Journal of Computational Physics

High-order central ENO finite-volume scheme for ideal MHD

CrossMark

A. Susanto [a], L. Ivan [a,*], H. De Sterck [a], C.P.T. Groth [b]

[a] Department of Applied Mathematics, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1
[b] University of Toronto Institute for Aerospace Studies, Toronto, Ontario, Canada M3H 5T6

ABSTRACT

A high-order accurate finite-volume scheme for the compressible ideal magnetohydrodynamics (MHD) equations is proposed. The high-order MHD scheme is based on a central essentially non-oscillatory (CENO) method combined with the generalized Lagrange multiplier divergence cleaning method for MHD. The CENO method uses k-exact multidimensional reconstruction together with a monotonicity procedure that switches from a high-order reconstruction to a limited low-order reconstruction in regions of discontinuous or under-resolved solution content. Both reconstructions are performed on central stencils, and the switching procedure is based on a smoothness indicator. The proposed high-order accurate MHD scheme can be used on general polygonal grids. A highly sophisticated parallel implementation of the scheme is described that is fourth-order accurate on two-dimensional dynamically-adaptive body-fitted structured grids. The hierarchical multi-block body-fitted grid permits grid lines to conform to curved boundaries. High-order accuracy is maintained at curved domain boundaries by employing high-order spline representations and constraints at the Gauss quadrature points for flux integration. Detailed numerical results demonstrate high-order convergence for smooth flows and robustness against oscillations for problems with shocks. A new MHD extension of the well-known Shu–Osher test problem is proposed to test the ability of the high-order MHD scheme to resolve small-scale flow features in the presence of shocks. The dynamic mesh adaptation capabilities of the approach are demonstrated using adaptive time-dependent simulations of the Orszag–Tang vortex problem with high-order accuracy and unprecedented effective resolution.

© 2013 Elsevier Inc. All rights reserved.

1. Introduction

This paper proposes a high-order accurate finite-volume (FV) scheme for the compressible ideal magnetohydrodynamics (MHD) equations. The high-order MHD scheme is based on the central essentially non-oscillatory (CENO) method that was introduced for the compressible Euler equations by Ivan and Groth [1], and has since been extended to the Navier–Stokes equations [2–4]. The CENO method uses Barth's k-exact reconstruction mechanism [5] to obtain high-order solution accuracy in combination with a monotonicity procedure that switches between a high-order reconstruction and a limited low-order reconstruction. Both reconstructions are performed on central stencils, and the switching is based on a smoothness indicator [1]. The hybrid CENO approach is combined in this paper with the generalized Lagrange multiplier (GLM) divergence cleaning method for MHD that was proposed by Dedner et al. [6] to obtain a FV MHD scheme that is high-order accurate in smooth flow regions and robust against spurious oscillations at discontinuities. The proposed high-order

accurate MHD scheme has several desirable properties. First, it is suitable for general polygonal grids because Barth's k-exact polynomial reconstruction procedure is inherently multi-dimensional and can be used on general stencils that do not need to be grid-aligned. Second, the scheme can in principle be implemented with arbitrary order. Third, it can be used directly on block-adaptive grids, which can pose significant challenges to MHD schemes due to the $\nabla \cdot \vec{B}$ constraint. And fourth, high-order accuracy is maintained at curved domain boundaries of the body-fitted mesh by employing accurate spline representations for the boundaries that enable high-order accurate surface flux computations. These features constitute significant new developments in high-order finite-volume schemes for MHD. The particular implementation of the proposed scheme that we present in this paper is a highly sophisticated fourth-order accurate parallel MHD method on two-dimensional (2D) dynamically-adaptive multi-block body-fitted quadrilateral grids. Our adaptive 2D body-fitted structured grids are composed of blocks with quadrilateral cells that are organized in a rectangular structure as on a Cartesian grid and, thus, the grid blocks are sometimes also referred to as logically Cartesian. The body-fitted meshes in our work can have grid lines conforming to curved boundary surfaces and stretching of the grid lines is permitted to allow for anisotropic mesh spacing.

Development of high-order numerical methods for MHD is an active area of research. Just like for other nonlinear hyperbolic systems, spurious oscillations at shocks are a major challenge in MHD, but the $\nabla \cdot \vec{B}$ constraint is an additional significant challenge in MHD. Indeed, it is well-known that simply extending conservation law methods for the Euler equations to the MHD hyperbolic system fails to work, since $\nabla \cdot \vec{B}$ may grow in an uncontrolled fashion (beyond truncation error levels), which may result in unphysical forces and numerical instability [7,8]. A variety of approaches have been proposed to remedy this issue. One option is to employ an elliptic correction scheme, called the "Hodge Projection", which essentially projects a vector field onto its solenoidal part [7,9]. While the elliptic correction scheme maintains solenoidality up to machine accuracy (in the chosen discretization), it requires a Poisson equation to be solved at each hyperbolic step. This is not natural in a 'hyperbolic' simulation and can be inconvenient in terms of implementation, especially in parallel since the discrete Poisson potential variables are tightly coupled across the whole computational domain. As an alternative, Powell [8] proposed a divergence control method that only attempts to approximately satisfy the divergence constraint. In Powell's approach, the ideal MHD system is rewritten into its symmetrizable and Galilean-invariant form through the introduction of source terms proportional to the divergence of the magnetic field. This modification maintains the hyperbolic character of the MHD equations, but comes at the cost of conservation, and may lead to incorrect jumps for problems with strong discontinuities [10]. For this reason, this approach has lost some of its initial popularity. A third method to control $\nabla \cdot \vec{B}$ is the class of schemes that fall under the category of 'constrained transport' methods, which preserve the solenoidality of the magnetic field through staggered spatial discretizations [11]. The normal components of the magnetic field are stored on cell faces, and in every time-step the field is updated in such a way that $\nabla \cdot \vec{B}$ remains zero up to machine accuracy (in the chosen discretization). This approach, however, requires the magnetic field variables to be treated differently from the fluid variables, which may be inconvenient for implementation. The approach is attractive from a physical point of view and is straightforward to derive and implement for second-order accurate codes on regular Cartesian grids. It can be extended with second-order accuracy to logically Cartesian grids and to triangular or tetrahedral unstructured grids [12,13], but extensions beyond second order [14] and to general polygonal grids are far from trivial. In particular, interpolation and restriction of the magnetic field at resolution changes on block-adaptive grids need to be treated very carefully and sophisticated approaches have been developed for this purpose [14–16].

More recently, Dedner et al. [6] proposed the GLM-based divergence cleaning technique. Through the introduction of a new transport variable, the divergence error is convected out of the domain, while keeping the hyperbolicity of the system intact. Unlike the Powell source term method, conservation in all physical variables is maintained. And unlike approaches following the constrained transport methodology, there is no need to stagger the grid, or to place the magnetic fields at locations different from those where the fluid variables are located. The GLM approach can easily and naturally be applied on general polygonal grids, and there is no need to integrate complicated source terms involving flow variable derivatives as in Powell's approach. The GLM-MHD approach thus provides an attractive alternative to the more commonly established ways of divergence control, because it is effective in controlling divergence error, is simple to implement, preserves conservation, and can easily be applied on general grids. We choose GLM for our MHD scheme because, combined with the CENO method, it leads to a high-order MHD scheme that can be applied on general grids (including our adaptive multi-block body-fitted structured grid) and naturally handles resolution changes on block-adaptive grids, as demonstrated herein.

Adaptive mesh refinement has proven to be very effective for treating problems with disparate length scales, providing the required spatial resolution while minimizing memory and storage requirements. Recently, Groth and co-researchers [17–21] have developed a flexible block-based adaptive mesh refinement (AMR) scheme allowing automatic solution-directed mesh adaptation on multi-block body-fitted meshes consisting of two-dimensional quadrilateral and three-dimensional hexahedral computational cells. We have implemented our new high-order MHD scheme that combines CENO and GLM into a hierarchical quadtree block-based AMR procedure for multi-block body-fitted quadrilateral mesh that is based on this previous work [17–19,21]. This block-based approach has been shown to enable efficient and scalable parallel implementations for a variety of flow problems, as well as to allow for local refinement of body-fitted mesh with anisotropic stretching. The latter aids in the treatment of complex flow geometry and flows with thin boundary, shear, and mixing layers and/or discontinuities and shocks. Extensions of the block-based body-fitted AMR approach for embedded boundaries not aligned with the mesh [22] and with an anisotropic refinement strategy [23] are also possible and have been developed.

In recent years, various high-order schemes have been proposed for the MHD system. Many recent developments employ discontinuous Galerkin (DG) finite-element methods [24–27], and others are based on essentially non-oscillatory (ENO) FV

schemes and on weighted ENO (WENO) FV schemes [13,28–33]. Most of these high-order approaches were only described and implemented for regular Cartesian grids. Our high-order MHD scheme uses a different approach. As already mentioned, it is based on Barth's k-exact reconstruction procedure [5], which uses a least-squares approach on overdetermined stencils to compute polynomial reconstruction coefficients, in a multi-dimensional way that can handle general polygonal grids. In order to control spurious oscillations at shocks, we use the CENO monotonicity procedure that was introduced by Ivan and Groth [1] for the Euler equations, and has since been extended to the Navier–Stokes equations [2–4]. Our implementation of this CENO monotonicity procedure switches between an unlimited piecewise cubic reconstruction (fourth-order accurate) and a limited piecewise-linear reconstruction (second-order accurate), with the switching based on the smoothness indicator introduced in [1]. Note that the scheme we describe can in principle be implemented with arbitrary order, but fourth-order accuracy is a suitable practical choice for the numerical results to be presented in this paper. The smoothness indicator is computed in each cell to determine whether the flow is locally smooth and well-resolved. For cells containing non-smooth or under-resolved solution content, the unlimited k-exact reconstruction is switched to limited piecewise linear reconstruction. The smoothness indicator can also be used directly to formulate a criterion for AMR. The CENO scheme is called central because both the high-order and the low-order stencils are central with respect to the cell. The method is called an ENO method because it satisfies the ENO property [34]:

$$TV(u^{n+1}) = TV(u^n) + O(\Delta x^{k+1}), \tag{1}$$

where u^n denotes a solution variable u at time level n, Δx is the grid spacing, k is the order of polynomial reconstruction, and TV stands for total variation. The ENO property allows the presence of small spurious oscillations that have a magnitude on the order of the truncation error, but it does not allow $O(1)$ Gibbs-like oscillations at discontinuities [34]. It is important to note that the CENO method proposed by Ivan and Groth [1] does not choose between asymmetric stencils as most other methods do that try to enforce the ENO property, but instead uses a hybrid approach that chooses between high-order and low-order central reconstructions. Note that Harten and Chakravarthy [35] also proposed a technique on Cartesian grids to obtain an ENO reconstruction using central stencils by hybridizing a high-order reconstruction with a first-order formulation, and this served as an inspiration for the CENO approach of [1,2]. The fixed stencil used during the CENO reconstruction procedure avoids the complexity of considering multiple non-central stencil configurations that characterizes traditional ENO schemes. Note also that the CENO method of Ivan and Groth is not a central ENO method in the sense of Nessyahu and Tadmor's staggered mesh philosophy [36], but it uses non-staggered central stencils of different order. We note that our limited low-order least-squares scheme for MHD with GLM divergence cleaning is similar to the discretization proposed by Yalim et al. in [37] (implemented on unstructured grids), and our high-order method is a high-order extension of this approach that combines least-squares reconstruction with GLM. Our CENO-GLM high-order MHD scheme thus provides an alternative to high-order DG and ENO/WENO methods for MHD, and is attractive because it can naturally be applied on general grids.

This paper is organized as follows. The ideal MHD equations and the GLM formulation are described in Section 2. In Section 3 we give a detailed description of our high-order MHD scheme, which is obtained by combining the CENO method with GLM divergence cleaning. Section 4 describes detailed numerical results that demonstrate high-order convergence for smooth flows, and robustness against oscillations for Riemann problems and other flows with shocks. In particular, we also present a new MHD extension of the well-known Shu–Osher test problem [38] to test the ability of our high-order MHD scheme to resolve small-scale flow features in the presence of shocks. Finally, we demonstrate the dynamic mesh refinement capabilities of our implementation using adaptive time-dependent simulations of the Orszag–Tang vortex problem [39] with high-order accuracy and unprecedented effective resolution. Concluding remarks are presented in Section 5.

2. Ideal MHD

2.1. Ideal MHD equations

The ideal MHD system is described by the following equations in conservation form:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{v}) = 0, \tag{2}$$

$$\frac{\partial (\rho \vec{v})}{\partial t} + \nabla \cdot \left(\rho \vec{v}\vec{v} + \vec{I}\left(p + \frac{\vec{B} \cdot \vec{B}}{2} \right) - \vec{B}\vec{B} \right) = 0, \tag{3}$$

$$\frac{\partial \vec{B}}{\partial t} + \nabla \cdot (\vec{v}\vec{B} - \vec{B}\vec{v}) = 0, \tag{4}$$

$$\frac{\partial e}{\partial t} + \nabla \cdot \left(\left(e + p + \frac{\vec{B} \cdot \vec{B}}{2} \right) \vec{v} - (\vec{v} \cdot \vec{B})\vec{B} \right) = 0. \tag{5}$$

Eqs. (2)–(5) are supplemented with a solenoidality condition for the magnetic field,

$$\nabla \cdot \vec{B} = 0. \tag{6}$$

The conserved quantities of the ideal MHD equation system are the density, ρ, the momentum, $\rho \vec{v}$ (with $\vec{v}$ being the velocity), the magnetic field, $\vec{B}$, and the energy, e. The plasma pressure, p, is given by the equation of state for a perfect gas

$$p = (\gamma - 1)\left(e - \frac{1}{2}\rho|\vec{v}|^2 - \frac{1}{2}|\vec{B}|^2\right), \tag{7}$$

where γ is the adiabatic index. We use $\gamma = 5/3$ in our numerical tests except where noted.

2.2. GLM control of the $\nabla \cdot \vec{B}$ constraint

The GLM-MHD formulation can be described as follows. Following a similar approach as for the Maxwell equations [40], the divergence constraint (Eq. (6)) can be coupled with the induction equation through the introduction of a new potential variable, ψ [6]. We consider the so-called hybrid hyperbolic-parabolic approach proposed in [6], in which the equations describing the evolution of the magnetic field, Eqs. (4) and (6), are replaced with

$$\frac{\partial \vec{B}}{\partial t} + \nabla \cdot (\vec{v}\vec{B} - \vec{B}\vec{v}) + \nabla \psi = 0, \tag{8}$$

$$\frac{\partial \psi}{\partial t} + c_h^2 \nabla \cdot \vec{B} = -\frac{c_h^2}{c_p^2}\psi. \tag{9}$$

As can be seen from these equations, the system is still conservative except for the evolution equation of ψ, which is not a physical variable. This preservation of conservation for physical variables is the main advantage of the GLM method over the Powell method that was proposed earlier to approximately satisfy the divergence constraint [41]. Two new eigenvalues arise in the GLM-MHD formulation, which are $\pm c_h$. The coefficients c_p and c_h control the amount of diffusion in ψ and the advection speed, respectively. The 'purely hyperbolic' correction can be obtained by taking c_p to infinity (no diffusion). To ensure that the error is advected as fast as possible, it is desirable to set c_h as high as possible. However, because the two new eigenvalues have magnitude c_h, it is also important to set c_h small enough so that it will not affect the time-step criterion of the simulation. Thus, c_h is often chosen to be the largest of all MHD eigenvalues in the whole domain over all cell interfaces (i,j), which can be written as

$$c_h = \max_{i,j}\left(|v_n| + c_{f_n}\right), \tag{10}$$

where v_n and c_{f_n} are the plasma velocity and the fast magnetosonic wave speed in the direction normal to the interfaces (i,j). In [6] it is recommended to choose c_p by setting the parameter $c_r = c_p^2/c_h$ to a constant value of 0.18. In [32,33], it is suggested to choose c_p by setting the parameter $\bar{\alpha} = \Delta x c_h/c_p^2$ to a constant value in $[0, 1]$, where Δx is a measure of the grid spacing. Note that $\bar{\alpha}$ is a dimensionless quantity reflecting the ratio of the diffusive and advective time scales.

