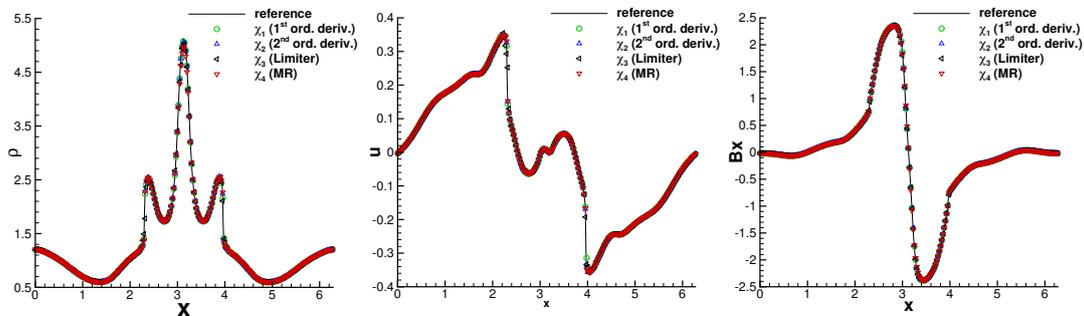

Figure 5. Results for the GRMHD Orszag-Tang vortex problem in flat space–time (SRMHD) at $t = 2$ obtained with ADER-DG-$\mathbb{P}_5$ schemes supplemented with *a posteriori* subcell finite volume limiter and using different refinement estimator functions χ. A set of 1D cuts taken at $y = 10^{-2}$ are shown. From (**left**) to (**right**): the rest-mass density, the velocity u and the magnetic field component B_x. One can note an excellent agreement between the reference solution and the ones obtained on different AMR grids.
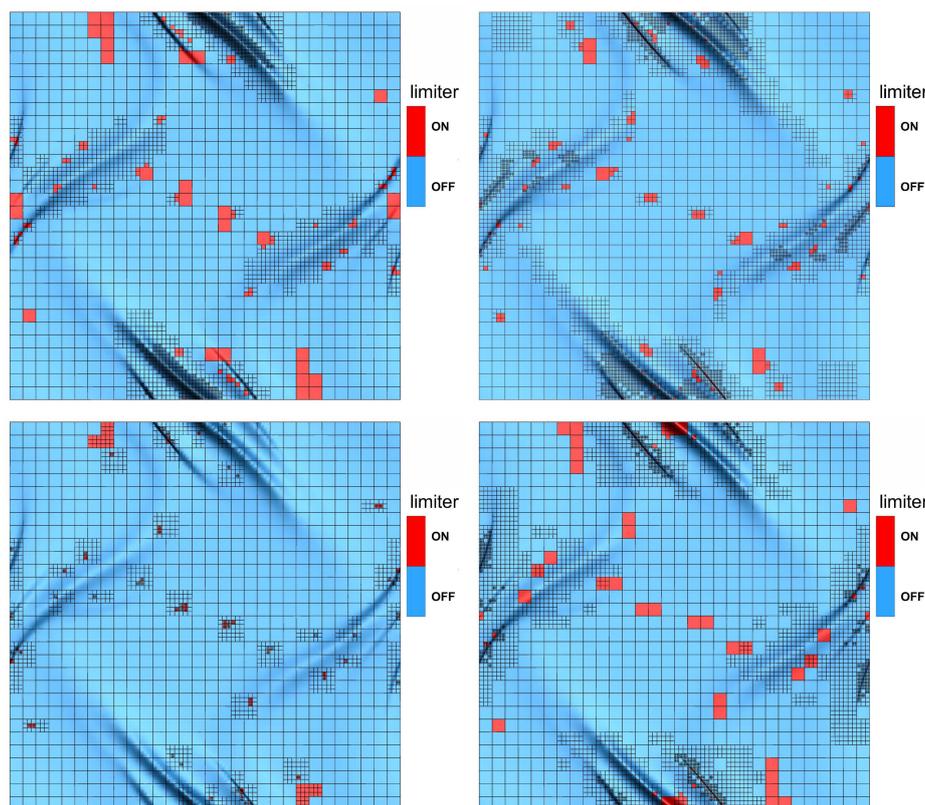
Figure 6. Results for the GRMHD Orszag-Tang vortex problem in flat space–time (SRMHD) at $t = 2$ obtained with ADER-DG-$\mathbb{P}_5$ schemes, supplemented with *a posteriori* subcell finite volume limiter and using different refinement estimator functions χ. (i) first order-derivative estimator χ_1 (**top left**); (ii) second-order derivative estimator χ_2 (**top right**); (iii) a new limiter-based estimator χ_3 (row two, **left**) and (iv) a new multi-resolution estimator χ_4 based on the difference between the discrete solution on two adjacent refinement levels (row two, **right**).