2.3. Boundary condition treatment of ψ at inflow and outflow boundaries

The choice of c_h as given by Eq. (10) ensures that no eigenvalue will exceed the largest physical eigenvalue in the domain, while at the same time, it guarantees that the divergence error will be advected out of the simulation domain with the fastest physical wave speed in the flow solution. Since the two additional eigenvalues are $\pm c_h$ regardless of the actual plasma velocities and wave speeds, eigenvalues of both signs will always exist at all cell interfaces. This means that treatment similar to subsonic inlet and outlet boundary conditions (see [42]) is always required for inflow and outflow boundary conditions. Since the waves with eigenvalues $\pm c_h$ only carry changes in the normal magnetic field and ψ [6], only these two variables need to be taken into account at boundaries to accommodate these waves. For example, consider superfast inflow boundary conditions and assume without loss of generality that $v_n > 0$. Since the inflow velocity is faster than the fast magnetosonic wave, all the MHD eigenvalues are positive (information travels into the computational domain). However, for GLM-MHD one cannot just prescribe all variables, because one of the eigenvalues, $-c_h$, is necessarily negative, even when the flow is superfast at the inflow boundary. One of either ψ or the normal magnetic field has to be extrapolated from the interior solution, and because the inflow magnetic field is prescribed at the boundary, it is ψ that has to be extrapolated from the interior. The same logic applies to superfast outflow. Assume again that $v_n > 0$. Without GLM, all of the variables would just be extrapolated from the inside of the domain, since all eigenvalues are positive, hence no information is propagating into the domain. However, due to the negative eigenvalue $-c_h$, ψ needs to be prescribed at the outflow boundary. A suitable choice for ψ is to set it to zero at superfast outflow boundaries. (This is consistent with Yalim et al. [37], who set ψ to a constant at the superfast outlet boundaries.)

3. High-order CENO scheme for ideal MHD with GLM divergence cleaning

In this section we give a detailed description of the proposed high-order CENO scheme for MHD, which is obtained by combining Ivan and Groth's CENO approach with GLM divergence cleaning. We first describe the high-order FV framework, followed by discussions on Barth's k-exact reconstruction and the CENO reconstruction selection process using the CENO smoothness indicator. While the CENO method applies to general mesh topologies, the discussion here is restricted to the application to 2D, multi-block, body-fitted (logically Cartesian) AMR grids having quadrilateral computational elements of the type considered by Groth and co-researchers [17–19,21]. Numerical flux calculation and source term integration for our high-order MHD CENO method are described next, followed by a discussion on how our MHD CENO implementation obtains high-order accuracy at curved boundaries.

3.1. High-order finite-volume formulation

Consider hyperbolic conservation law

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \vec{\mathbf{F}} = \mathbf{S}, \tag{11}$$

where $\mathbf{U}$ is the vector of conserved variables, $\vec{\mathbf{F}}$ consists of the flux terms of the system, and $\mathbf{S}$ is a source term vector. For a quadrilateral cell (i,j), the semi-discrete FV form of Eq. (11) is given as

$$\frac{d\overline{\mathbf{U}}_{i,j}}{dt} = -\frac{1}{A_{i,j}} \sum_{l=1}^{4} \sum_{m=1}^{N_g} (\omega \vec{\mathbf{F}}_{num} \cdot \vec{n} \Delta l)_{i,j,l,m} + \overline{\mathbf{S}}_{i,j}, \tag{12}$$

where $\overline{\mathbf{U}}_{i,j}$ is the numerical approximation of the average value of $\mathbf{U}$ in cell (i,j), $\vec{\mathbf{F}}_{num}$ is the numerical flux function, $A_{i,j}$ is the area of the computational cell (i,j), N_g is the number of Gauss quadrature points on each cell face and ω is the associated Gauss quadrature weight to each of the Gauss points. The actual number of flux quadrature points, N_g, depends on the order of solution reconstruction, with two Gauss quadrature points per face for third- and fourth-order accurate schemes, but only one Gauss quadrature point per face for second-order or lower [1]. The order of the polynomial reconstruction then determines the spatial accuracy of the solution by providing more accurate approximations of the solution values at the Gauss quadrature points for flux calculation. In general, an order-k polynomial reconstruction provides an order-$(k+1)$ accurate spatial discretization for smooth problems. We use standard explicit second-order and fourth-order Runge–Kutta methods [42] to integrate Eq. (12) in time for the second-order and fourth-order accurate spatial discretizations to be compared in our time-dependent numerical test problems. For steady-state simulations, we use a five-stage optimally smoothing method regardless of the solution accuracy [43].

3.2. k-Exact piecewise polynomial reconstruction

Following Barth [5], the variation of a solution variable, u, at any location within the quadrilateral computational cell (i,j), assumes the form

$$u_{i,j}^k(\vec{X}) = \sum_{\substack{p_1=0 \\ (p_1+p_2 \leqslant k)}}^{k} \sum_{p_2=0}^{k} (x - \overline{x}_{i,j})^{p_1} (y - \overline{y}_{i,j})^{p_2} D_{p_1 p_2}^k, \tag{13}$$

where k is the order of the polynomial function, $\vec{X} = (x,y)$ are the coordinates at which the solution is sought, $(\overline{x}_{i,j}, \overline{y}_{i,j})$ are the coordinates of the centroid of cell (i,j), and $D_{p_1 p_2}^k$ are high-order polynomial coefficients that will need to be determined for each of the reconstructed (e.g., primitive) variables for every cell, based on a set of cell averages, $\overline{u}_{\gamma,\delta}$, in the neighbourhood of cell (i,j). For the test cases presented in this paper, linear ($k=1$) and cubic ($k=3$) reconstructions are chosen to obtain second- and fourth-order accurate schemes. The monotonicity-preserving procedure, which is discussed in Section 3.3, reduces k to 1 and applies limiters in regions of the flow that are deemed under-resolved or to contain discontinuities.

The coefficients $D_{p_1 p_2}^k$ are determined by solving an overdetermined system of linear equations in a least-squares sense, fitting the reconstruction polynomial to the solution averages for cell (i,j) and for its neighbouring cells in the stencil of cell (i,j). For a polynomial of degree k, the number of coefficients $D_{p_1 p_2}^k$ is given by $\mathcal{N}_D = \frac{(k+1)(k+2)}{2}$ [1,2,4]. Thus, there are 3 coefficients to be determined for $k=1$ or linear reconstruction and 10 coefficients for $k=3$ or cubic reconstruction. Following the requirements imposed by Barth [5], it is important that these coefficients are determined in such a way that the following conditions are satisfied:

- *Conservation of the mean.* The average of the reconstructed polynomial function over cell (i,j) should recover exactly the cell-averaged value $\overline{u}_{i,j}$:

$$\overline{u}_{i,j} = \frac{1}{A_{i,j}} \iint_{\mathcal{A}_{i,j}} u_{i,j}^k(\vec{X}) \, dA. \tag{14}$$

- *k-Exactness.* The reconstructed polynomial function should be able to reconstruct polynomials of degree up to k exactly [5]:

$$u_{i,j}^k(\vec{X}) - u_{\text{exact}}(\vec{X}) = O(\Delta x^{k+1}).$$

(15)

- *Compact support.* The reconstructed polynomial function should depend only on average values within a relatively small neighbourhood [5]. Only the cell-averaged data within the supporting stencil is used for reconstruction purposes.

In theory, $\mathcal{N}_D$ determines the minimum size of the supporting stencil, but in practice more neighbours are included to make the reconstruction more robust for complicated and stretched meshes [1,2,4]. On our 2D body-fitted structured grid blocks, first-degree neighbours are included for $k = 0$ and $k = 1$ reconstruction stencils (a total of 8 neighbours), and first- and second-degree neighbours are included for $k = 2$ and $k = 3$ reconstruction stencils (a total of 24 neighbours).

Consider reconstruction for cell (i,j). In the reconstruction step an overdetermined system $\mathbb{A}\mathbf{D} - \mathbf{B} = 0$ is solved in the least-squares sense, together with the constraint of Eq. (14), which is imposed exactly. Here, $\mathbf{D}$ is the array of polynomial coefficients, $D_{p_1 p_2}^k$, and the equations $\mathbb{A}\mathbf{D} - \mathbf{B} = 0$ are given by

$$(\mathbb{A}\mathbf{D} - \mathbf{B})_{\gamma,\delta} = \left(\frac{1}{A_{\gamma,\delta}} \iint_{\mathcal{A}_{\gamma,\delta}} u_{i,j}^k(\vec{X})\, dA \right) - \overline{u}_{\gamma,\delta} = 0.$$

(16)

There is one equation for each cell (γ,δ) in the stencil of cell (i,j). Each equation matches the actual cell average $\overline{u}_{\gamma,\delta}$ in cell (γ,δ) with the average over cell (γ,δ) of the reconstructed polynomial $u_{i,j}^k(\vec{X})$ for cell (i,j). Equation (14) is enforced analytically by replacing $u_{i,j}^k$ with Eq. (13) and expressing the first coefficient, D_{00}^k, as a function of the other $M = \mathcal{N}_D - 1$ polynomial unknowns as

$$D_{00}^k = \overline{u}_{i,j} - \sum_{\substack{p_1=0 \\ (p_1+p_2 \neq 0)}}^{k} \sum_{p_2=0}^{k} D_{p_1 p_2}^k \left(\overline{x^{p_1} y^{p_2}} \right)_{i,j},$$

(17)

where the geometric moment $\left(\overline{x^{p_1} y^{p_2}} \right)_{i,j}$ of powers (p_1, p_2) is given by

$$\left(\overline{x^{p_1} y^{p_2}} \right)_{i,j} = \frac{1}{A_{i,j}} \iint_{\mathcal{A}_{i,j}} \left(x - \overline{x}_{i,j} \right)^{p_1} \left(y - \overline{y}_{i,j} \right)^{p_2} dA.$$

(18)

Substituting $u_{i,j}^k$ from Eq. (13) in Eq. (16) and using Eq. (17) for D_{00}^k the following overdetermined linear system for the M unknowns is obtained

$$\begin{bmatrix} L_1 \\ L_2 \\ \vdots \\ L_J \\ \vdots \\ L_{N_n} \end{bmatrix}_{N_n \times M} \begin{pmatrix} D_{01}^k \\ D_{02}^k \\ \vdots \\ D_{p_1 p_2} \\ \vdots \\ D_{k0}^k \end{pmatrix}_{M \times 1} - \begin{pmatrix} w_1(\overline{u}_1 - \overline{u}_I) \\ w_2(\overline{u}_2 - \overline{u}_I) \\ \vdots \\ w_J(\overline{u}_J - \overline{u}_I) \\ \vdots \\ w_{N_n}(\overline{u}_{N_n} - \overline{u}_I) \end{pmatrix}_{N_n \times 1} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \vdots \\ 0 \end{pmatrix}_{N_n \times 1},$$

(19)

where a unique index $J = (\gamma,\delta)$ has been assigned to each of the N_n neighbours in the supporting reconstruction stencil and the index $I = (i,j)$ denotes the cell having the solution reconstructed. The generic row L_J of the matrix $\mathbb{A}$ for a neighbouring cell J is given by

$$L_J = \left(w_J \left(\widehat{x^0 y^1} \right)_{IJ} \quad w_J \left(\widehat{x^0 y^2} \right)_{IJ} \quad \cdots \quad w_J \left(\widehat{x^{p_1} y^{p_2}} \right)_{IJ} \quad \cdots \quad w_J \left(\widehat{x^k y^0} \right)_{IJ} \right),$$

(20)

in which w_J is a geometric weight specific to each neighbour J which serves the purpose of improving the locality of the reconstruction, becoming especially important for stretched meshes with boundary curvature [44]. (In essence, equations corresponding to close-by neighbour cells in the reconstruction stencil get larger weights in the least-squares solution than neighbour cells that are further away.) The matrix coefficients $\left(\widehat{x^{p_1} y^{p_2}} \right)_{IJ}$ for the pair of I and J cells have the expression

$$\left(\widehat{x^{p_1} y^{p_2}} \right)_{IJ} = \left(\frac{1}{A_{\gamma,\delta}} \iint_{\mathcal{A}_{\gamma,\delta}} \left(x - \overline{x}_{i,j} \right)^{p_1} \left(y - \overline{y}_{i,j} \right)^{p_2} dA \right) - \left(\overline{x^{p_1} y^{p_2}} \right)_{i,j},$$

(21)

where the quantities $\left(\widehat{x^{p_1} y^{p_2}} \right)_{IJ}$ depend only on the geometry of I and J cells, and involve a monomial integration that can be computed by applying quadrature rules. An efficient way to calculate the geometric moments $\left(\widehat{x^{p_1} y^{p_2}} \right)_{IJ}$ using only the $\left(\overline{x^{p_1} y^{p_2}} \right)$ moments is described in [2].

QR factorization or multiplication with the pseudo-inverse of $\mathbb{A}$ can be used to determine the solution of Eq. (19), as described in more detail in [1,2,4]. The complete solution of the constrained least-squares problems is then obtained by calculating D_{00} using Eq. (17). In each time step, the constrained least-squares reconstruction problem is solved for each cell and for each primitive variable. Matrix $\mathbb{A}$ depends completely on the geometry and is the same for all least-squares problems in a given cell (i,j) (i.e., for each solution variable) and for all time steps, so it can be precomputed and stored for computational efficiency (see [1,2,4] for details). As explained in Section 3.5, one-sided stencils and additional constraints on the least-squares solution are used to handle boundary conditions with high-order accuracy at curved boundaries.

3.3. CENO smoothness indicator to enforce monotonicity

The CENO method controls monotonicity throughout the computational domain by selecting a limited linear reconstruction in cells where the flow is deemed to be non-smooth or under-resolved, and a high-order k-exact reconstruction elsewhere. The limited linear reconstruction is based on k-exact reconstruction with $k = 1$ combined with the standard Venkatakrishnan limiter, see [2,45]. To estimate whether the flow in cell (i,j) is under-resolved or non-smooth, a variable S, the smoothness indicator, is computed [1]:

$$S = \frac{\alpha c_s}{\max(1 - \alpha, \epsilon)},$$ (22)

where α is given by

$$\alpha = 1 - \frac{\sum_\gamma \sum_\delta \left(u_{\gamma,\delta}^k(\vec{X}_{\gamma,\delta}) - u_{i,j}^k(\vec{X}_{\gamma,\delta}) \right)^2}{\sum_\gamma \sum_\delta \left(u_{\gamma,\delta}^k(\vec{X}_{\gamma,\delta}) - \overline{u}_{i,j} \right)^2}$$ (23)

and $c_s = (\mathcal{N}_{SOS} - \mathcal{N}_D)/(\mathcal{N}_D - 1)$ is a positive constant. Here, $\mathcal{N}_{SOS}$ stands for 'size of stencil' used for reconstruction, $\mathcal{N}_D$ stands for 'degrees of freedom' and denotes the number of unknown polynomial coefficients, and ϵ is introduced to avoid division by zero (we use $\epsilon = 10^{-8}$). Further, γ and δ denote the indices of the neighbouring cells to the cell (i,j) that are part of its reconstruction stencil, and $\vec{X}_{\gamma,\delta}$ is the centroid of cell (γ,δ). (Note that the stencil used for computing the smoothness indicator can also be chosen smaller than the reconstruction stencil. In our numerical results, we compute the smoothness indicator associated with each primitive solution variable in cell (i,j) using a stencil with nine cells, i.e., the cell (i,j) and its eight first-degree neighbours.) The parameter α basically measures how accurately centroidal solution values of neighbouring cells can be reproduced using the reconstruction for cell (i,j). The range of α is $-\infty < \alpha \leqslant 1$: for smooth variation, the second term of the right-hand side of Eq. (23) tends to be close to zero and α is very close to one; for cells close to a discontinuity or with an under-resolved feature, the magnitude of α tends away from one and it can also become negative. The range of the smoothness indicator S is $-c_s < S < c_s/\epsilon$: for smooth variation (α very close to one), S is large; for non-smooth or under-resolved features (α away from one), S is small. The smoothness indicator S is then compared with a cutoff value S_C: when $S > S_C$ the solution is deemed locally smooth and the high-order reconstruction is used, and for $S \leqslant S_C$ the solution is locally non-smooth or under-resolved, and the limited low-order reconstruction is used. We also use S in our adaptive procedure to refine regions where the solution is non-smooth or under-resolved. A potential disadvantage of this approach is that it is not fully parameter-free. However, we have found it easy to pick suitable values of S_C based on the range recommended in [2] for the numerical tests shown in Section 4. The selection of an appropriate cutoff value is also made easier by the use of the transition function $\alpha/(1 - \alpha)$, which rapidly magnifies small variations in α very close to one. Additionally, it is worth emphasizing that a single value S_C is selected and applied to all solution variables and all mesh resolutions used for solving a particular problem. Note also that the use of the adjustment coefficient, c_s, in the expression of S helps making the selection of S_C relatively independent of the order of the scheme and making the smoothness indicators comparable for different stencil sizes that may occur at domain boundaries. The form of the smoothness indicator is inspired by the definition of multiple-correlation coefficients and least-squares goodness-of-fit testing; see [2] for a more detailed discussion with further motivation for the approach. As is shown by extensive testing for the Euler and Navier–Stokes equations in [1–4] and is further confirmed by the numerical MHD tests presented below, the CENO approach with smoothness indicator S is robust in terms of providing high-order accurate numerical approximations while avoiding spurious oscillations.

One more element has to be added to the approach in order to get good results for problems with uniform regions. In uniform regions, the formula for α in Eq. (23) may lead to 0/0 in the second term of the right-hand side, rendering the smoothness indicator unpredictable. It has been observed before [46] in a related context that it is desirable to eliminate the effect of switching mechanisms altogether in nearly uniform regions, and just use high-order reconstruction. To do so, we define the newly proposed parameter

$$\xi_{i,j} = \sqrt{\sum_{\substack{p_1=0 \\ (0 < p_1 + p_2 \leqslant k)}}^{k} \sum_{p_2=0}^{k} (D_{p_1 p_2}^k)^2 (A_{i,j})^{p_1 + p_2}},$$ (24)

which measures the variability of solution variable u in cell (i,j). (It takes into account all the derivatives at the centroid of cell (i,j).) When $\xi_{i,j}$ is smaller than a threshold value (low variability), high-order reconstruction is always used, and only when $\xi_{i,j}$ is greater than the threshold the smoothness indicator is computed and the CENO switching mechanism is activated. In particular, the smoothness indicator for the solution variable u is evaluated in cell (i,j) when

$$\xi_{i,j} > \epsilon_A + \epsilon_R \overline{u}_{i,j}, \tag{25}$$

where ϵ_A and ϵ_R represent absolute and relative variability thresholds, chosen to be 10^{-5} for the simulations performed in this paper.

3.4. Numerical flux function and source term integration

In this subsection we discuss numerical flux computation and treatment of the GLM source term (in Eq. (9)) for the high-order MHD CENO scheme. We use the Lax–Friedrichs numerical flux function for the implementation of the proposed high-order MHD CENO scheme. Following Dedner et al. [6], the equations for B_x and ψ are decoupled from the rest of the system, so the Lax–Friedrichs numerical fluxes are applied only to the other seven variables. The fluxes at the interfaces for B_x and ψ are calculated by setting these variables to the following values at the cell interfaces [6]:

$$B_{x,m} = \frac{1}{2}(B_{x,r} + B_{x,l}) - \frac{1}{2 c_h}(\psi_r - \psi_l), \tag{26}$$

$$\psi_m = \frac{1}{2}(\psi_r + \psi_l) - \frac{c_h}{2}(B_{x,r} - B_{x,l}), \tag{27}$$

where the subscripts l and r denote the left and right reconstructed states at cell interfaces and c_h is the global maximum of $|v_x| + c_{f_x}$ at cell interfaces. These values are substituted directly into the exact flux formulas for the B_x and ψ equations. In a multi-dimensional setting, B_x is effectively B_n, which is the magnetic field component normal to the interface. These ψ_m and $B_{n,m}$ values are also used for flux calculation of the other seven variables, which uses the Lax–Friedrichs numerical flux with local values of $|v_x| + c_{f_x}$ as the largest wave speed that determines the size of the numerical dissipation.

As an alternative, one can also apply the standard Lax–Friedrichs flux directly to the full system with nine variables, without decoupling the 2×2 system. One can expect this to be more diffusive since in this case c_h (the global maximum of $|v_x| + c_{f_x}$) determines the numerical diffusion, but we have not found much difference with the decoupled approach when trying this for our numerical tests. Nevertheless, in the numerical results presented below we use the decoupled approach. Other flux functions such as Roe and HLLE can also be considered. Wheatley et al. [47] compared flux functions for high-order DG methods, and found that using more accurate Riemann solvers improves results in some cases (e.g., at shocks), but does often not make much difference in smooth regions of the flow. This is also expected for CENO since the intercellular solution jumps diminish in size as the reconstruction order increases, but a detailed investigation of this for the CENO MHD scheme is beyond the scope of this paper.

The ideal (non-modified) MHD system (Eqs. (2)–(5)) is a hyperbolic system of equations, so it easily fits within the CENO framework. The GLM-MHD formulation adds a source term to the ψ-update equation (Eq. (9)), which can be treated in two different ways in our implementation. A first option, as proposed in Dedner et al. [6], is to incorporate the source term using an operator splitting approach, in which the following ordinary differential equation is solved analytically for each cell in each time step:

$$\frac{d\overline{\psi}_{i,j}}{dt} = -\frac{c_h^2}{c_p^2} \overline{\psi}_{i,j}. \tag{28}$$

The second option is to integrate the source term as part of the hyperbolic system update. Note that this can be done automatically with high-order accuracy: integrating Eq. (9) over cell (i,j) gives

$$\frac{d}{dt} \left(\iint_{\mathcal{A}_{i,j}} \psi \, dA \right) = -c_h^2 \left(\iint_{\mathcal{A}_{i,j}} \nabla \cdot \vec{B} \, dA \right) - \frac{c_h^2}{c_p^2} \left(\iint_{\mathcal{A}_{i,j}} \psi \, dA \right), \tag{29}$$

which directly leads to the discrete equation

$$\frac{d\overline{\psi}_{i,j}}{dt} = -\frac{1}{A_{i,j}} \sum_{l=1}^{4} \sum_{m=1}^{N_g} (\omega \vec{f}_{num} \cdot \vec{n} \Delta l)_{i,j,l,m} - \frac{c_h^2}{c_p^2} \overline{\psi}_{i,j}, \tag{30}$$

where $\vec{f}_{num}$ is the numerical flux function for Eq. (9). This is a high-order discretization of Eq. (9) as long as the fluxes are computed with high-order accuracy, relying on high-order polynomial reconstruction.

An advantage of the operator splitting approach is that exact analytical integration of Eq. (28) does not impose an additional stability constraint on the time step. In contrast, integrating the source term numerically in a coupled fashion as part of the hyperbolic update may incur an additional source term time step constraint of the type $\Delta t \leqslant C_S \Delta t_S$, with $\Delta t_S = 2 c_p^2 / c_h^2$ and C_S a constant of $O(1)$ related to the time integration scheme. (For example, C_S would be one for Forward Euler time inte-

gration.) This should be compared with the hyperbolic time step constraint of the type $\Delta t \leqslant C_H \Delta t_H$, with C_H the CFL number of the scheme and $\Delta t_H = \Delta x/c_h$, where Δx is a measure of the grid spacing. Recalling the ratio of the diffusive and advective time scales, $\bar{\alpha} = \Delta x \, c_h / c_p^2$ [32,33], and assuming that $C_S \approx C_H$, it can be seen that the ratio of the hyperbolic and source term time step limits is approximately given by $\Delta t_H/\Delta t_S = \bar{\alpha}/2$. It follows that, for example, for the choices of $\bar{\alpha}$ advocated in [32,33], namely, $\bar{\alpha} \in [0, 1]$, the source term time step constraint would normally be less stringent than the hyperbolic time step constraint. Also, for constant c_h and c_p, the hyperbolic time step constraint becomes increasingly dominant as the grid is refined. It also follows that setting $\bar{\alpha}$ to a constant for a sequence of grid sizes implies that the ratio between the hyperbolic and source term time step limits remains the same on those grids, which may be an advantage if the source term is integrated numerically as part of the hyperbolic step and one wants to make sure that the hyperbolic time step restriction dominates the source term time step restriction on all grids in the sequence.

While an operator splitting approach may in principle reduce the order of accuracy, depending on the type of the equation, it is often observed that it does not reduce accuracy in practice even if the formal order of accuracy is reduced [48]. We have done extensive numerical comparisons of the two mechanisms for integrating the GLM source term in our code and have not found any sign of reduced accuracy or reduced convergence order for the operator splitting approach. Further investigation revealed that, in the case of GLM-MHD with mixed hyperbolic-parabolic correction, it can be shown formally that the splitting error vanishes, see Appendix A.

On a related note, we have also confirmed in numerical tests for smooth flows that employing low-order (e.g., linear) reconstruction for ψ while using reconstruction with degree-three polynomials for the physical variables does not lead to convergence degradation: fourth-order accuracy is maintained. This can be explained by relying on similar arguments as those used in Appendix A to show that the operator splitting error vanishes: due to the fact that the exact solution of $\psi(x, t)$ is the zero function for smooth flow, all constants in the Taylor series expansion of the exact solution for ψ vanish, which implies that discretization of ψ does not introduce truncation errors and high-order reconstruction is not required for accurately approximating ψ (which converges to the zero function). Note, however, that during the convergence process the error in ψ is nonzero: it is generated by the truncation error in the components of the magnetic field, and is of the same order of magnitude. Low-order reconstruction of ψ may in principle lead to some computational savings, but it may make implementation somewhat more complex since the reconstruction process for ψ is then different from the other variables.

In our implementation we have several options for integrating the GLM source term, and for choosing the order of reconstruction for ψ and the value of c_p. In the numerical tests presented in the next section we choose the following options. We reconstruct ψ with the same polynomial order as the physical variables, and we integrate the source term numerically as part of the hyperbolic update. For setting c_p, we followed [6] and set c_p by fixing the constant $c_r = c_p^2/c_h$ to a value of 0.18. We have verified that, for this choice, the hyperbolic time step restriction was dominant for all problems and grids we considered.

3.5. High-order accuracy at curved boundaries

In our CENO MHD implementation, two general mechanisms are available to prescribe boundary conditions. The first mechanism uses ghost cells. Every grid block in our hierarchical block-adaptive body-fitted quadrilateral grid framework is equipped with three or four layers of ghost cells. In the numerical results to be presented in Section 4, we compare second-order results with fourth-order results. The second-order simulations employ three layers of ghost cells for each block, and the fourth-order results employ four layers of ghost cells for each block. All blocks have the same size, and the parallelization strategy distributes blocks over parallel message passing interface (MPI) processes [49,50] as uniformly as possible (with typically multiple blocks per MPI process and one MPI process per CPU core), resulting in adequate load balancing. The ghost cells enable the message passing that parallelizes the code. They are also used in the adaptivity mechanism to transfer information between coarse and fine blocks, as is explained in Section 4.3. Note that the number of ghost cell layers is one greater than the number required to enable reconstruction in the first ghost cell layer; this additional ghost cell layer is necessary for computing the smoothness indicator in the first layer of ghost cells (which determines whether the high-order or low-order reconstruction is used there) [2]. The ghost cells can also be used to impose boundary conditions at the domain boundaries in standard ways. All second-order simulations use ghost cells to impose boundary conditions. Ghost cells are also used to impose boundary conditions for our fourth-order tests in certain cases, for example in the case of periodic boundary conditions. However, for high-order accuracy near curved boundaries, a more accurate second mechanism for boundary conditions is needed.

The second boundary condition mechanism relies on accurate representation of the curved boundaries with high-order piecewise polynomial splines. It uses one-sided stencils near boundaries that only contain cells within the computational domain, and it imposes additional constraints on the least-squares reconstruction problem at the Gauss points [51]. It is also important to compute the geometric data such as cell areas, centroid locations, etc. to the same order of accuracy as that of the interior scheme [2,4]. We represent curved boundaries with piecewise polynomial splines of an order consistent with that of the FV numerical scheme, which allows us to locate Gauss quadrature points and compute flux integrals with high accuracy. One-sided reconstruction stencils are used for the first and second layer of cells in the computational domain at the boundaries, and constraints are added to the least-squares reconstruction of the cells in the first layer to accurately impose certain types of boundary conditions on the curved boundaries at the Gauss points. When ghost cells are not used, the one-

sided reconstructed values at the Gauss points are directly plugged into the exact MHD flux functions to obtain the numerical flux. For variables to be left free at the boundaries (extrapolation from the computational domain), no additional constraints are necessary. For variables to be imposed at the boundaries, the appropriate constraints are added at the Gauss points used in the flux integration.

More generally, our framework accepts Robin boundary conditions (which consist of linear combinations of Dirichlet and Neumann conditions), and it also accepts linear relations among variables which form a coupling constraint for a set of reconstructed variables [2]. These coupling constraints can be used to impose wall conditions at curved boundaries with high accuracy. This has been explained for Euler flows in [1,2], and we extend it here to perfectly conducting walls in MHD problems. For perfectly conducting walls, we impose that $\vec{B} \cdot \vec{n} = 0$ and $\vec{v} \cdot \vec{n} = 0$ in each Gauss quadrature point. Let (n_x^g, n_y^g) be the normal vector in Gauss quadrature point g of cell (i,j), and let (x_g, y_g) be its coordinates. Then, using the polynomial expansion of Eq. (13), the conditions $\vec{B} \cdot \vec{n} = 0$ and $\vec{v} \cdot \vec{n} = 0$ at the Gauss point can be expressed as

$$\sum_{\substack{p_1=0 \\ (p_1+p_2 \leqslant k)}}^{k} \sum_{p_2=0}^{k} (x_g - \overline{x}_{i,j})^{p_1} (y_g - \overline{y}_{i,j})^{p_2} n_x^g (D_{p_1 p_2}^k)_{B_x} + \sum_{\substack{p_1=0 \\ (p_1+p_2 \leqslant k)}}^{k} \sum_{p_2=0}^{k} (x_g - \overline{x}_{i,j})^{p_1} (y_g - \overline{y}_{i,j})^{p_2} n_y^g (D_{p_1 p_2}^k)_{B_y} = 0, \tag{31}$$

and

$$\sum_{\substack{p_1=0 \\ (p_1+p_2 \leqslant k)}}^{k} \sum_{p_2=0}^{k} (x_g - \overline{x}_{i,j})^{p_1} (y_g - \overline{y}_{i,j})^{p_2} n_x^g (D_{p_1 p_2}^k)_u + \sum_{\substack{p_1=0 \\ (p_1+p_2 \leqslant k)}}^{k} \sum_{p_2=0}^{k} (x_g - \overline{x}_{i,j})^{p_1} (y_g - \overline{y}_{i,j})^{p_2} n_y^g (D_{p_1 p_2}^k)_v = 0 \tag{32}$$

with $(D_{p_1 p_2}^k)_{B_x}$ the polynomial coefficients for the B_x magnetic field component, and similar for the B_y, u and v vector components. To impose $\vec{B} \cdot \vec{n} = 0$ in the reconstruction, we solve the least-squares reconstruction problems for the B_x and B_y polynomials together, with the additional constraints of Eq. (31) for each Gauss point. Similarly, the least-squares reconstruction problems for u and v are solved together to impose $\vec{v} \cdot \vec{n} = 0$, with the additional constraints of Eq. (32). For full implementation details, see [2].