As a last test case we simulate a stationary neutron star in three space dimensions using the Cowling approximation, i.e., assuming a fixed *static* background spacetime. The initial data for the matter and the spacetime are both compatible with the Einstein field equations and are given by the solution of the Tolman–Oppenheimer–Volkoff (TOV) equations, which constitute a nonlinear ODE system in the radial coordinate that can be numerically solved up to any precision at the aid of a fourth order Runge-Kutta scheme using a very fine grid. We setup a stable nonrotating TOV star without magnetic field and with central rest mass density $\rho(\mathbf{0}, 0) = 1.28 \times 10^{-3}$ and adiabatic exponent $\Gamma = 2$ in a computational domain $\Omega = [-10, +10]^3$ discretized with a fourth order ADER-DG scheme ($N = 3$) using 32^3 elements, which corresponds to 128^3 spatial degrees of freedom. The pressure in the atmosphere outside the compact object is set to $p_{atm} = 10^{-13}$. We run the simulation until a final time of $t = 1000$ and measure the L_∞ error norms of the rest mass density and the pressure against the exact solution, which is given by the initial condition. The error measured at $t = 1000$ for the rest mass density is $L_\infty(\rho) = 1.553778 \times 10^{-5}$ while the error for the pressure is $L_\infty(p) = 1.605334 \times 10^{-7}$. The simulation was carried out with the vectorized version of the code on 512 CPU cores of the SuperMUC phase 2 system (based on AVX2) and required only 3010 s of wallclock time. The same simulation with the established finite difference GR code WhiskyTHC [124] required 8991 s of wall clock time on the same machine with the same spatial mesh resolution and the same number of CPU cores. The time series of the relative error of the central rest mass density in the origin of the domain is plotted in the left panel of Figure 7. At the final time $t = 1000$, the relative error of the central rest mass density is still below 0.1%. In the right panel of Figure 7 we show the contour surfaces of the pressure at the final time $t = 1000$. In Figure 8 we show a 1D cut along the x axis, comparing the numerical solution at time $t = 1000$ with the exact one. We note that the numerical scheme is very accurate, but it is *not well-balanced* for the GRMHD equations, i.e., the method *cannot* preserve the stationary equilibrium solution of the TOV equations *exactly* at the discrete level. Therefore, further work along the lines of research reported recently in Reference [86] for the Euler equations with Newtonian gravity are needed, extending the framework of well-balanced methods [79,80,125] also to general relativity. Finally, in Figure 9 we compare the exact and the numerical solution at time $t = 1000$ in the x–y plane.
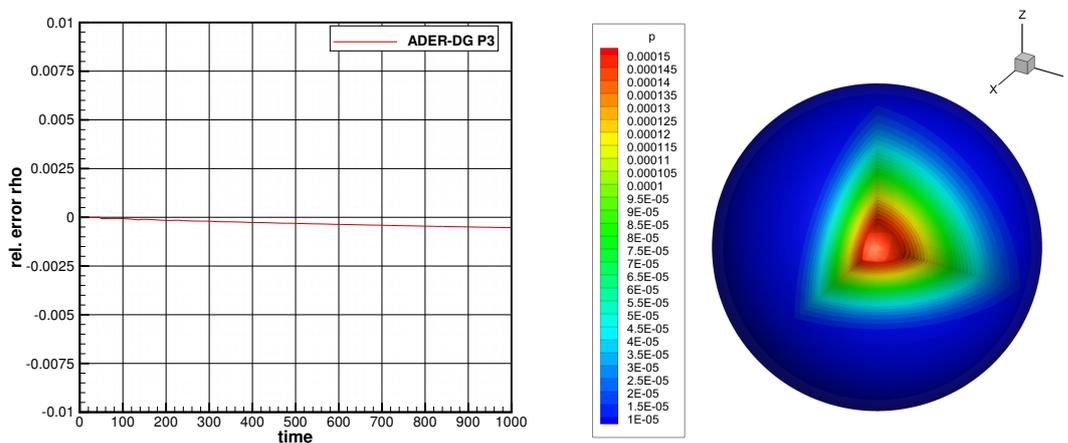

Figure 7. Computational results for a stable 3D neutron star. Time series of the relative error of the central rest mass density $(\rho(\mathbf{0}, t) - \rho(\mathbf{0}, 0))/\rho(\mathbf{0}, 0)$ (**left**) and 3D view of of the pressure contour surfaces at time $t = 1000$ (**right**).
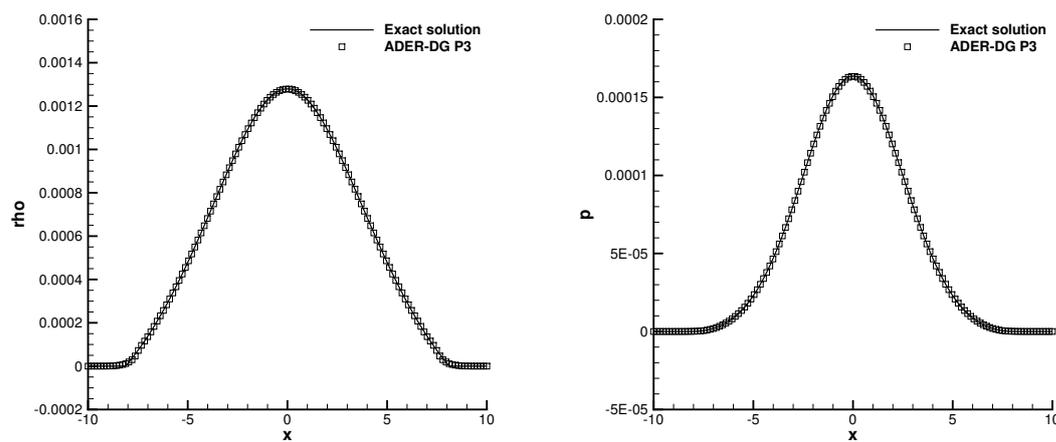
Figure 8. Computational results for a stable 3D neutron star. Comparison of the numerical solution with the exact one at time $t = 1000$ on a 1D cut along the x-axis for the rest mass density (**left**) and the pressure (**right**).
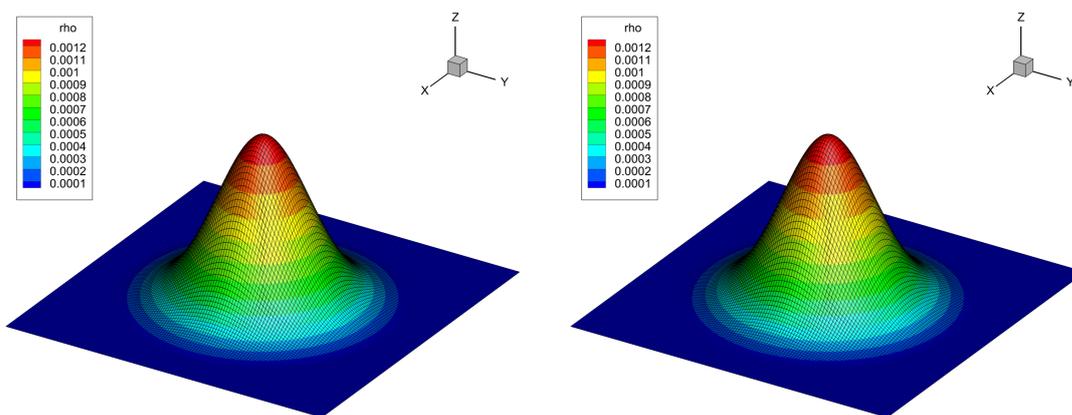

Figure 9. Computational results for a stable 3D neutron star. Cut through the x–y plane with pressure on the z axis and rest mass density contour colors. Exact solution (**left**) and numerical solution at time $t = 1000$ (**right**).