4. Numerical results

In this section we present numerical results that demonstrate high-order convergence for smooth flows and robustness against oscillations for flows with shocks. We present four continuous test problems followed by two problems with discontinuities, including a new MHD extension of the well-known Shu–Osher test problem [38]. Finally, we demonstrate the dynamic AMR capabilities of our implementation using adaptive time-dependent simulations of the Orszag–Tang vortex problem [39] with high-order accuracy and unprecedented effective resolution. Note that the proposed finite-volume method has been implemented in parallel using the C++ programming language and MPI [49,50] and closely follows the techniques and implementations described by Groth and co-workers [17–19,21], with the necessary extensions to high-order accuracy as described by Ivan and Groth [1,4]. All of the numerical results presented below were obtained using a parallel computing cluster. In particular, the computations were performed on a cluster consisting of 3,780 Intel Xeon E5540 (2.53 GHz) nodes with 16 GB RAM per node. The cluster nodes were interconnected with a high-speed, low-latency, Infiniband switched network. We used from 16 up to 512 cores, depending on the problem under consideration. Please refer to the thesis of Ivan [4] for an assessment of the parallel performance of the high-order finite-volume scheme with AMR, demonstrating the near-ideal parallel efficiency and scalability of the parallel implementation.

4.1. Continuous problems

We first present two smooth test problems on Cartesian grids, which are the rotated Alfvén problem from [10], and the magnetostatic problem from [24]. We then present two continuous test problems on body-fitted multi-block structured grids with non-rectangular cells and curved boundaries: the rotating radial outflow problem and the expanding tube problem from [52].

To quantify the accuracy of the numerical solution, the errors are measured in the L_1, L_2, and L_∞ norms:

$$L_1 = |E|_1 = \frac{1}{A_T} \sum_{i,j} \iint_{\mathcal{A}_{i,j}} \left| u_{i,j}^k(x,y) - u_{exact}(x,y) \right| dA, \tag{33}$$

$$L_2 = |E|_2 = \sqrt{\frac{1}{A_T} \sum_{i,j} \iint_{\mathcal{A}_{i,j}} \left[u_{i,j}^k(x,y) - u_{exact}(x,y) \right]^2 dA}, \tag{34}$$

$$L_\infty = |E|_\infty = \max_{i,j} \left(\frac{1}{A_{i,j}} \iint_{\mathcal{A}_{i,j}} \left| u_{i,j}^k(x,y) - u_{exact}(x,y) \right| dA \right), \tag{35}$$

where A_T is the total area of the computational domain. The integrals are evaluated with high-order accurate Gaussian quadrature, see [2] for details. In most of our numerical tests, we compare convergence for four numerical methods: fourth-order

CENO, fourth-order unlimited k-exact reconstruction, second-order CENO, and second-order unlimited k-exact reconstruction. The CENO methods switch between the k-exact reconstruction and the limited piecewise linear reconstruction based on the smoothness indicator. We use a smoothness indicator cut-off value $S_C = 800$ except where noted.

4.1.1. Rotated Alfvén travelling wave propagation

The circularized Alfvén wave problem from [10] represents analytical solutions of the MHD equations for arbitrary amplitudes. The wave propagates with an angle of $\alpha = 30°$ with respect to a Cartesian grid, and assumes the initial conditions (as in [10]): $\rho = 1$, $v_\parallel = 0$, $p = 0.1$, $B_\parallel = 1$, $v_\perp = B_\perp = 0.1 \sin(2\pi(x\cos(\alpha) + y\sin(\alpha)))$, and $v_z = B_z = 0.1 \cos(2\pi(x\cos(\alpha) + y\sin(\alpha)))$. The parallel velocity, $v_\parallel$, is set to zero, which corresponds to the travelling wave test case. The perpendicular and parallel directions are defined with respect to the direction of wave propagation. These initial conditions give an Alfvén speed of 1, which corresponds to a transit period of 1. The computational domain is set to be periodic (using ghost cells), with ranges $[0, 1/\cos(\alpha)]$ for x, and $[0, 1/\sin(\alpha)]$ for y. As in [10], the number of cells in the x-direction is equal to the number of cells in the y-direction, which corresponds to a ratio of $1/\sqrt{3}$ between Δx and Δy. The simulations are run for 5 transit periods (or up to $t = 5$). Density and other scalar variables are expected to be constant throughout the simulation since they are not perturbed by the Alfvén wave, so only the accuracy of the $\vec{v}$ and $\vec{B}$ fields were assessed for convergence studies.

As can be seen from Fig. 1, the expected order of convergence is achieved for the x-direction magnetic field, at least in the asymptotic limit. For the sake of brevity, only the results of the x-direction magnetic field are shown, but the other variables behave in a similar manner. The effect of the CENO monotonicity-preserving reconstruction switching procedure ($S_C = 800$) can be seen: a "transition" regime occurs where the mesh is not fine enough and the smooth flow features are not sufficiently resolved. This transition regime does not occur for the fourth-order method because it sufficiently resolves the flow already with low resolution, see also [2,4]. The high-order scheme represents significant savings in the number of computational cells required for some specific level of accuracy: a 64-by-64 grid resolution was sufficient for the fourth-order scheme to obtain a smaller error than the limited second-order scheme on a 384-by-384 grid.

4.1.2. Two-dimensional magnetostatic problem

We next consider the magnetostatic problem from [24]. The exact solution of this stationary problem is known: $\rho = 1$, $v_x = 0$, $v_y = 0$, $v_z = 0$, $B_x = -\cos(\pi x)e^{-\pi y}$, $B_y = \sin(\pi x)e^{-\pi y}$, $B_z = 0$, $p = 19.84(\gamma - 1)$, $\psi = 0$. Following Warburton et al. [24], this exact solution is used as the initial condition for the simulation, and the error at steady-state is a measure of the deviation of the numerical solution from the exact solution. The second-order methods use ghost cells to impose boundary conditions: $\vec{v}$ and $\vec{B}$ are imposed in the ghost cells (accurate average values of the exact solution, obtained by numerical quadrature), and ρ, p and ψ are extrapolated to the ghost cells (the average values are linearly extrapolated). The fourth-order methods use one-sided reconstruction, with the exact values of $\vec{v}$ and $\vec{B}$ imposed at the Gauss points using constraints,
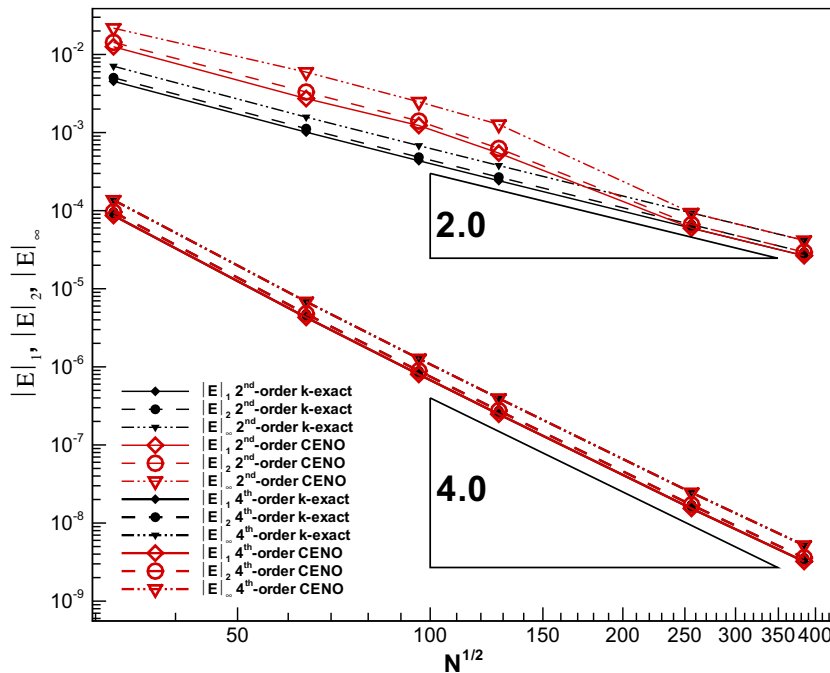

Fig. 1. The L_1-, L_2-, and L_∞-norm errors for the magnetic field in the x-direction for the rotated Alfvén wave problem, calculated at $t = 5$ (five transit periods). N is the total number of grid cells. The solution is compared with the initial conditions to compute the error. The error converges to zero with the expected order of accuracy in the asymptotic limit. A transition region is observed for the second-order CENO scheme, consistent with the findings of Ivan and Groth [2,4].
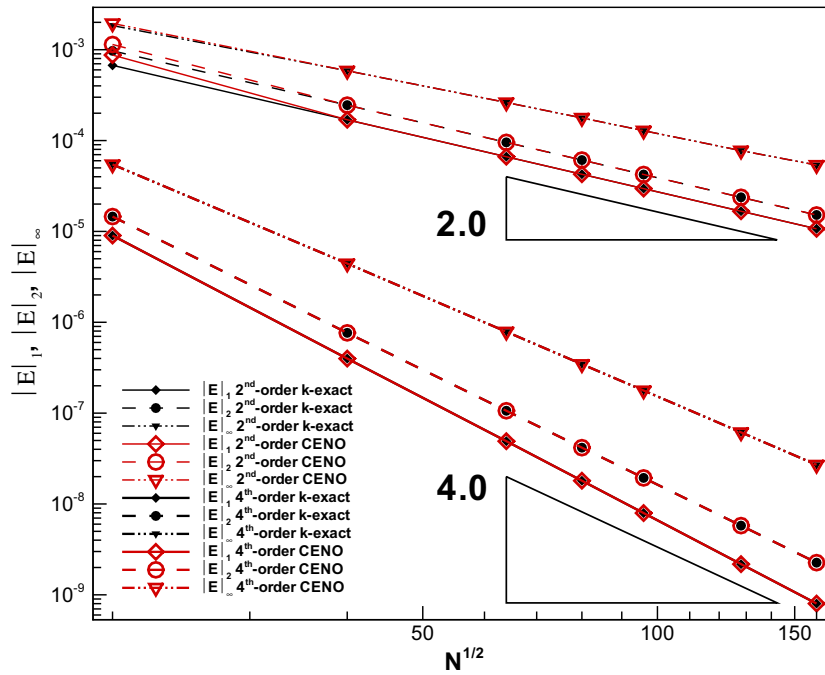
Fig. 2. The L_1-, L_2-, and L_∞-norm errors for the magnetic field in the x-direction for the magnetostatic problem. The steady-state solution is compared with the initial conditions to compute the error. The error converges to zero with the expected order of accuracy in the asymptotic limit.

and ρ, p and ψ are left free at the Gauss points. Fig. 2 shows how the error norm of B_x converges to zero as a function of grid size with the expected order. The fourth-order scheme requires much fewer computational cells to achieve a specified level of error (in this case, the error can differ by as much as 4 orders of magnitude for the same number of cells).

4.1.3. Superfast rotating outflow from a cylinder

We next consider the rotated outflow problem from [52] on a body-fitted structured grid with non-rectangular cells and curved boundaries. While the exact analytical solution is not available, several theoretical flow invariants are available [52], against which the corresponding computed quantities can be compared. We measure error in the entropy, s, and the radial magnetic flux.

The problem is defined on a domain between two concentric circles, and superfast inflow conditions (normal velocity faster than the fast magnetosonic wave speed) are imposed at the inner circle. The domain goes from $r = 1$ to $r = 6$, and the inflow conditions imposed at the $r = 1$ boundary are $\rho = 1$, $p = 1$, $v_r = 3$, $v_\theta = 1$, and $B_r = 1$. The second-order methods use ghost cells to impose boundary conditions. At the inner boundary (inflow), ρ, p, $\vec{v}$ and $\vec{B}$ are imposed in the ghost cells using linear extrapolation (to impose the desired values exactly at the domain boundary), and ψ is extrapolated linearly from the interior of the domain. At the outer boundary (outflow), ρ, p, $\vec{v}$ and $\vec{B}$ are extrapolated linearly, and ψ is set to zero using linear interpolation (to impose the desired value exactly at the domain boundary). The fourth-order methods use high-order piecewise polynomial spline representation of the curved boundaries, combined with one-sided reconstruction and constraints. At the inner boundary (inflow), ρ, p, $\vec{v}$ and $\vec{B}$ are imposed by constraints at the Gauss points, and ψ is left free. At the outer boundary (outflow), ψ is set to zero by constraints at the Gauss points, and ρ, p, $\vec{v}$ and $\vec{B}$ are left free.

The steady-state solution of the rotated outflow problem obtained with the fourth-order CENO scheme on a mesh with 80-by-80 cells can be seen in Fig. 3. The magnetic field lines are not aligned with the streamlines. The solutions obtained with second- and fourth-order CENO schemes are compared in Fig. 4.

It can be seen that the errors converge to zero with the expected order of accuracy. For this problem, the second-order scheme has not reached the asymptotic regime beyond the transition region yet for the resolutions we tested, and the second-order CENO error remains above the unlimited second-order error due to ongoing switching from unlimited k-exact reconstruction to limited second-order reconstruction, especially for the error in the radial magnetic field as seen in Fig. 4(b). This is possibly due to the inability of the piecewise linear function to capture the curvature of the boundaries properly, so the switching procedure continues to see some cells close to the boundaries as under-resolved, thus limiting the reconstruction functions at these places and affecting the magnitude of the error. In contrast, for fourth-order CENO and the unlimited k-exact schemes produce the same error for resolutions above 80-by-80. It is clear that our approach can handle curved boundaries with high-order accuracy, and the fourth-order method requires significantly fewer cells than the second-order method to obtain a given error level.

In Fig. 5 we illustrate that the high-order CENO-GLM approach can naturally handle resolution changes on block-adaptive grids and the errors introduced by AMR restriction and interpolation procedures. We have performed a grid convergence study on a series of grids where there is a change in resolution between some blocks (corresponding to one level of refine-
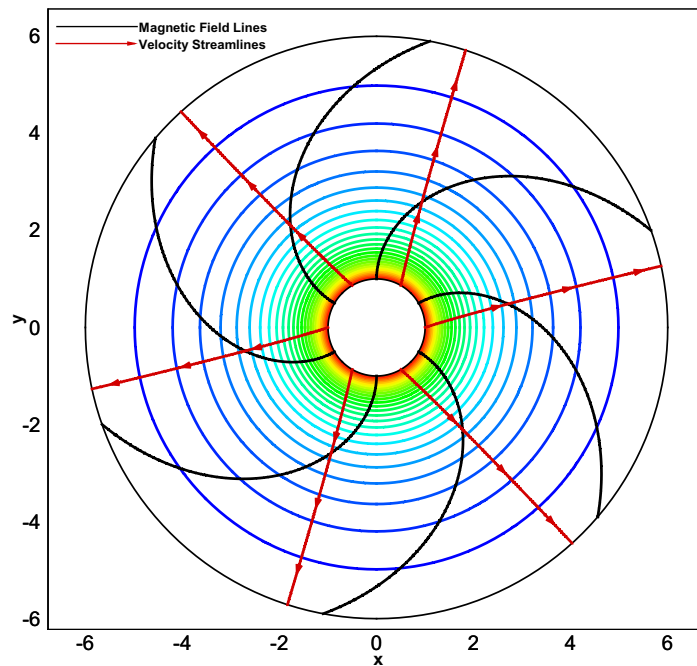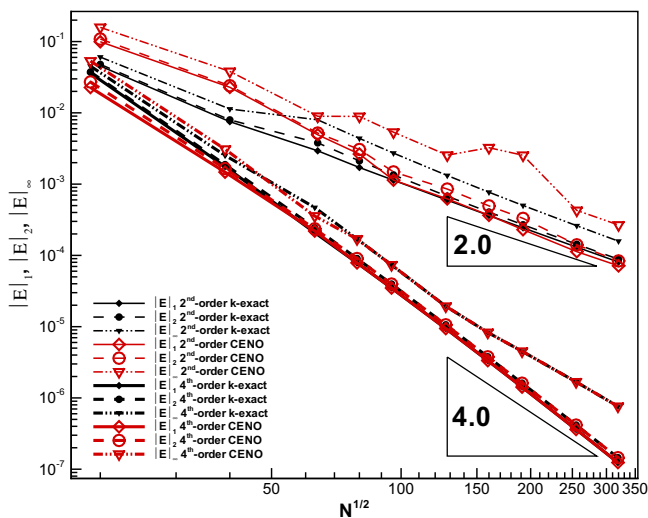
Fig. 3. Density contour lines, magnetic field lines, and streamlines for the rotated outflow problem, obtained on a mesh with 80-by-80 cells. The magnetic field lines and the streamlines are not aligned. The flow is smooth throughout the entire domain, which enables high-order convergence rates. The density contour lines are equally spaced in the range (0.17,0.97).

(a) The L_1-, L_2-, and L_∞-norm errors for entropy, which is constant in the domain.
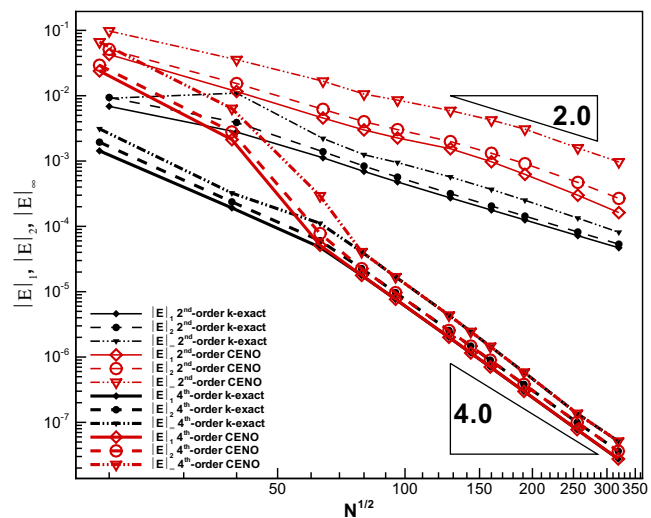
(b) The L_1-, L_2-, and L_∞-norm errors for the radial magnetic flux. While there is no analytical solution for the full magnetic field, the radial component of the magnetic field can be determined, due to the conservation of the radial magnetic flux.

Fig. 4. Convergence study for the rotated outflow problem using both unlimited k-exact reconstruction (black lines) and CENO with $S_C = 800$ (red lines). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

ment). In each successive point of the convergence study, all blocks in the grid of Fig. 5(b) are refined by dividing them into four blocks. The convergence plot of Fig. 5(a) shows that fourth-order convergence is automatically maintained by the CENO-GLM approach, and magnetic field lines remain smooth at the jump in grid resolution.

4.1.4. Expanding tube problem

The expanding tube problem from [52] is another continuous problem that uses a body-fitted structured grid with curved boundaries. It models plasma flow in an expanding tube, which gives rise to an MHD solution that contains a rarefaction wave with a weak discontinuity at the edge of the rarefaction (see Fig. 6). Across the weak discontinuity, the first spatial
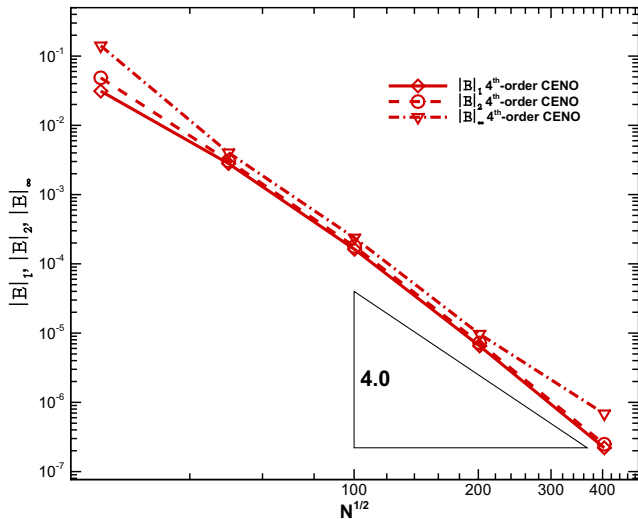
(a) The L_1-, L_2-, and L_∞-norm errors for the radial magnetic flux obtained with the fourth-order CENO scheme on a grid with resolution change between blocks (right panel). Fourth-order accuracy is maintained.
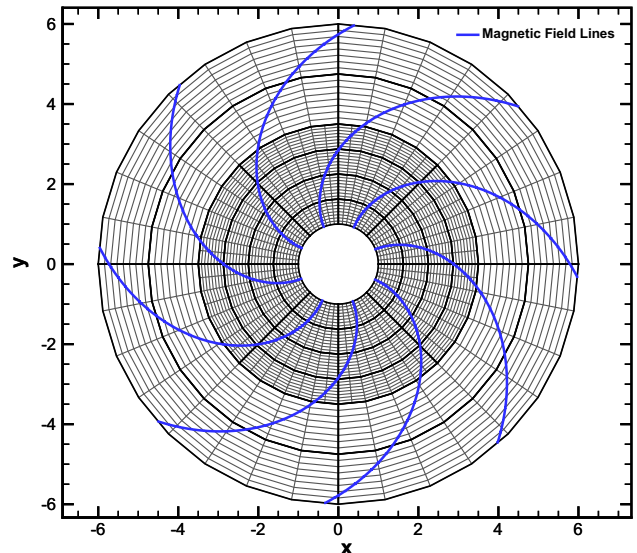

(b) The grid with resolution change used for the convergence study in the left panel. This grid has 8 coarse blocks and 32 fine blocks, each with 8 × 8 cells. In each successive point of the convergence study, all blocks in the grid are refined by dividing them into four blocks. The magnetic field lines of the solution on this grid are plotted in blue. The smoothness of the magnetic field lines is not adversely affected by the jump in grid resolution.

Fig. 5. Convergence study for the rotated outflow problem on a grid with change in grid resolution between blocks, illustrating that the high-order CENO-GLM approach naturally handles resolution changes on block-adaptive grids.

derivative of the flow variables is discontinuous. Therefore, even if fourth-order reconstruction accuracy is targeted (by employing degree-3 polynomial functions to reconstruct the solution), the solution accuracy is still limited to second-order near the weak discontinuity.

The flow is simulated on a domain with $x \in [0, 1]$, and $y \in [y_0(x), 1]$, where $y_0(x) = \cos(\frac{\pi}{4}(x - 0.3)) - 1$ for $x \in [0.3, 1]$, and zero elsewhere in the domain. The lower wall starts to curve at $x = 0.3$, giving rise to a rarefaction wave downstream of the weak discontinuity. The boundary curve follows a cosine function rather than a straight line to avoid a geometrical singularity in the boundary, which results in the rarefaction wave not converging to a single point, as can be seen from Fig. 6(a) [52]. At the $x = 0$ boundary, a uniform inflow with the following conditions is imposed: $\rho = 1$, $p = 1$, $v_x = 8$, and $B_x = 4$. These initial conditions correspond to superfast horizontal inflow conditions, with an acoustic Mach number $v_x/c = 8\sqrt{3/5}$, and Alfvénic Mach number $v_x/c_{A_x} = 2$. The second-order methods use ghost cells to impose boundary conditions. The superfast inflow and outflow boundary conditions at the left and right boundaries, respectively, are implemented as for the rotating outflow test problem of Section 4.1.3. For the top and bottom boundaries, standard wall boundary conditions are implemented that symmetrically copy ρ, p and ψ to the ghost cells, and mirror $\vec{v}$ and $\vec{B}$ with respect to the wall. The fourth-order methods use high-order piecewise polynomial spline representation of the wall boundaries, combined with one-sided reconstruction and constraints. For the top and bottom wall boundaries, ρ, p, ψ and the tangential components of $\vec{v}$ and $\vec{B}$ are left free, while the normal components of $\vec{v}$ and $\vec{B}$ are set to zero at the Gauss points using constraints. The high-order outflow boundary condition is handled as in the rotating outflow test problem of Section 4.1.3. We simply use ghost cells for the high-order inflow boundary condition, since the flow remains uniform close to the inflow boundary and ρ, p, $\vec{v}$ and $\vec{B}$ can just be imposed in all ghost cell layers, while ψ can be extrapolated linearly.

To assess the accuracy of the solution, entropy, which is one of the invariants for this flow, is measured. Fig. 6(b) shows convergence analysis of the entropy error. As can be seen in this figure, second-order accuracy is achieved for the L_1-norm error of the entropy for both the second-order and the fourth-order accurate methods. While Fig. 6(b) illustrates how the weak discontinuity in the solution limits the order of accuracy, reduction in the total error is still observed when higher-order polynomial functions (the fourth-order method) are used to represent the solution. It is interesting to note that the log error of the CENO solution decreases linearly, whereas some zigzagging is present in the unlimited k-exact error plot. This can be explained by the fact that the flow is not fully smooth, and the weak discontinuity that exists can potentially generate spurious oscillations when monotonicity is not enforced, though the level at which these oscillations occur is apparently much smaller than the solution variation. Note also that, even at the highest attempted resolution, the convergence plots of the CENO error do not converge to those of unlimited k-exact reconstruction (as was the case for the other test cases), implying that, due to the weak discontinuity, reconstruction switching is always performed for at least a few cells.
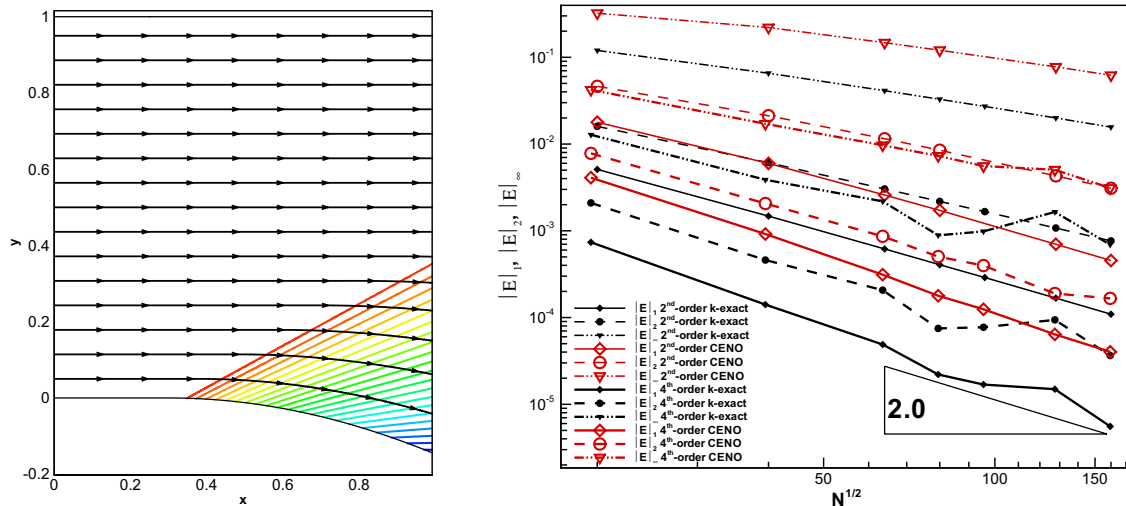
(a) The expanding tube flow, solved on a 160-by-160 grid with the fourth-order CENO scheme. The contour lines are equally spaced in the range (0.45,0.95) (21 contours).

(b) Convergence study for the expanding tube problem. The entropy is compared with the entropy at the inflow to compute the error, and nearly second-order accuracy for the L_1-norm error is observed for both the second- and fourth-order methods.

Fig. 6. Expanding tube flow: density contour lines and entropy convergence study. The error converges with at most second-order accuracy, due to the non-existence of higher-order derivatives across the weak discontinuity. Convergence study is performed for entropy using both unlimited k-exact reconstruction (black lines) and CENO with $S_c = 800$ (red lines). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)
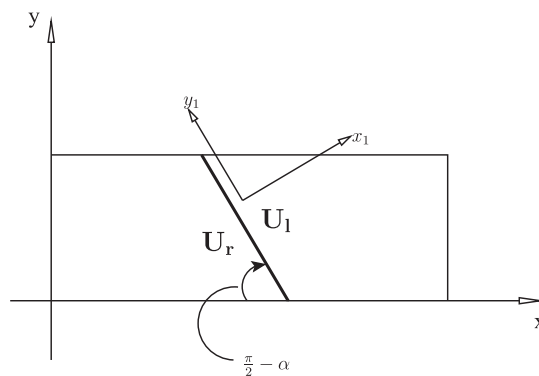
Fig. 7. Setup for the rotated one-dimensional problems with discontinuities. The discontinuity is rotated 45° counterclockwise with respect to the x-axis. The solution thus exhibits a translational symmetry in the y_1-direction.

4.2. Problems with discontinuities

4.2.1. Rotated Brio–Wu shock tube problem

The Brio–Wu shock tube problem [53] is a standard test case to demonstrate the capability of a numerical MHD scheme to handle discontinuities. The initial conditions are given by

$$(\rho, v_\perp, v_\parallel, v_z, B_\perp, B_\parallel, B_z, p, \psi) = \begin{cases} (1, 0, 0, 0, 0.75, 1, 0, 1, 0) & \text{for } x_1 < 0, \\ (0.125, 0, 0, 0, 0.75, -1, 0, 0.1, 0) & \text{for } x_1 > 0, \end{cases} \tag{36}$$

with $\gamma = 2$. Here, x_1 is the coordinate variable perpendicular to the shock, given by $x_1 = x \cos \alpha + y \sin \alpha$ with α the angle at which the shock frame of reference is rotated with respect to the x-axis (we choose $\alpha = 45°$). This setup is illustrated in Fig. 7.

Ghost cells are used and constant extrapolation boundary conditions are applied to all boundaries, though the top and the bottom boundaries require that the cells not only be copied to the ghost cells, but also shifted to the left or the right by one cell (similar to Fig. 10 from [10]). It is important to note that, for this boundary condition to work, the ratio between the spacing in the x-direction and the spacing in the y-direction needs to be 1, because otherwise the 45° symmetry would not translate to a (1,1) translational symmetry, see also [10,33,54].
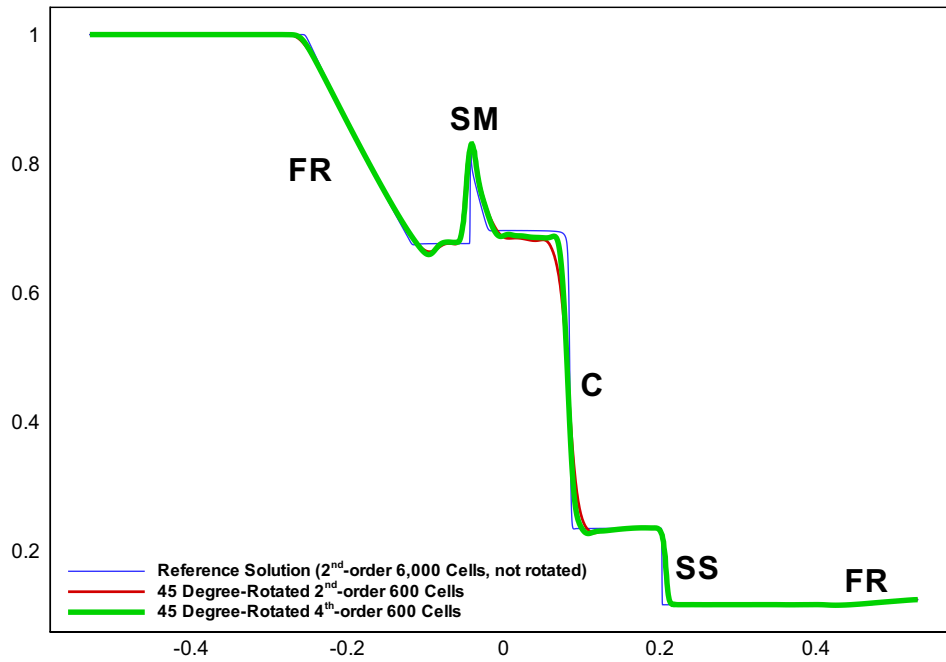
Fig. 8. Comparison of the density solution of the Brio–Wu shock tube problem at $t = 0.1414$, rotated at 45°. Here, FR denotes fast rarefaction, SM slow compound wave, C contact discontinuity, and SS slow shock. A cutoff value of $s_C = 8000$, is chosen for these simulations.