3.5. A Strongly Hyperbolic First Order Reduction of the CCZ4 Formulation of the Einstein Field Equations (FO-CCZ4)

The last PDE system under consideration here are the Einstein field equations that describe the evolution of dynamic spacetimes. Here we consider the so-called CCZ4 formulation [126], which is based on the Z4 formalism that takes into account the involutions (stationary differential constraints) inherent in the Einstein equations via an augmented system similar to the generalized Lagrangian multiplier (GLM) approach of Dedner et al. [127] that takes care of the stationary divergence-free constraint of the magnetic field in the MHD equations. In compact covariant notation the undamped Z4 Einstein equations in vacuum, which can be derived from the Einstein–Hilbert action integral associated with the Z4 Lagrangian $\mathcal{L} = g^{\mu\nu} \left(R_{\mu\nu} + 2\nabla_\mu Z_\nu \right)$, read

$$R_{\mu\nu} + \nabla_{(\mu} Z_{\nu)} = 0, \tag{32}$$

where $g^{\mu\nu}$ is the 4-metric of the spacetime, $R_{\mu\nu}$ is the 4-Ricci tensor and the 4-vector Z_ν accounts for the stationary constraints of the Einstein equations, as already mentioned before. After introducing the usual 3+1 ADM split of the 4-metric as

$$ds^2 = -\alpha^2 dt^2 + \gamma_{ij}\left(dx^i + \beta^i dt\right)\left(dx^j + \beta^j dt\right),\tag{33}$$

the equations can be cast into a time-dependent system of 25 partial differential equations that involve first order derivatives in time and both first and second order derivatives in space, see Reference [126]. Nevertheless, the system is *not* dissipative, but a rather unusual formulation of a wave equation, see Reference [128]. In the expression above, α denotes the so-called lapse, β^i is the spatial shift vector and γ_{ij} is the spatial metric. In the original form presented in Reference [126], the PDE system does *not* fit into the formalism given by Equation (1). After the introduction of 33 auxiliary variables, which are the spatial gradients of some of the 25 primary evolution quantities, it is possible to derive a first order reduction of the system that contains a total of 58 evolution quantities. However, a naive procedure of converting the original second order evolution system into a first order system leads only to a *weakly hyperbolic* formulation, which is not suitable for numerical simulations since the initial value problem is not well posed in this case. Only after adding suitable first and second order ordering constraints, which arise from the definition of the auxiliary variables, it is possible to obtain a provably strongly hyperbolic and thus well-posed evolution system, denoted by FO-CCZ4 in the following. For all details of the derivation, the strong hyperbolicity proof and numerical results achieved with high order ADER-DG schemes, the reader is referred to Reference [129]. In order to give an idea about the complexity of the Einstein field equations, it should be mentioned that one single evaluation of the FO-CCZ4 system requires about 20,000 floating point operations! In order to obtain still a computationally efficient implementation, the entire PDE system has been carefully vectorized using blocks of the size `VECTORLENGTH`, so that in the end a level of 99.9% of vectorization of the code has been reached. Using a fourth order ADER-DG scheme ($N = 3$) the time per degree of freedom update (TDU) metric per core on a modern workstation with Intel i9-7900X CPU that supports the novel AVX 512 instructions is TDU = 4.7 μs.

4. Strong MPI Scaling Study for the FO-CCZ4 System

A major focus of this paper is the efficient implementation of ADER-DG schemes for high performance computing (HPC) on massively parallel distributed memory supercomputers. For this purpose, we have very recently carried out a systematic study of the strong MPI scaling efficiency of our new high order fully-discrete one-step ADER-DG schemes on the Hazel Hen supercomputer of the HLRS center in Stuttgart, Germany, using from 720 up to 180,000 CPU cores. We have furthermore carried out a systematic comparison with conventional Runge-Kutta DG schemes using the SuperMUC phase 1 system of the LRZ center in Munich, Germany.

As already discussed before, the particular feature of ADER-DG schemes compared to traditional Runge-Kutta DG schemes (RKDG) is that they are intrinsically *communication-avoiding* and *cache-blocking*, which makes them particularly well suited for modern massively parallel distributed memory supercomputers. As governing PDE system for the strong scaling test the novel first-order reduction of the CCZ4 formulation of the 3+1 Einstein field equations has been been adopted [129]. We recall that FO-CCZ4 is a very large nonlinear hyperbolic PDE system that contains 58 evolution quantities.

The first strong scaling study on the SuperMUC phase 1 system uses 64 to 64,000 CPU cores. The test problem was the gauge wave problem [129] setup on the 3D domain $\Omega = [-0.5, 0.5]^3$. For the test we have compared a fourth order ADER-DG scheme ($N = 3$) with a fourth order accurate RKDG scheme on a uniform Cartesian grid composed of 120^3 elements. It has to be stressed, that when using 64,000 CPU cores for this setup each CPU has to update only $3^3 = 27$ elements. The wall clock time as a function of the used number of CPU cores (nCPU) and the obtained parallel efficiency with respect

to an ideal linear scaling are reported in the left panel of Figure 10. We find that ADER-DG schemes provide a better parallel efficiency than RKDG schemes, as expected.

The second strong scaling study has been performed on the Hazel-Hen supercomputer, using 720 to 180,000 CPU cores. Again we have used a fourth order accurate ADER-DG scheme ($N = 3$), this time using a uniform grid of $200 \times 180 \times 180$ elements, solving again the 3D gauge wave benchmark problem detailed in Reference [129]. The measured wall-clock-times (WCT) as a function of the employed number of CPU cores, as well as the corresponding parallel scaling-efficiency are shown in Figure 10. The results depicted in Figure 10 clearly show that our new ADER-DG schemes **scale very well** up to 90,000 CPU cores with a parallel efficiency **greater than 95%**, and up to 180,000 cores with a parallel efficiency that is still greater than **93%**. Furthermore, the code was instrumented with manual FLOP counters in order to measure the floating point performance quantitatively. The full machine run on **180,000 CPU cores** of Hazel Hen took place on 7 May 2018. During the run, each core has provided an average performance of 8.2 GFLOPS, leading to a total of **1.476 PFLOPS** of sustained performance. To our knowledge, this was the largest test run ever carried out with high order ADER-DG schemes for nonlinear hyperbolic systems of partial differential equations. For large runs with sustained petascale performance of ADER-DG schemes for linear hyperbolic PDE systems on unstructured tetrahedral meshes, see Reference [105].
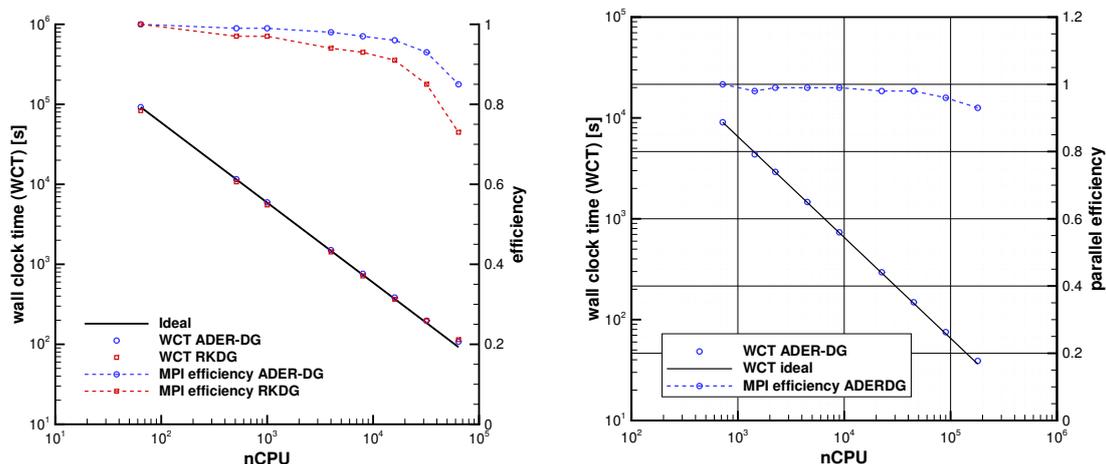

Figure 10. Strong MPI scaling study of ADER-DG schemes for the novel FO-CCZ4 formulation of the Einstein field equations recently proposed in Reference [129]. (**Left**) comparison of ADER-DG schemes with conventional Runge-Kutta DG schemes from 64 to 64,000 CPU cores on the SuperMUC phase 1 system of the LRZ supercomputing center (Garching, Germany). (**Right**) strong scaling study from 720 to 180,000 CPU cores, including a full machine run on the Hazel Hen supercomputer of HLRS (Stuttgart, Germany) with ADER-DG schemes (**right**). Even on the full machine we observe still more than 90% of parallel efficiency.

5. Conclusions

In this paper we have presented an efficient implementation of high order ADER-DG schemes on modern massively parallel supercomputers using the ExaHyPE engine. The key ingredients are the communication-avoiding and cache-blocking properties of ADER-DG, together with an efficient vectorization of the high level user functions that provide the evaluation of the physical fluxes $\mathbf{F}(\mathbf{Q})$, of the non-conservative products $\mathbf{B}(\mathbf{Q}) \cdot \nabla \mathbf{Q}$ and of the algebraic source terms $\mathbf{S}(\mathbf{Q})$. The engine is highly versatile and flexible and allows to solve a very broad spectrum of different hyperbolic PDE systems in a very efficient and highly scalable manner. In order to support this claim, we have provided a rather large set of different numerical examples solved with ADER-DG schemes. To show the excellent parallel scalability of the ADER-DG method, we have provided strong scaling results

on 64 to 64,000 CPU cores including a detailed and quantitative comparison with RKDG schemes. We have furthermore shown strong scaling results of the vectorized ADER-DG implementation for the FO-CCZ4 formulation of the Einstein field equations using 720 to 180,000 CPU cores of the Hazel Hen supercomputer at the HLRS in Stuttgart, Germany, where a sustained performance of more than one petaflop has been reached.

Future research in ExaHyPE will concern an extension of the GPR model to full general relativity, able to describe nonlinear elastic and plastic solids as well as viscous and ideal fluids in one single governing PDE system. We furthermore plan an implementation of the FO-CCZ4 system [129] directly based on AVX intrinsics, in order to further improve the performance of the scheme and to reduce computational time. The final aim of our developments are the simulation of ongoing nonlinear dynamic rupture processes during earthquakes, as well as the inspiral and merger of binary neutron star systems and the associated generation of gravitational waves. Although both problems seem to be totally different and unrelated, it is indeed possible to write the mathematical formulation of both applications under the same form of a hyperbolic system given by Equation (1) and thus to solve both problems within the same computer software.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Riemann, B. *Über die Fortpflanzung ebener Luftwellen von Endlicher Schwingungsweite*; Göttinger Nachrichten: Göttingen, Germany, 1859; Volume 19.
2. Riemann, B. Über die Fortpflanzung ebener Luftwellen von endlicher Schwingungsweite. *Abhandlungen der Königlichen Gesellschaft der Wissenschaften zu Göttingen* **1860**, *8*, 43–65.
3. Noether, E. Invariante Variationsprobleme. In *Nachrichten der königlichen Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-Physikalische Klasse*; Weidmannsche Buchhandlung: Berlin, Germany, 1918; pp. 235–257.
4. Courant, R.; Friedrichs, K.; Lewy, H. Über die partiellen Differenzengleichungen der mathematischen Physik. *Math. Annal.* **1928**, *100*, 32–74. [CrossRef]
5. Courant, R.; Isaacson, E.; Rees, M. On the solution of nonlinear hyperbolic differential equations by finite differences. *Commun. Pure Appl. Math.* **1952**, *5*, 243–255. [CrossRef]
6. Courant, R.; Hilbert, D. *Methods of Mathematical Physics*; John Wiley and Sons, Inc.: New York, NY, USA, 1962.
7. Courant, R.; Friedrichs, K.O. *Supersonic Flows and Shock Waves*; Springer: Berlin, Germany, 1976.
8. Friedrichs, K. Symmetric positive linear differential equations. *Commun. Pure Appl. Math.* **1958**, *11*, 333–418. [CrossRef]
9. Godunov, S. An interesting class of quasilinear systems. *Dokl. Akad. Nauk SSSR* **1961**, *139*, 521–523.
10. Friedrichs, K.; Lax, P. Systems of conservation equations with a convex extension. *Proc. Natl. Acad. Sci. USA* **1971**, *68*, 1686–1688. [CrossRef] [PubMed]
11. Godunov, S.; Romenski, E. Thermodynamics, conservation laws, and symmetric forms of differential equations in mechanics of continuous media. In *Computational Fluid Dynamics Review 95*; John Wiley: New York, NY, USA, 1995; pp. 19–31.

12. Godunov, S.; Romenski, E. *Elements of Continuum Mechanics and Conservation Laws*; Kluwer Academic/Plenum Publishers: New York, NY, USA, 2003.

13. Godunov, S. Symmetric form of the magnetohydrodynamic equation. *Numer. Methods Mech. Contin. Med.* **1972**, *3*, 26–34.

14. Godunov, S.; Romenski, E. Nonstationary equations of the nonlinear theory of elasticity in Euler coordinates. *J. Appl. Mech. Tech. Phys.* **1972**, *13*, 868–885. [CrossRef]

15. Romenski, E. Hyperbolic systems of thermodynamically compatible conservation laws in continuum mechanics. *Math. Comput. Model.* **1998**, *28*, 115–130. [CrossRef]

16. Peshkov, I.; Romenski, E. A hyperbolic model for viscous Newtonian flows. *Contin. Mech. Thermodyn.* **2016**, *28*, 85–104. [CrossRef]

17. Dumbser, M.; Peshkov, I.; Romenski, E.; Zanotti, O. High order ADER schemes for a unified first order hyperbolic formulation of continuum mechanics: Viscous heat-conducting fluids and elastic solids. *J. Comput. Phys.* **2016**, *314*, 824–862. [CrossRef]

18. Dumbser, M.; Peshkov, I.; Romenski, E.; Zanotti, O. High order ADER schemes for a unified first order hyperbolic formulation of Newtonian continuum mechanics coupled with electro-dynamics. *J. Comput. Phys.* **2017**, *348*, 298–342. [CrossRef]

19. Boscheri, W.; Dumbser, M.; Loubère, R. Cell centered direct Arbitrary-Lagrangian-Eulerian ADER-WENO finite volume schemes for nonlinear hyperelasticity. *Comput. Fluids* **2016**, *134-135*, 111–129. [CrossRef]

20. Neumann, J.V.; Richtmyer, R.D. A method for the numerical calculation of hydrodynamic shocks. *J. Appl. Phys.* **1950**, *21*, 232–237. [CrossRef]

21. Godunov, S.K. A finite difference Method for the Computation of discontinuous solutions of the equations of fluid dynamics. *Math. USSR Sbornik* **1959**, *47*, 357–393.