The Brio–Wu problem gives rise to several types of waves and shocks: fast rarefaction waves, a contact discontinuity, a slow compound wave, and a slow shock [53]. Simulations were performed for the rotated cases using 600 cells in the x-direction, and 4 cells in the y-direction. The density plot is shown in Fig. 8, and illustrates that our method is robust with respect to spurious oscillations. The fourth-order solution has slightly sharper features than the second-order solution. All of the important wave features are captured well without spurious oscillations, except for a slight undershoot between the fast rarefaction (FR) and the slow compound wave (SM) (which is also observed in other work on high-order MHD schemes [27,33]).

4.2.2. MHD Extension to Shu–Osher shock tube problem

The shock tube problem proposed by Shu and Osher [38] is commonly used to test the ability of high-order numerical schemes to resolve small-scale flow features in the presence of shocks. A sinusoidal density perturbation is added downstream of a purely advecting supersonic shock wave. The interaction of the shock wave with the sinusoidal part of the density field gives rise to fast oscillations and complex flow features downstream to the shock. The Shu–Osher shock tube problem provides an excellent testbed to highlight the benefits of the improved accuracy of high-order numerical schemes, while at the same time the presence of the shock puts the robustness and stability of the schemes to test. In what follows, we develop a new MHD version of the Shu–Osher shock tube problem.

Consider the Rankine–Hugoniot conditions in the shock frame:

$$\mathbf{F}(\mathbf{U_l}) = \mathbf{F}(\mathbf{U_r}), \tag{37}$$

where $\mathbf{U_r}$ and $\mathbf{U_l}$ denote the state vectors of the right and left state, respectively, and $\mathbf{F}(\mathbf{U_i})$ denotes the flux evaluated at state i. We choose the following initial conditions that satisfy the Rankine–Hugoniot condition:

$$(\rho, u_\perp, u_\parallel, u_z, B_\perp, B_\parallel, B_z, p, \psi) = \begin{cases} (1, 0, 0, 0, 1, 1, 0, 1, 0) & \text{for } x < 4, \\ (3.5, 5.8846, 1.1198, 0, 1, 3.6359, 0, 42.0267, 0) & \text{for } x > 4. \end{cases} \tag{38}$$

The numbers in Eq. (38) were obtained by choosing the left state and numerically solving the MHD Rankine–Hugoniot conditions for a fast shock moving to the left with speed 8.2385. The numerical solution values were rounded to four decimal digits (which is sufficiently accurate for the numerical tests).

Eq. (38) represents the unperturbed portion of our newly proposed MHD version of the Shu–Osher shock tube problem. Similar to the Shu–Osher problem, sinusoidal perturbation is added to the downstream part of the density field (because the shock and the flow travel to the left, the sinusoidal perturbation is added to ρ_l). The initial density function is then chosen as

$$\rho_l = 1 + 0.2 \sin(5x), \quad \rho_r = 3.5 \tag{39}$$

and all the other variables are kept as given in Eq. (38).

As in the case of the rotated Brio–Wu problem, the initial condition given by Eq. (38) and (39) has been applied in the rotated frame of reference $x_1 - y_1$ (see Fig. 7). The boundary conditions for our simulation of this problem are as explained in Section 4.2.1. The left and right boundaries are taken sufficiently far from the initial discontinuity, such that they do not
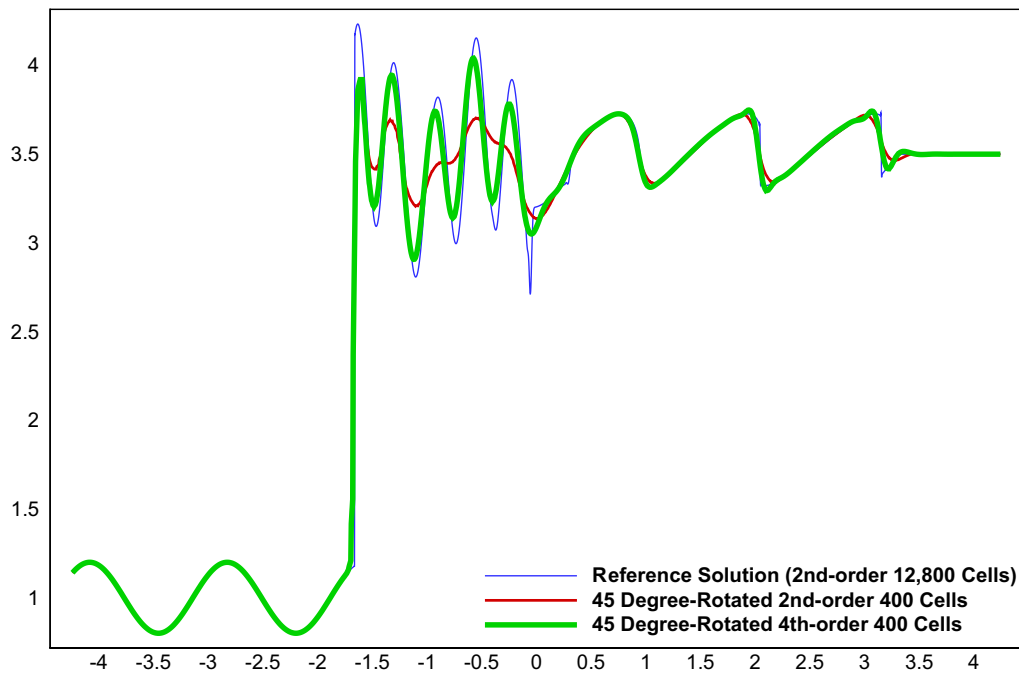
Fig. 9. Comparison of the density solution of the MHD Shu–Osher problem at $t = 0.6906$, rotated at 45°. As can be observed from the figure, the fourth-order method produces results that are much closer to the non-rotated reference result in the highly oscillatory region, illustrating the benefits of high-order accuracy. A cutoff value of $S_c = 80$ is used for these simulations.

influence the solution. The comparison of the density profiles between the different methods is shown in Fig. 9. The benefit of the high-order method is clear: using the same number of cells, the fourth-order method captures the small-scale flow features much better than the second-order method. For all simulations performed for this section, no stability or overshoot problem were observed, which indicates that the monotonicity-preserving mechanism is doing its job properly to ensure that the method is stable in the presence of discontinuities.

4.3. Application of CENO with dynamic adaptive mesh refinement: Orszag–Tang vortex problem

In this section we demonstrate the dynamic AMR capabilities of our implementation using adaptive time-dependent simulations of the Orszag–Tang vortex problem [27,30,39] with high-order accuracy and unprecedented effective resolution.

We have implemented our new high-order MHD scheme that combines CENO and GLM into a hierarchical quadtree block-based AMR procedure for multi-block body-fitted quadrilateral mesh that is based on the previous work of Groth and co-workers [17–19,21] and is extended to high-order accuracy as in [1,4]. We give a brief summary of the approach, and details are described in [1,2,18]. In our hierarchical quadtree block-based AMR algorithm, mesh adaptation is accomplished by dividing and coarsening appropriate solution blocks. In regions requiring increased cell resolution, a 'parent' block is refined by dividing it into four 'children'. Each of the four quadrants or sectors of a parent block becomes a new block having the same number of cells as the parent, thereby doubling the cell resolution in the region of interest. This process can be reversed in regions that are deemed over-resolved and four children can be coarsened into a single parent block. The mesh refinement is constrained such that the grid resolution changes by at most a factor of two between adjacent blocks, and the minimum resolution is not less than that of the initial mesh. A hierarchical quadtree data structure and additional interconnects between the 'leaves' of the trees are used to keep track of mesh refinement and the connectivity between solution blocks. The hybrid CENO solution reconstruction procedure is used in conjunction with standard multigrid-type restriction and interpolation operators to evaluate the solution on all blocks created by the coarsening and division processes. Interpolation is performed with high-order accuracy by computing reconstructed polynomials for solution variables in each coarse-grid cell and integrating them over the fine-grid children cells to determine the fine-grid cell averages with high-order accuracy (see [2] for details). Restriction and interpolation are performed in such a way that conservation is maintained, but in our CENO-GLM MHD approach no special treatment is required for restricting or interpolating the cell-centred magnetic fields: restriction or interpolation may introduce errors of the order of the discretization error, and they are handled properly by the GLM mechanism for controlling $\nabla \cdot \vec{B}$.

Grid refinement and coarsening are based on the maximum value of the CENO smoothness indicator over each block for the density variable. For each cell, the variable

$$\mathcal{R}_c = e^{-\frac{\max(0,S)}{S_c}} \tag{40}$$

is calculated, where s is the value of the smoothness indicator and s_c is the cutoff value for the smoothness indicator. The range of $\mathcal{R}_c$ is $(0,1)$. The maximum $\mathcal{R}_c^B$ of all $\mathcal{R}_c$ values within a block is computed. In blocks with $\mathcal{R}_c^B$ close to 0, all cells are smooth and resolved, and blocks with $\mathcal{R}_c^B$ close to 1 have cells that are non-smooth or under-resolved. The block-based $\mathcal{R}_c^B$ values are compared with refinement and coarsening thresholds to determine if a block should undergo refinement, or if a group of blocks should be combined for coarsening. Full details on the algorithm followed for coarsening and refinement are given in [18,2]. The refinement/coarsening algorithm is invoked at regular intervals during the simulation to obtain dynamic AMR.

For the Orszag–Tang vortex problem, the same initial conditions and domain as in [27] are used, with $\rho = \gamma^2$, $v_x = -\sin(y)$, $v_y = \sin(x)$, $B_x = -\sin(y)$, $B_y = \sin(2x)$, and $p = \gamma$. The remaining variables (v_z, B_z, and ψ) are initialized to zero. The computational domain is a square with x and y values between 0 and 2π, and periodic boundary conditions (ghost cells are used). The simulation is performed with CENO cutoff tolerance $s_c = 500$. The mesh is refined every 0.025 s up to $t = 1$. For
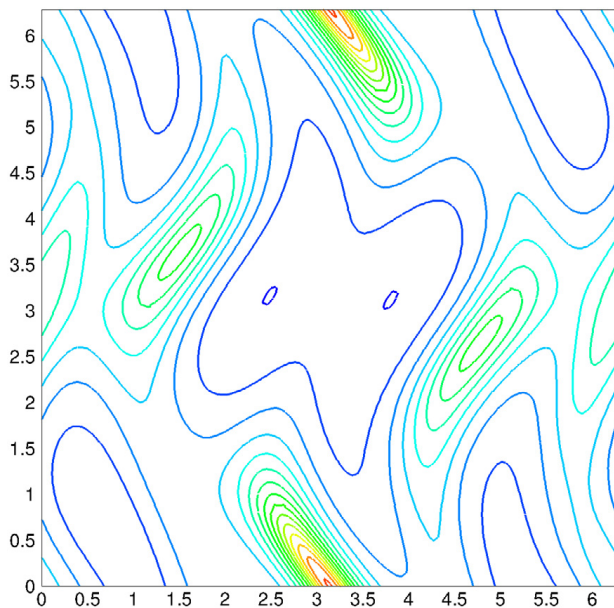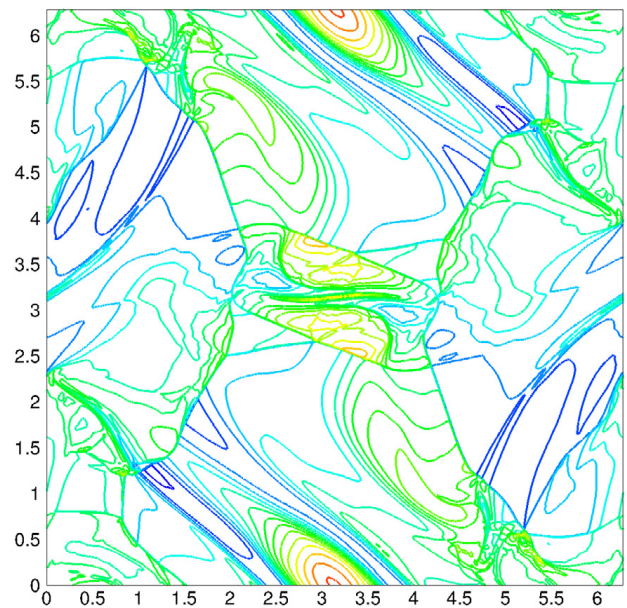

(a) Density solution at $t = 0.5$. The contour lines are equally spaced in the range (2.11,5.82) (15 contours).
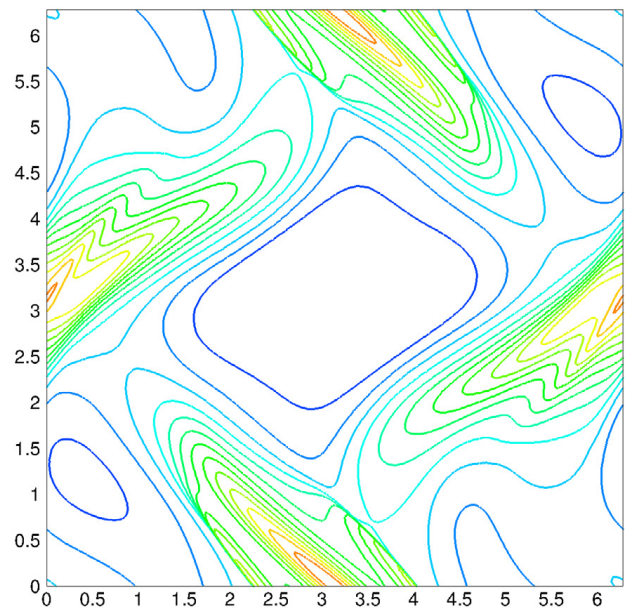
(b) Density solution at $t = 1.0$. The contour lines are equally spaced in the range (1.25,6.9) (15 contours).
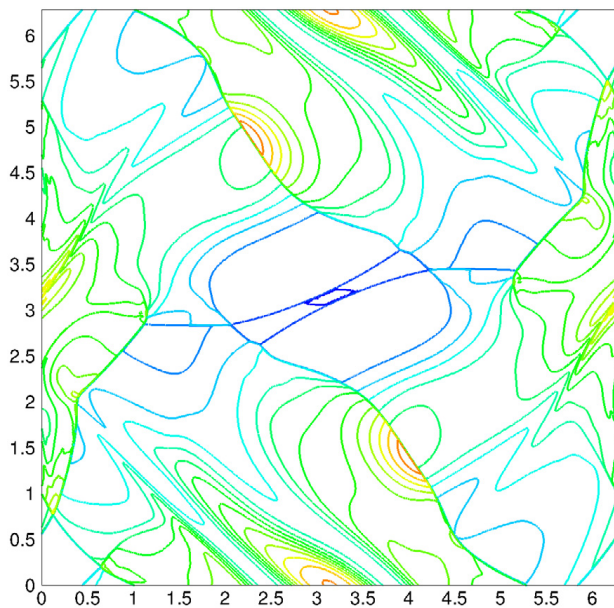
(c) Density solution at $t = 2.0$. The contour lines are equally spaced in the range (0.62,6.41) (15 contours).

(d) Density solution at $t = 3.0$. The contour lines are equally spaced in the range (1.16,6.42) (15 contours).

Fig. 10. The evolution of density for the Orszag–Tang vortex problem at different times: $t = 0.5$, $t = 1.0$, $t = 2.0$, and $t = 3.0$. The ranges for the contour lines shown here are as in [27]. These fourth-order accurate results were obtained using dynamic grid adaption with the meshes shown in Fig. 11.
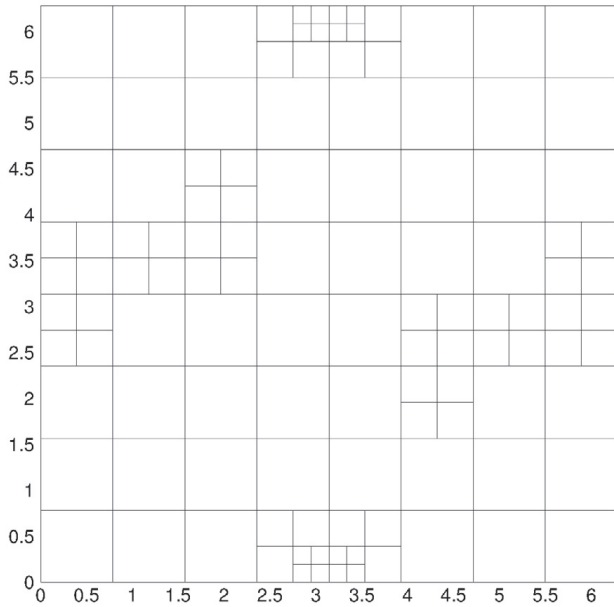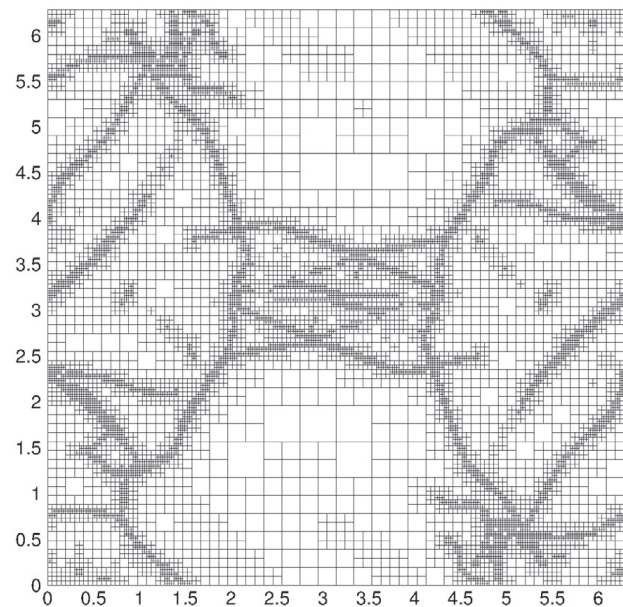
(a) AMR as applied to the Orszag-Tang vortex problem at t = 0.5. At this point, the mesh consists of 118 8-by-8 blocks, or 7,552 cells in total.
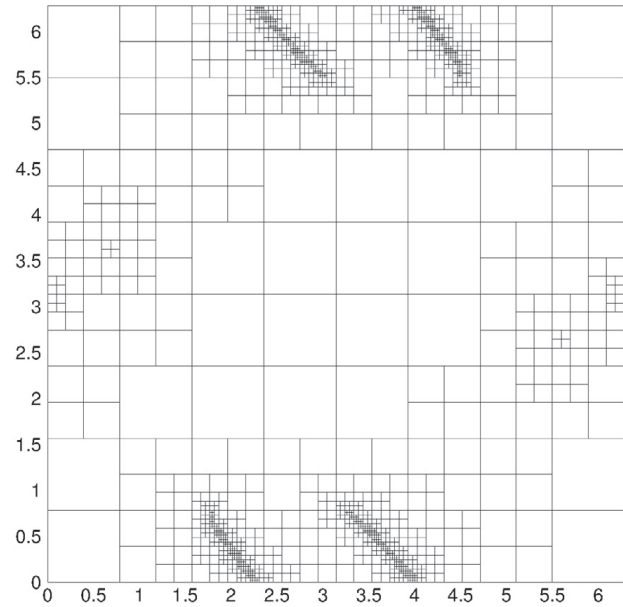
(b) AMR as applied to the Orszag-Tang vortex problem at t = 1.0. At this point, the mesh consists of 1,474 8-by-8 blocks, or 95,810 cells in total.
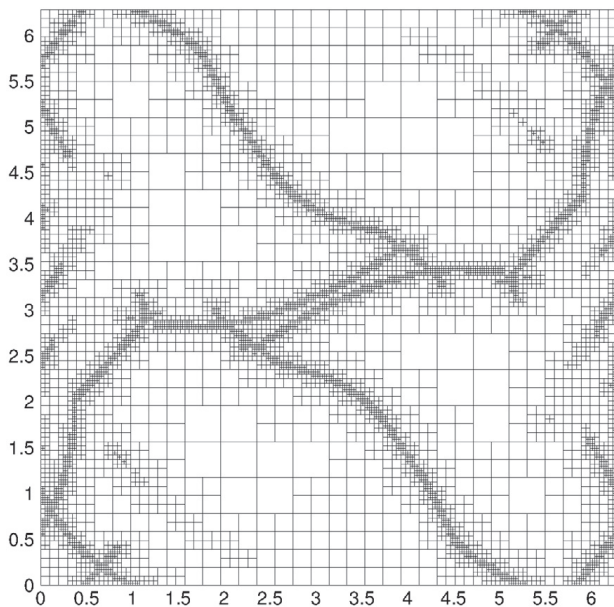
(c) AMR as applied to the Orszag-Tang vortex problem at t = 2.0. At this point, the mesh consists of 8,428 8-by-8 blocks, or 539,136 cells in total.

(d) AMR as applied to the Orszag-Tang vortex problem at t = 3.0. At this point, the mesh consists of 13,522 8-by-8 blocks, or 865,408 cells in total.

Fig. 11. The evolution of the mesh for the simulation of Fig. 10 with adaptive refinement. Up to t = 1.0, the mesh is refined every 0.025 s, after which it is refined every 50 time steps. The lines in the figure represent the boundaries of the 8-by-8 Cartesian blocks.

later times, AMR is performed every 50 time steps because Δt decreases rapidly. The contour lines of the density for the Ors-zag–Tang vortex problem are shown at t = 0.5, t = 1.0, t = 2.0, and t = 3.0 in Fig. 10. The results show agreement with results shown in other papers [10,27,30,55]. Figure 11 shows the sequence of adaptive meshes. Comparing the density contour lines shown in Fig. 10 with the way the grid is refined as shown in Fig. 11, it can be seen that the refinement closely follows the parts of the solution where interesting flow features and discontinuities occur, illustrating the effectiveness of the smooth-ness indicator-based refinement criterion.

Following [27,30], pressure distribution cuts at t = 2.0 and t = 3.0 along the line y = 1.9635 are shown in Fig. 12. The AMR results are compared to results on a uniform 1024-by-1024 mesh. The uniform mesh corresponds to the smallest cell reso-lution at 7 levels of refinement, while 8 levels of refinement are used in the AMR results, so that the smallest cell in the
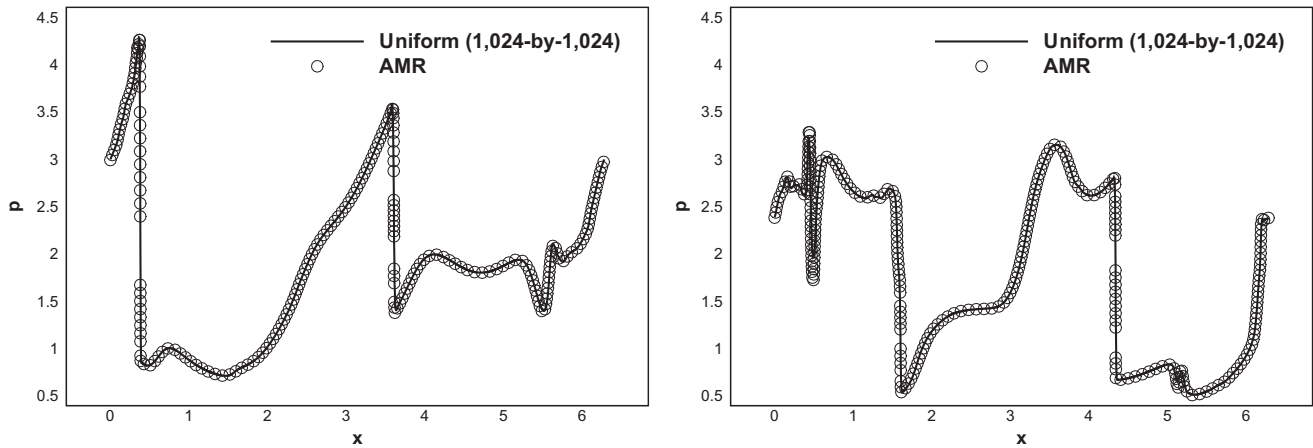
Fig. 12. Pressure cuts at $y = 1.9635$ at two different times ($t = 2.0$ [left], and $t = 3.0$ [right]). High-order results obtained in combination with adaptive mesh refinement are compared with uniform high-order high-resolution results on a 1024-by-1024 mesh, and found to be similar. Our results agree with the results from [27,30].

adaptive mesh (Fig. 11) corresponds to a resolution of 2048-by-2048 if done uniformly. From Fig. 12, it can be seen that the AMR results in general agree well with the uniform reference results. Note that our results have a much higher effective resolution than previously shown results (and they are fourth-order accurate). It is also interesting to note that, while the uniform mesh has 1,048,576 computational cells, the AMR mesh has 865,408 cells at $t = 3.0$, which is smaller than the uniform mesh, despite having twice the effective resolution at the highest level of refinement. Note also that, before $t = 3.0$, much fewer cells are used by the adaptive simulation (see Fig. 11). This illustrates the effectiveness of the CENO scheme in combination with the block-based AMR algorithm to reduce the number of required computational cells.

We conclude this section with some comments on AMR performance for the Orszag–Tang problem. Numerical experiments show that, for the adaptive simulation of Fig. 11, the speedup compared to a uniform simulation with the same maximal resolution (on a 2048-by-2048 mesh) is 2.066. In spite of refinement/coarsening overhead and some increase in load imbalance, the total CPU time is improved by a factor of more than two for AMR, and the number of cells at $t = 3$ is reduced by a factor of about four. In fact, AMR can be expected to be more advantageous for problems with greater disparity in spatial scales where a smaller portion of the domain is at the highest level of refinement, but it is already worthwhile for this simulation in which small-scale structure is spread through much of the domain and a relatively large fraction of the domain is at the finest mesh level. For the simulation of Fig. 11 in the developed flow after $t = 2$, one refinement/coarsening step takes, on average, approximately as much time as four to five simulation time steps. In each of these refinement/coarsening steps approximately 20% of the blocks were refined or coarsened. In terms of relative costs of the components of the AMR process, computing the refinement/coarsening indicators takes about 10% of the time of a refinement step, restriction and prolongation take about 60%, and moving blocks between nodes for load balancing takes about 30%. In terms of parallel scaling of the AMR process, repeating the simulation while doubling the number of cores from 128 to 256 cores and keeping the problem size constant leads to an efficiency loss of about 10% to 15%, which is expected due to the increase in communication and the reduction of the number of blocks per core, which exacerbates the load imbalances that may occur due to AMR. Note that this loss of efficiency depends on the block size and the specifications of the communication network. Note also that AMR steps do not need to be performed after every time step because of the block-based nature of our approach, and because bulk solution features do not necessarily move with maximum wave speed. Therefore we follow the strategy that is common in dynamic block-based AMR to perform multiple time steps between AMR steps (see [17–21] and references therein). Numerical experimentation has shown that an offset of 50 time steps between AMR steps is a suitable choice for the AMR simulation of Fig. 11. Note that this choice is not very sensitive; we found mostly similar results with offsets up to 200.

4.4. Discussion on CENO performance and divergence errors

We conclude this section with some discussion on performance and on the behaviour of $\nabla \cdot \vec{B}$ in the numerical results obtained with the fourth-order CENO-GLM method.

A range of numerical experiments carried out on several computing platforms indicate that the fourth-order CENO spatial discretization optimized for computational efficiency (i.e., storing the pseudo-inverse matrix) is on average about three times more expensive than a second-order spatial discretization. This ratio is independent of the grid size, and since the error drops much faster for high-order schemes as grids are refined, there is significant potential for computational savings in regions of smooth flow. This was, for example, illustrated for the fourth-order CENO method in Section 5.10 of [2], where it was found that for a particular smooth 2D Euler flow the fourth-order CENO method reached the same solution accuracy as a second-order method on a grid with 200 times fewer cells, in 1/40th of the time. It has to be noted, however, that reaching

these kinds of savings for realistic flow problems with varying spatial and temporal scales and non-smooth features remains a significant research challenge.

The high-order explicit CENO scheme spends up to 90% of the total CPU time in computing the high-order spatial residual, out of which 75% is dedicated to performing the high-order CENO solution reconstruction. The remaining 15% consists of polynomial evaluations at the interfaces, evaluation of the Riemann fluxes there, and geometry evaluation. The high-order CENO solution reconstruction procedure consists in performing the least-squares reconstruction, the computation and analysis of smoothness indicators and the enforcement of solution monotonicity in cells deemed as non-smooth. Among these, the least-squares reconstruction represents 55% of the total simulation cost, consisting mostly of matrix–vector multiplications, which represent about 37% of the total simulation cost. Computation of smoothness indicators and monotonicity enforcement account for the other 20% of the total simulation cost. Note however that these values can vary depending on the particular flow problem under study.

As in other finite-volume or finite-difference schemes that employ GLM divergence cleaning, the divergence of the magnetic field is expected to be of the magnitude of the truncation error in the magnetic field components. For our smooth test problems, for example, the rotating outflow problem of Section 4.1.3, we have verified that this is indeed the case, and that ψ and $\nabla \cdot \vec{B}$ converge to zero with the same order as the other solution variables (plots not shown due to space constraints). For problems with shocks, the maximum values of ψ and $\nabla \cdot \vec{B}$ do generally not decrease with increasing grid resolution, consistent with the $O(1)$ jumps in the magnetic field components at the discontinuities, but the GLM mechanism effectively dampens the divergence errors and maintains the stability of the scheme.

5. Concluding remarks

We have proposed a high-order CENO FV scheme for ideal MHD. The scheme is based on the CENO approach that was proposed by Ivan and Groth for compressible Euler flows in [1] and uses the GLM divergence cleaning method for MHD of Dedner et al. [6]. The resulting FV MHD scheme is high-order accurate in smooth flow regions and robust against spurious oscillations at discontinuities. The proposed high-order accurate MHD scheme can be used on general polygonal grids and can deal naturally with resolution changes on hierarchical quadtree block-adaptive body-fitted grids. The proposed scheme was implemented in a highly sophisticated fourth-order accurate parallel MHD code on 2D dynamically-adaptive multi-block body-fitted structured grids, and curved boundaries are handled with high-order accuracy using high-order spline representations and constraints at the Gauss points.

Detailed numerical results were given that demonstrate high-order convergence for smooth flows, and robustness against oscillations for Riemann problems and other flows with shocks. A new MHD extension of the well-known Shu–Osher test problem [38] was proposed to test the ability of the high-order MHD scheme to resolve small-scale flow features in the presence of shocks. The dynamic AMR capabilities of our approach were demonstrated using adaptive time-dependent simulations of the Orszag–Tang vortex problem with high-order accuracy and unprecedented effective resolution.

The scheme we described can in principle be implemented with arbitrary order. It can also be extended naturally to three spatial dimensions and to unstructured grids, which are topics of future work.

Acknowledgements

Appendix A. Analysis of operator splitting error

Consider the conservation-form equations of ideal MHD with mixed hyperbolic-parabolic GLM correction, given by Eqs. (2), (3), (5), (8) and (9). For simplicity, we consider planar 2D MHD.