22. Kolgan, V.P. Application of the minimum-derivative principle in the construction of finite-difference schemes for numerical analysis of discontinuous solutions in gas dynamics. *Trans. Central Aerohydrodyn. Inst.* **1972**, *3*, 68–77. (In Russian)

23. Van Leer, B. Towards the Ultimate Conservative Difference Scheme II: Monotonicity and conservation combined in a second order scheme. *J. Comput. Phys.* **1974**, *14*, 361–370. [CrossRef]

24. van Leer, B. Towards the Ultimate Conservative Difference Scheme V: A second Order sequel to Godunov's Method. *J. Comput. Phys.* **1979**, *32*, 101–136. [CrossRef]

25. Harten, A.; Osher, S. Uniformly high–order accurate nonoscillatory schemes I. *SIAM J. Num. Anal.* **1987**, *24*, 279–309. [CrossRef]

26. Jiang, G.; Shu, C. Efficient Implementation of Weighted ENO Schemes. *J. Comput. Phys.* **1996**, *126*, 202–228. [CrossRef]

27. Cockburn, B.; Shu, C.W. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws II: General framework. *Math. Comput.* **1989**, *52*, 411–435.

28. Cockburn, B.; Lin, S.Y.; Shu, C. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: One dimensional systems. *J. Comput. Phys.* **1989**, *84*, 90–113. [CrossRef]

29. Cockburn, B.; Hou, S.; Shu, C.W. The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: The multidimensional case. *Math. Comput.* **1990**, *54*, 545–581.

30. Cockburn, B.; Shu, C.W. The Runge-Kutta discontinuous Galerkin method for conservation laws V: Multidimensional systems. *J. Comput. Phys.* **1998**, *141*, 199–224. [CrossRef]

31. Cockburn, B.; Shu, C.W. The Local Discontinuous Galerkin Method for Time-Dependent Convection Diffusion Systems. *SIAM J. Numer. Anal.* **1998**, *35*, 2440–2463. [CrossRef]

32. Cockburn, B.; Shu, C.W. Runge-Kutta Discontinuous Galerkin Methods for Convection-Dominated Problems. *J. Sci. Comput.* **2001**, *16*, 173–261. [CrossRef]

33. Shu, C. High order WENO and DG methods for time-dependent convection-dominated PDEs: A brief survey of several recent developments. *J. Comput. Phys.* **2016**, *316*, 598–613. [CrossRef]

34. Dumbser, M.; Balsara, D.; Toro, E.; Munz, C. A Unified Framework for the Construction of One-Step Finite–Volume and discontinuous Galerkin schemes. *J. Comput. Phys.* **2008**, *227*, 8209–8253. [CrossRef]

35. Dumbser, M.; Castro, M.; Parés, C.; Toro, E. ADER Schemes on Unstructured Meshes for Non–Conservative Hyperbolic Systems: Applications to Geophysical Flows. *Comput. Fluids* **2009**, *38*, 1731–1748. [CrossRef]

36. Dumbser, M.; Zanotti, O.; Loubère, R.; Diot, S. A posteriori subcell limiting of the discontinuous Galerkin finite element method for hyperbolic conservation laws. *J. Comput. Phys.* **2014**, *278*, 47–75. [CrossRef]

37. Zanotti, O.; Fambri, F.; Dumbser, M.; Hidalgo, A. Space-time adaptive ADER discontinuous Galerkin finite element schemes with a posteriori sub-cell finite volume limiting. *Comput. Fluids* **2015**, *118*, 204–224. [CrossRef]

38. Dumbser, M.; Loubère, R. A simple robust and accurate a posteriori sub-cell finite volume limiter for the discontinuous Galerkin method on unstructured meshes. *J. Comput. Phys.* **2016**, *319*, 163–199. [CrossRef]

39. Titarev, V.; Toro, E. ADER: Arbitrary High Order Godunov Approach. *J. Sci. Comput.* **2002**, *17*, 609–618. [CrossRef]

40. Toro, E.; Titarev, V. Solution of the generalized Riemann problem for advection-reaction equations. *Proc. R. Soc. Lond.* **2002**, *458*, 271–281. [CrossRef]

41. Titarev, V.; Toro, E. ADER schemes for three-dimensional nonlinear hyperbolic systems. *J. Comput. Phys.* **2005**, *204*, 715–736. [CrossRef]

42. Toro, E.F.; Titarev, V.A. Derivative Riemann solvers for systems of conservation laws and ADER methods. *J. Comput. Phys.* **2006**, *212*, 150–165. [CrossRef]

43. Bungartz, H.; Mehl, M.; Neckel, T.; Weinzierl, T. The PDE framework Peano applied to fluid dynamics: An efficient implementation of a parallel multiscale fluid dynamics solver on octree-like adaptive Cartesian grids. *Comput. Mech.* **2010**, *46*, 103–114. [CrossRef]

44. Weinzierl, T.; Mehl, M. Peano-A traversal and storage scheme for octree-like adaptive Cartesian multiscale grids. *SIAM J. Sci. Comput.* **2011**, *33*, 2732–2760. [CrossRef]

45. Khokhlov, A. Fully Threaded Tree Algorithms for Adaptive Refinement Fluid Dynamics Simulations. *J. Comput. Phys.* **1998**, *143*, 519–543. [CrossRef]

46. Baeza, A.; Mulet, P. Adaptive mesh refinement techniques for high-order shock capturing schemes for multi-dimensional hydrodynamic simulations. *Int. J. Numer. Methods Fluids* **2006**, *52*, 455–471. [CrossRef]

47. Colella, P.; Dorr, M.; Hittinger, J.; Martin, D.F.; McCorquodale, P. High-order finite-volume adaptive methods on locally rectangular grids. *J. Phys. Conf. Ser.* **2009**, *180*, 012010. [CrossRef]