Let $\mathbf{U} = [\rho, \rho v_x, \rho v_y, B_x, B_y, e, \psi]^T$ be the vector of conserved variables, and let $\mathbf{V} = [\rho, v_x, v_y, B_x, B_y, p, \psi]^T$ be the vector of primitive variables. Consider smooth solutions of the conservation law. The quasi-linear form of the equations in conservative variables is given by

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbb{A}_c \frac{\partial \mathbf{U}}{\partial x} + \mathbb{B}_c \frac{\partial \mathbf{U}}{\partial y} + \mathbb{C}_c \mathbf{U} = 0, \tag{A.1}$$

and in primitive variables by

$$\frac{\partial \mathbf{V}}{\partial t} + \mathbb{A}_p \frac{\partial \mathbf{V}}{\partial x} + \mathbb{B}_p \frac{\partial \mathbf{V}}{\partial y} + \mathbb{C}_p \mathbf{V} = 0. \tag{A.2}$$

Here, the flux Jacobian matrices $\mathbb{A}_c$ and $\mathbb{B}_c$ are related to the coefficient matrices $\mathbb{A}_p$ and $\mathbb{B}_p$ as follows:

$$\mathbb{A}_c = \frac{\partial \mathbf{U}}{\partial \mathbf{V}} \, \mathbb{A}_p \, \frac{\partial \mathbf{V}}{\partial \mathbf{U}}, \qquad \mathbb{B}_c = \frac{\partial \mathbf{U}}{\partial \mathbf{V}} \, \mathbb{B}_p \, \frac{\partial \mathbf{V}}{\partial \mathbf{U}}. \tag{A.3}$$

These expressions can be used to compute $\mathbb{A}_c$ and $\mathbb{B}_c$ from

$$\mathbb{A}_p = \begin{bmatrix} v_x & \rho & 0 & 0 & 0 & 0 & 0 \\ 0 & v_x & 0 & -B_x/\rho & B_y/\rho & 1/\rho & 0 \\ 0 & 0 & v_x & B_y/\rho & -B_x/\rho & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & B_y & -B_x & -v_y & v_x & 0 & 0 \\ 0 & \gamma p & 0 & (\gamma-1)\vec{v}\cdot\vec{B} & 0 & v_x & -(\gamma-1)B_x \\ 0 & 0 & 0 & c_h^2 & 0 & 0 & 0 \end{bmatrix}, \tag{A.4}$$

$$\mathbb{B}_p = \begin{bmatrix} v_y & 0 & \rho & 0 & 0 & 0 & 0 \\ 0 & v_y & 0 & -B_y/\rho & -B_x/\rho & 0 & 0 \\ 0 & 0 & v_y & B_x/\rho & -B_y/\rho & 1/\rho & 0 \\ 0 & -B_y & B_x & v_y & -v_x & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & \gamma p & 0 & (\gamma-1)\vec{v}\cdot\vec{B} & v_y & -(\gamma-1)B_y \\ 0 & 0 & 0 & 0 & c_h^2 & 0 & 0 \end{bmatrix}, \tag{A.5}$$

(see [6]), and from

$$\frac{\partial \mathbf{U}}{\partial \mathbf{V}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_x & \rho & 0 & 0 & 0 & 0 & 0 \\ v_y & 0 & \rho & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ \vec{v}\cdot\vec{v}/2 & \rho v_x & \rho v_y & B_x & B_y & 1/(\gamma-1) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{A.6}$$

$$\frac{\partial \mathbf{V}}{\partial \mathbf{U}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -v_x/\rho & 1/\rho & 0 & 0 & 0 & 0 & 0 \\ -v_y/\rho & 0 & 1/\rho & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ \gamma\vec{v}\cdot\vec{v}/2 & -\gamma v_x & -\gamma v_y & -\gamma B_x & -\gamma B_y & \gamma & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{A.7}$$

It is easily seen from Eq. (9) that

$$\mathbb{C}_c = \mathbb{C}_p = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & c_h^2/c_p^2 \end{bmatrix}. \tag{A.8}$$

Further, for Eq. (A.1), define the differential operator

$$\mathbb{D}_c = \mathbb{A}_c \frac{\partial}{\partial x} + \mathbb{B}_c \frac{\partial}{\partial y}. \tag{A.9}$$

Following Section 17.3 of [48], the splitting error $\mathbf{E}$ arising from operator splitting on the time integration of Eq. (A.1) at some time t is given by

$$\mathbf{E} = \frac{1}{2}\Delta t^2 (\mathbb{D}_c \mathbb{C}_c - \mathbb{C}_c \mathbb{D}_c)\mathbf{U} + O(\Delta t^3), \tag{A.10}$$

where **U** is the exact solution of Eq. (A.1) at that time t. Simple multiplication of the operators $\mathbb{C}_c$ and $\mathbb{D}_c$ shows that

$$(\mathbb{D}_c\mathbb{C}_c - \mathbb{C}_c\mathbb{D}_c)\mathbf{U} = \frac{c_h^2}{c_p^2}\begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{\partial \psi}{\partial x} \\ \frac{\partial \psi}{\partial y} \\ 0 \\ 0 \end{bmatrix} - \frac{c_h^4}{c_p^2}\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \frac{\partial B_x}{\partial x} + \frac{\partial B_y}{\partial y} \end{bmatrix}. \tag{A.11}$$

This vanishes because the exact solution of Eq. (A.1) satisfies $\psi(x,y,t) = 0$ and $\nabla \cdot \vec{B}(x,y,t) = 0$ for all x, y and t, showing that operator splitting is at least third-order accurate locally in time (and second-order accurate globally). Again following [48], all higher-order error terms also vanish when $(\mathbb{D}_c\mathbb{C}_c - \mathbb{C}_c\mathbb{D}_c)\mathbf{U} = 0$. This shows that no extra error arises from performing the time integration using operator splitting: the result is accurate up to the order of accuracy of the methods used to compute the solution in the separate steps of the operator splitting. Note that this result is obtained essentially because all quantities in the equation for ψ converge to zero for smooth flows, and because there is a source term in the equation for ψ only. The same result can be obtained for the case of 3D MHD in an analogous fashion.

References

[1] L. Ivan, C.P.T. Groth, High-order central ENO finite-volume scheme with adaptive mesh refinement, AIAA Paper (2007) 2007–4323.
[2] L. Ivan, Development of high-order CENO finite-volume schemes with block-based adaptive mesh refinement, Ph.D. thesis, University of Toronto, 2011.
[3] L. Ivan, C.P.T. Groth, High-order solution adaptive central essentially non-oscillatory (CENO) method for viscous flows, AIAA Paper (2011) 2011. 367.
[4] L. Ivan, C.P.T. Groth, High-order solution-adaptive central essentially non-oscillatory (CENO) method for viscous flows, Journal of Computational Physics, submitted for publication.
[5] T.J. Barth, Recent developments in high order k-exact reconstruction on unstructured mesh, AIAA Paper (1993) 93–0668.
[6] A. Dedner, F. Kemm, D. Kroner, C-D. Munz, T. Schitzer, M. Wesenberg, Hyberbolic divergence cleaning for the MHD equations, Journal of Computational Physics 175 (2002) 645–673.
[7] J.U. Brackbill, D.C. Barnes, Note: The effect of nonzero $\nabla \cdot \mathbf{B}$ on the numerical solution of the magnetohydrodynamic equations, Journal of Computational Physics 35 (1980) 426.
[8] K.G. Powell, An approximate Riemann solver for magnetohydrodynamics, The University of Michigan, Technical report, 1994.
[9] A.J. Chorin, A numerical method for solving incompressible viscous flow problems, Journal of Computational Physics 2 (1967) 12–26.
[10] G. Toth, The $\nabla \cdot \mathbf{B} = 0$ constraint in shock-capturing magnetohydrodynamics codes, Journal of Computational Physics 161 (2000) 605–652.
[11] C.R. Evans, J.F. Hawley, Simulation of magnetohydrodynamic flows: A constrained transport method, Astrophysical Journal 332 (1988) 659.
[12] H. De Sterck, Multi-dimensional upwind constrained transport on unstructured grids for 'shallow water' magnetohydrodynamics, AIAA Paper (2001) 2001–2623.
[13] D.S. Balsara, Second-order accurate schemes for magnetohydrodynamics with divergence-free reconstruction, The Astrophysical Journal Supplement Series 151 (2004) 149–184.
[14] D.S. Balsara, Divergence-free reconstruction of magnetic fields and WENO schemes for magnetohydrodynamics, Journal of Computational Physics 228 (2009) 5040–5056.
[15] Dinshaw S. Balsara, Divergence-free adaptive mesh refinement for magnetohydrodynamics, Journal of Computational Physics 174 (2) (2001) 614–648.
[16] G. Toth, P.L. Roe, Divergence- and curl-preserving prolongation and restriction formulas, Journal of Computational Physics 180 (2) (2002) 736–750.
[17] C.P.T. Groth, S.A. Northrup, Parallel implicit adaptive mesh refinement scheme for body-fitted multi-block mesh, Paper 2005-5333, AIAA, June 2005.
[18] J.S. Sachdev, C.P.T. Groth, J.J. Gottlieb, A parallel solution-adaptive scheme for predicting multi-phase core flows in solid propellant rocket motors, International Journal of Computational Fluid Dynamics 19 (2) (2005) 157–175.
[19] X. Gao, C.P.T. Groth, A parallel adaptive mesh refinement algorithm for predicting turbulent non-premixed combusting flows, International Journal of Computational Fluid Dynamics 20 (5) (2006) 349–357.
[20] X. Gao, C.P.T. Groth, A parallel solution-adaptive method for three-dimensional turbulent non-premixed combusting flows, Journal of Computational Physics 229 (5) (2010) 3250–3275.
[21] X. Gao, S.A. Northrup, C.P.T. Groth, Parallel solution-adaptive method for two-dimensional non-premixed combusting flows, Progress in Computational Fluid Dynamics 11 (2) (2011) 76–95.
[22] J.S. Sachdev, C.P.T. Groth, A mesh adjustment scheme for embedded boundaries, Communications in Computational Physics 2 (6) (2007) 1095–1124.
[23] Z.J. Zhang, C.P.T. Groth, Parallel high-order anisotropic block-based adaptive mesh refinement finite-volume scheme, Paper 2011-3695, AIAA, June 2011.
[24] T.C. Warburton, G.E. Karniadakis, A discontinuous Galerkin method for the viscous MHD equations, Journal of Computational Physics 152 (1999) 608–641.
[25] F. Li, C.-W. Shu, Locally divergence-free discontinuous Galerkin methods for MHD equations, Journal of Scientific Computing 22–23 (2005) 413–442.
[26] C. Altmann, Thomas Belat, Michael Gutnic, Philippe Helluy, Helene Mathis, Eric Sonnendruecker, Wilfredo Angulo, and Jean-Marc Herard. A local time-stepping discontinuous Galerkin algorithm for the MHD system, in: ESAIM: Proceedings, vol. 28, EDP Sciences, August 2009, pp. 33–54.
[27] F. Li, L. Xu, S. Yakovlev, Central discontinuous Galerkin methods for ideal MHD equations with the exactly divergence-free magnetic field, Journal of Computational Physics 230 (2011) 4828–4847.
[28] G-S. Jiang, C. Wu, A high-order WENO finite difference scheme for the equations of ideal magnetohydrodynamics, Journal of Computational Physics 150 (1999) 561–594.
[29] J. Kleimann, A. Kopp, H. Fichtner, R. Grauer, K. Germaschewski, Three-dimensional MHD high-resolution computations with CWENO employing adaptive mesh refinement, Computer Physics Communications 158 (2004) 47–56.
[30] J. Balbás, E. Tadmor, Non-oscillatory central schemes for one- and two-dimensional MHD equations. II: High-order semi-discrete schemes, SIAM Journal of Scientific Computing 28 (2006) 533–560.
[31] D.S. Balsara, T. Rumpf, M. Dumbser, C-D. Munz, Efficient, high accuracy ADER-WENO schemes for hydrodynamics and divergence-free magnetohydrodynamics, Journal of Computational Physics 228 (2009) 2480–2516.
[32] A. Mignone, P. Tzeferacos, A second-order unsplit Godunov scheme for cell-centered MHD: The CTU-GLM scheme, Journal of Computational Physics 229 (2010) 2117–2138.
[33] A. Mignone, P. Tzeferacos, G. Bodo, High-order conservative finite difference GLM-MHD schemes for cell-centered MHD, Journal of Computational Physics 229 (2010) 5896–5920.

[34] A. Harten, High resolution schemes for hyperbolic conservation laws, Journal of Computational Physics 49 (1983) 357–393.
[35] A. Harten, S.R. Chakravarthy. Multi-dimensional ENO schemes for general geometries, ICASE Report 91-76, 1991.
[36] H. Nessyahu, E. Tadmor, Non-oscillatory central differencing for hyperbolic conservation laws, Journal of Computational Physics 87 (1990) 408.
[37] M.S. Yalim, D. Vanden Abeele, A. Lani, T. Quintino, H. Deconinck, A finite volume implicit time integration method for solving the equations of ideal magnetohydrodynamics for the hyperbolic divergence cleaning method, Journal of Computational Physics 230 (2011) 6136–6154.
[38] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, II, Journal of Computational Physics 83 (1989) 32–78.
[39] S. Orszag, C.M. Tang, Small-scale structure of two-dimensional magnetohydrodynamic turbulence, Journal of Fluid Mechanics 90 (1979) 129–143.
[40] C.D. Munz, P. Omnes, R. Schneider, E. Sonnendrücker, U. Voss, Divergence correction techniques for Maxwell solvers based on a hyperbolic model, Journal of Computational Physics 161 (2000) 484.
[41] K.G. Powell, P.L. Roe, T.J. Linde, D.L. De Zeeuw, A solution-adaptive upwind scheme for ideal magnetohydrodynamics, Journal of Computational Physics 154 (1999) 284–309.
[42] C. Hirsch, Numerical Computation of Internal and External Flows, John Wiley and Sons, 1990.
[43] B. van Leer, W.-T. Lee, P.L. Roe, K.G. Powell, C.-H. Tai, Design of optimally smoothing multistage schemes for the Euler equations, Communications in Applied Numerical Methods 8 (10) (1992) 761–769.
[44] D.J. Mavriplis, Revisiting the least-squares procedure for gradient reconstruction on unstructured meshes, Paper 2003-3986, AIAA, June 2003.
[45] V. Venkatakrishnan, On the accuracy of limiters and convergence to steady state solutions, Paper 93-0880, AIAA, January 1993.
[46] C. Michalak, C. Ollivier-Gooch, Accuracy preserving limiter for the high-order accurate solution of the Euler equations, Journal of Computational Physics 228 (2009) 8693–8711.
[47] V. Wheatley, H. Kumar, P. Huguenot, On the role of Riemann solvers in discontinuous Galerkin methods for magnetohydrodynamics, Journal of Computational Physics 229 (2010) 660–680.
[48] R.J. LeVeque, Finite-Volume Methods for Hyperbolic Problems, Cambridge University Press, 2002.
[49] W. Gropp, E. Lusk, A. Skjellum, Using MPI, MIT Press, Cambridge, Massachussets, 1999.
[50] W. Gropp, E. Lusk, R. Thakur, Using MPI-2, MIT Press, Cambridge, Massachussets, 1999.
[51] C.F. Ollivier-Gooch, M. Van Altena, A high-order accurate unstructured mesh finite-volume scheme for the advection-diffusion equation, Journal of Computational Physics 181 (2002) 729–752.
[52] H. De Sterck, A. Csik, D. Vanden Abeele, S. Poedts, H. Deconinck, Stationary two-dimensional magnetohydrodynamics flows with shocks: Characteristic analysis and grid convergence study, Journal of Computational Physics 166 (2001) 28–62.
[53] M. Brio, C.C. Wu, An upwind differencing scheme for the equations of ideal magnetohydrodynamics, Journal of Computational Physics 75 (1988) 400.
[54] J.A. Rossmanith, An unstaggered, high-resolution constained transport method for magnetohydrodynamic flows, SIAM Journal on Scientific Computing 28 (2006) 1766–1797.
[55] J. Balbás, E. Tadmor, Cheng-Chin Wu, Non-oscillatory central schemes for one- and two-dimensional MHD equations: I, Journal of Computational Physics, 201 (2004) 261–285.