48. Bürger, R.; Mulet, P.; Villada, L. Spectral WENO schemes with Adaptive Mesh Refinement for models of polydisperse sedimentation. *ZAMM J. Appl. Math. Mech. Z. Math. Mech.* **2013**, *93*, 373–386. [CrossRef]

49. Ivan, L.; Groth, C. High-order solution-adaptive central essentially non-oscillatory (CENO) method for viscous flows. *J. Comput. Phys.* **2014**, *257*, 830–862. [CrossRef]

50. Buchmüller, P.; Dreher, J.; Helzel, C. Finite volume WENO methods for hyperbolic conservation laws on Cartesian grids with adaptive mesh refinement. *Appl. Math. Comput.* **2016**, *272*, 460–478. [CrossRef]

51. Semplice, M.; Coco, A.; Russo, G. Adaptive Mesh Refinement for Hyperbolic Systems based on Third-Order Compact WENO Reconstruction. *J. Sci. Comput.* **2016**, *66*, 692–724. [CrossRef]

52. Shen, C.; Qiu, J.; Christlieb, A. Adaptive mesh refinement based on high order finite difference WENO scheme for multi-scale simulations. *J. Comput. Phys.* **2011**, *230*, 3780–3802. [CrossRef]

53. Dumbser, M.; Zanotti, O.; Hidalgo, A.; Balsara, D. ADER-WENO Finite Volume Schemes with Space-Time Adaptive Mesh Refinement. *J. Comput. Phys.* **2013**, *248*, 257–286. [CrossRef]

54. Dumbser, M.; Hidalgo, A.; Zanotti, O. High Order Space-Time Adaptive ADER-WENO Finite Volume Schemes for Non-Conservative Hyperbolic Systems. *Comput. Methods Appl. Mech. Eng.* **2014**, *268*, 359–387. [CrossRef]

55. Zanotti, O.; Fambri, F.; Dumbser, M. Solving the relativistic magnetohydrodynamics equations with ADER discontinuous Galerkin methods, a posteriori subcell limiting and adaptive mesh refinement. *Mon. Not. R. Astron. Soc.* **2015**, *452*, 3010–3029. [CrossRef]

56. Fambri, F.; Dumbser, M.; Zanotti, O. Space-time adaptive ADER-DG schemes for dissipative flows: Compressible Navier-Stokes and resistive MHD equations. *Comput. Phys. Commun.* **2017**, *220*, 297–318. [CrossRef]

57. Fambri, F.; Dumbser, M.; Köppel, S.; Rezzolla, L.; Zanotti, O. ADER discontinuous Galerkin schemes for general-relativistic ideal magnetohydrodynamics. *Mon. Not. R. Astron. Soc.* **2018**, *477*, 4543–4564. [CrossRef]

58. Berger, M.J.; Oliger, J. Adaptive Mesh Refinement for Hyperbolic Partial Differential Equations. *J. Comput. Phys.* **1984**, *53*, 484–512. [CrossRef]

59. Berger, M.J.; Jameson, A. Automatic adaptive grid refinement for the Euler equations. *AIAA J.* **1985**, *23*, 561–568. [CrossRef]

60. Berger, M.J.; Colella, P. Local adaptive mesh refinement for shock hydrodynamics. *J. Comput. Phys.* **1989**, *82*, 64–84. [CrossRef]

61. Leveque, R. Clawpack Software. Available online: http://depts.washington.edu/clawpack/ (accessed on 23 August 2018).

62. Berger, M.J.; LeVeque, R. Adaptive mesh refinement using wave-propagation algorithms for hyperbolic systems. *SIAM J. Numer. Anal.* **1998**, *35*, 2298–2316. [CrossRef]

63. Bell, J.; Berger, M.; Saltzman, J.; Welcome, M. Three-dimensional adaptive mesh refinement for hyperbolic conservation laws. *SIAM J. Sci. Comput.* **1994**, *15*, 127–138. [CrossRef]

64. Quirk, J. A parallel adaptive grid algorithm for computational shock hydrodynamics. *Appl. Numer. Math.* **1996**, *20*, 427–453. [CrossRef]

65. Coirier, W.; Powell, K. Solution-adaptive Cartesian cell approach for viscous and inviscid flows. *AIAA J.* **1996**, *34*, 938–945. [CrossRef]

66. Deiterding, R. A parallel adaptive method for simulating shock-induced combustion with detailed chemical kinetics in complex domains. *Comput. Struct.* **2009**, *87*, 769–783. [CrossRef]

67. Lopes, M.M.; Deiterding, R.; Gomes, A.F.; Mendes, O.; Domingues, M.O. An ideal compressible magnetohydrodynamic solver with parallel block-structured adaptive mesh refinement. *Comput. Fluids* **2018**, *173*, 293–298. [CrossRef]

68. Dezeeuw, D.; Powell, K.G. An Adaptively Refined Cartesian Mesh Solver for the Euler Equations. *J. Comput. Phys.* **1993**, *104*, 56–68. [CrossRef]

69. Balsara, D. Divergence-free adaptive mesh refinement for magnetohydrodynamics. *J. Comput. Phys.* **2001**, *174*, 614–648. [CrossRef]

70. Teyssier, R. Cosmological hydrodynamics with adaptive mesh refinement. A new high resolution code called RAMSES. *Astron. Astrophys.* **2002**, *385*, 337–364. [CrossRef]

71. Keppens, R.; Nool, M.; Tóth, G.; Goedbloed, J.P. Adaptive Mesh Refinement for conservative systems: Multi-dimensional efficiency evaluation. *Comput. Phys. Commun.* **2003**, *153*, 317–339. [CrossRef]

72. Ziegler, U. The NIRVANA code: Parallel computational MHD with adaptive mesh refinement. *Comput. Phys. Commun.* **2008**, *179*, 227–244. [CrossRef]

73. Mignone, A.; Zanni, C.; Tzeferacos, P.; van Straalen, B.; Colella, P.; Bodo, G. The PLUTO Code for Adaptive Mesh Computations in Astrophysical Fluid Dynamics. *Astrophys. J. Suppl. Ser.* **2012**, *198*, 7. [CrossRef]

74. Cunningham, A.; Frank, A.; Varnière, P.; Mitran, S.; Jones, T.W. Simulating Magnetohydrodynamical Flow with Constrained Transport and Adaptive Mesh Refinement: Algorithms and Tests of the AstroBEAR Code. *Astrophys. J. Suppl. Ser.* **2009**, *182*, 519. [CrossRef]

75. Keppens, R.; Meliani, Z.; van Marle, A.; Delmont, P.; Vlasis, A.; van der Holst, B. Parallel, grid-adaptive approaches for relativistic hydro and magnetohydrodynamics. *J. Comput. Phys.* **2012**, *231*, 718–744. [CrossRef]

76. Porth, O.; Olivares, H.; Mizuno, Y.; Younsi, Z.; Rezzolla, L.; Moscibrodzka, M.; Falcke, H.; Kramer, M. The black hole accretion code. *Comput. Astrophys. Cosmol.* **2017**, *4*, 1–42. [CrossRef]

77. Dubey, A.; Almgren, A.; Bell, J.; Berzins, M.; Brandt, S.; Bryan, G.; Colella, P.; Graves, D.; Lijewski, M.; Löffler, F.; et al. A survey of high level frameworks in block-structured adaptive mesh refinement packages. *J. Parallel Distrib. Comput.* **2014**, *74*, 3217–3227. [CrossRef]

78. Stroud, A. *Approximate Calculation of Multiple Integrals*; Prentice-Hall Inc.: Englewood Cliffs, NJ, USA, 1971.

79. Castro, M.; Gallardo, J.; Parés, C. High-order finite volume schemes based on reconstruction of states for solving hyperbolic systems with nonconservative products. Applications to shallow-water systems. *Math. Comput.* **2006**, *75*, 1103–1134. [CrossRef]

80. Parés, C. Numerical methods for nonconservative hyperbolic systems: A theoretical framework. *SIAM J. Numer. Anal.* **2006**, *44*, 300–321. [CrossRef]

81. Rhebergen, S.; Bokhove, O.; van der Vegt, J. Discontinuous Galerkin finite element methods for hyperbolic nonconservative partial differential equations. *J. Comput. Phys.* **2008**, *227*, 1887–1922. [CrossRef]

82. Dumbser, M.; Hidalgo, A.; Castro, M.; Parés, C.; Toro, E. FORCE Schemes on Unstructured Meshes II: Non–Conservative Hyperbolic Systems. *Comput. Methods Appl. Mech. Eng.* **2010**, *199*, 625–647. [CrossRef]

83. Müller, L.O.; Parés, C.; Toro, E.F. Well-balanced high-order numerical schemes for one-dimensional blood flow in vessels with varying mechanical properties. *J. Comput. Phys.* **2013**, *242*, 53–85. [CrossRef]

84. Müller, L.; Toro, E. Well-balanced high-order solver for blood flow in networks of vessels with variable properties. *Int. J. Numer. Methods Biomed. Eng.* **2013**, *29*, 1388–1411. [CrossRef] [PubMed]

85. Gaburro, E.; Dumbser, M.; Castro, M. Direct Arbitrary-Lagrangian-Eulerian finite volume schemes on moving nonconforming unstructured meshes. *Comput. Fluids* **2017**, *159*, 254–275. [CrossRef]

86. Gaburro, E.; Castro, M.; Dumbser, M. Well balanced Arbitrary-Lagrangian-Eulerian finite volume schemes on moving nonconforming meshes for the Euler equations of gasdynamics with gravity. *Mon. Not. R. Astron. Soc.* **2018**, *477*, 2251–2275. [CrossRef]

87. Toro, E.; Hidalgo, A.; Dumbser, M. FORCE Schemes on Unstructured Meshes I: Conservative Hyperbolic Systems. *J. Comput. Phys.* **2009**, *228*, 3368–3389. [CrossRef]

88. Dumbser, M.; Toro, E.F. A Simple Extension of the Osher Riemann Solver to Non-Conservative Hyperbolic Systems. *J. Sci. Comput.* **2011**, *48*, 70–88. [CrossRef]

89. Castro, M.; Pardo, A.; Parés, C.; Toro, E. On some fast well-balanced first order solvers for nonconservative systems. *Math. Comput.* **2010**, *79*, 1427–1472. [CrossRef]

90. Dumbser, M.; Toro, E.F. On Universal Osher–Type Schemes for General Nonlinear Hyperbolic Conservation Laws. *Commun. Comput. Phys.* **2011**, *10*, 635–671. [CrossRef]

91. Einfeldt, B.; Roe, P.L.; Munz, C.D.; Sjogreen, B. On Godunov-type methods near low densities. *J. Comput. Phys.* **1991**, *92*, 273–295. [CrossRef]

92. Dumbser, M.; Balsara, D. A New, Efficient Formulation of the HLLEM Riemann Solver for General Conservative and Non-Conservative Hyperbolic Systems. *J. Comput. Phys.* **2016**, *304*, 275–319. [CrossRef]

93. Dumbser, M.; Enaux, C.; Toro, E. Finite Volume Schemes of Very High Order of Accuracy for Stiff Hyperbolic Balance Laws. *J. Comput. Phys.* **2008**, *227*, 3971–4001. [CrossRef]

94. Dumbser, M.; Zanotti, O. Very High Order PNPM Schemes on Unstructured Meshes for the Resistive Relativistic MHD Equations. *J. Comput. Phys.* **2009**, *228*, 6991–7006. [CrossRef]

95. Harten, A.; Engquist, B.; Osher, S.; Chakravarthy, S. Uniformly high order essentially non-oscillatory schemes, III. *J. Comput. Phys.* **1987**, *71*, 231–303. [CrossRef]

96. Dumbser, M.; Käser, M.; Titarev, V.; Toro, E. Quadrature-Free Non-Oscillatory Finite Volume Schemes on Unstructured Meshes for Nonlinear Hyperbolic Systems. *J. Comput. Phys.* **2007**, *226*, 204–243. [CrossRef]

97. Taube, A.; Dumbser, M.; Balsara, D.; Munz, C. Arbitrary High Order Discontinuous Galerkin Schemes for the Magnetohydrodynamic Equations. *J. Sci. Comput.* **2007**, *30*, 441–464. [CrossRef]

98. Hidalgo, A.; Dumbser, M. ADER Schemes for Nonlinear Systems of Stiff Advection-Diffusion-Reaction Equations. *J. Sci. Comput.* **2011**, *48*, 173–189. [CrossRef]

99. Jackson, H. On the eigenvalues of the ADER-WENO Galerkin predictor. *J. Comput. Phys.* **2017**, *333*, 409–413. [CrossRef]

100. Zanotti, O.; Dumbser, M. Efficient conservative ADER schemes based on WENO reconstruction and space-time predictor in primitive variables. *Comput. Astrophys. Cosmol.* **2016**, *3*, 1. [CrossRef]

101. Owren, B.; Zennaro, M. Derivation of efficient, continuous, explicit Runge–Kutta methods. *SIAM J. Sci. Stat. Comput.* **1992**, *13*, 1488–1501. [CrossRef]

102. Gassner, G.; Dumbser, M.; Hindenlang, F.; Munz, C. Explicit one-step time discretizations for discontinuous Galerkin and finite volume schemes based on local predictors. *J. Comput. Phys.* **2011**, *230*, 4232–4247. [CrossRef]

103. Heinecke, A.; Pabst, H.; Henry, G. LIBXSMM: A High Performance Library for Small Matrix Multiplications. Technical Report, SC'15: The International Conference for High Performance Computing, Networking, Storage and Analysis, Austin (Texas), 2015. Available online: https://github.com/hfp/libxsmm (accessed on 23 August 2018).

104. Breuer, A.; Heinecke, A.; Bader, M.; Pelties, C. Accelerating SeisSol by generating vectorized code for sparse matrix operators. *Adv. Parallel Comput.* **2014**, *25*, 347–356.

105. Breuer, A.; Heinecke, A.; Rettenberger, S.; Bader, M.; Gabriel, A.; Pelties, C. Sustained petascale performance of seismic simulations with SeisSol on SuperMUC. *Lect. Notes Comput. Sci. (LNCS)* **2014**, *8488*, 1–18.

106. Charrier, D.; Weinzierl, T. Stop talking to me—A communication-avoiding ADER-DG realisation. *arXiv* **2018**, arXiv:1801.08682.

107. Boscheri, W.; Dumbser, M. Arbitrary–Lagrangian–Eulerian Discontinuous Galerkin schemes with a posteriori subcell finite volume limiting on moving unstructured meshes. *J. Comput. Phys.* **2017**, *346*, 449–479. [CrossRef]

108. Clain, S.; Diot, S.; Loubère, R. A high-order finite volume method for systems of conservation laws—Multi-dimensional Optimal Order Detection (MOOD). *J. Comput. Phys.* **2011**, *230*, 4028–4050. [CrossRef]

109. Diot, S.; Clain, S.; Loubère, R. Improved detection criteria for the Multi-dimensional Optimal Order Detection (MOOD) on unstructured meshes with very high-order polynomials. *Comput. Fluids* **2012**, *64*, 43–63. [CrossRef]

110. Diot, S.; Loubère, R.; Clain, S. The MOOD method in the three-dimensional case: Very-High-Order Finite Volume Method for Hyperbolic Systems. *Int. J. Numer. Methods Fluids* **2013**, *73*, 362–392. [CrossRef]

111. Loubère, R.; Dumbser, M.; Diot, S. A New Family of High Order Unstructured MOOD and ADER Finite Volume Schemes for Multidimensional Systems of Hyperbolic Conservation Laws. *Commun. Comput. Phys.* **2014**, *16*, 718–763. [CrossRef]

112. Levy, D.; Puppo, G.; Russo, G. Central WENO schemes for hyperbolic systems of conservation laws. *M2AN Math. Model. Numer. Anal.* **1999**, *33*, 547–571. [CrossRef]

113. Levy, D.; Puppo, G.; Russo, G. Compact central WENO schemes for multidimensional conservation laws. *SIAM J. Sci. Comput.* **2000**, *22*, 656–672. [CrossRef]

114. Dumbser, M.; Boscheri, W.; Semplice, M.; Russo, G. Central weighted ENO schemes for hyperbolic conservation laws on fixed and moving unstructured meshes. *SIAM J. Sci. Comput.* **2017**, *39*, A2564–A2591. [CrossRef]

115. Hu, C.; Shu, C. Weighted essentially non-oscillatory schemes on triangular meshes. *J. Comput. Phys.* **1999**, *150*, 97–127. [CrossRef]

116. Sedov, L. *Similarity and Dimensional Methods in Mechanics*; Academic Press: New York, NY, USA, 1959.

117. Kamm, J.; Timmes, F. *On Efficient Generation of Numerically Robust Sedov Solutions*; Technical Report LA-UR-07-2849; LANL: Los Alamos, NM, USA, 2007.

118. Tavelli, M.; Dumbser, M.; Charrier, D.; Rannabauer, L.; Weinzierl, T.; Bader, M. A simple diffuse interface approach on adaptive Cartesian grids for the linear elastic wave equations with complex topography. *arXiv* **2018**, arXiv:1804.09491.

119. Dumbser, M.; Käser, M. An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes—II. The three-dimensional isotropic case. *Geophys. J. Int.* **2006**, *167*, 319–336. [CrossRef]

120. Godunov, S.K.; Zabrodin, A.V.; Prokopov, G.P. A Difference Scheme for Two-Dimensional Unsteady Aerodynamics. *J. Comp. Math. Math. Phys. USSR* **1961**, *2*, 1020–1050.

121. Antón, L.; Zanotti, O.; Miralles, J.A.; Martí, J.M.; Ibáñez, J.M.; Font, J.A.; Pons, J.A. Numerical 3+1 general relativistic magnetohydrodynamics: A local characteristic approach. *Astrophys. J.* **2006**, *637*, 296. [CrossRef]

122. Zanna, L.D.; Zanotti, O.; Bucciantini, N.; Londrillo, P. ECHO: An Eulerian Conservative High Order scheme for general relativistic magnetohydrodynamics and magnetodynamics. *Astron. Astrophys.* **2007**, *473*, 11–30. [CrossRef]

123. Löhner, R. An adaptive finite element scheme for transient problems in CFD. *Comput. Methods Appl. Mech. Eng.* **1987**, *61*, 323–338. [CrossRef]

124. Radice, D.; Rezzolla, L.; Galeazzi, F. High-order fully general-relativistic hydrodynamics: New approaches and tests. *Class. Quantum Gravity* **2014**, *31*, 075012. [CrossRef]

125. Bermúdez, A.; Vázquez, M. Upwind methods for hyperbolic conservation laws with source terms. *Comput. Fluids* **1994**, *23*, 1049–1071. [CrossRef]

126. Alic, D.; Bona-Casas, C.; Bona, C.; Rezzolla, L.; Palenzuela, C. Conformal and covariant formulation of the Z4 system with constraint-violation damping. *Phys. Rev. D* **2012**, *85*, 064040. [CrossRef]

127. Dedner, A.; Kemm, F.; Kröner, D.; Munz, C.D.; Schnitzer, T.; Wesenberg, M. Hyperbolic Divergence Cleaning for the MHD Equations. *J. Comput. Phys.* **2002**, *175*, 645–673. [CrossRef]

128. Gundlach, C.; Martin-Garcia, J. Symmetric hyperbolic form of systems of second-order evolution equations subject to constraints. *Phys. Rev. D* **2004**, *70*, 044031. [CrossRef]

129. Dumbser, M.; Guercilena, F.; Köppel, S.; Rezzolla, L.; Zanotti, O. Conformal and covariant Z4 formulation of the Einstein equations: Strongly hyperbolic first–order reduction and solution with discontinuous Galerkin schemes. *Phys. Rev. D* **2018**, *97*, 084053. [CrossRef]