

High-order Magnetohydrodynamics for Astrophysics with an Adaptive Mesh Refinement Discontinuous Galerkin Scheme

Thomas Guillet^{1,2*}, Rüdiger Pakmor¹, Volker Springel^{1,2,3},
Praveen Chandrashekar⁴, Christian Klingenberg⁵

¹Heidelberg Institute for Theoretical Studies, Schloss-Wolfsbrunnengasse 35, 69118 Heidelberg, Germany

²Zentrum für Astronomie der Universität Heidelberg, Astronomisches Rechen-Institut, Mönchhofstr. 12–14, 69120 Heidelberg, Germany

³Max-Planck-Institut für Astrophysik, Karl-Schwarzschild-Str. 1, 85741 Garching, Germany

⁴TIFR Center for Applicable Mathematics, Bangalore, India

⁵Department of Mathematics, Universität Würzburg, Emil Fischer Str. 40, 97074 Würzburg, Germany

Accepted XXX. Received YYY; in original form ZZZ

ABSTRACT

Modern astrophysical simulations aim to accurately model an ever-growing array of physical processes, including the interaction of fluids with magnetic fields, under increasingly stringent performance and scalability requirements driven by present-day trends in computing architectures. Discontinuous Galerkin methods have recently gained some traction in astrophysics, because of their arbitrarily high order and controllable numerical diffusion, combined with attractive characteristics for high performance computing. In this paper, we describe and test our implementation of a discontinuous Galerkin (DG) scheme for ideal magnetohydrodynamics in the AREPO-DG code. Our DG-MHD scheme relies on a modal expansion of the solution on Legendre polynomials inside the cells of an Eulerian octree-based AMR grid. The divergence-free constraint of the magnetic field is enforced using one out of two distinct cell-centered schemes: either a Powell-type scheme based on nonconservative source terms, or a hyperbolic divergence cleaning method. The Powell scheme relies on a basis of divergence-free vector polynomials inside each cell to represent the magnetic field. Limiting prescriptions are implemented to ensure non-oscillatory and positive solutions. We show that the resulting scheme is accurate and robust: it can achieve high-order and low numerical diffusion, as well as accurately capture strong MHD shocks. In addition, we show that our scheme exhibits a number of attractive properties for astrophysical simulations, such as lower advection errors and better Galilean invariance at reduced resolution, as well as more accurate capturing of barely resolved flow features. We discuss the prospects of our implementation, and DG methods in general, for scalable astrophysical simulations.

Key words: methods: numerical – MHD – hydrodynamics – shock waves

1 INTRODUCTION

Numerical simulations in astrophysics have established themselves as a critical tool to study the complex interactions of physical processes in the Universe. Galaxy formation simulations, in particular, are faced with the challenge of accounting for ever expanding models of the physics, while requiring increasingly accurate numerical methods and scalable high-performance implementations.

Among these physical processes, cosmic magnetic fields are now recognized as playing a key role in star formation in the interstellar medium, and possibly impacting the dynamics of gas in galaxies at larger scales as they get amplified by multiple dynamo

processes which are still only partially understood. For this reason, many astrophysical codes have been implementing magnetohydrodynamics (MHD) solvers, to compute the joint evolution of gas and magnetic fields (e.g. Fryxell et al. 2000; Fromang et al. 2006; Mignone et al. 2007; Stone et al. 2008; Collins et al. 2010; Pakmor et al. 2011). These methods have been applied to setups of growing physical complexity, from the study of magnetized turbulence in the interstellar medium (Stone et al. 1998; Mac Low 1999; Schekochihin et al. 2004; Federrath et al. 2010), to magnetic field evolution in isolated galaxy setups (e.g. Wang & Abel 2009; Pakmor & Springel 2013; Rieder & Teyssier 2016), and more recently, the amplification of magnetic fields in galaxies in full cosmological context (Pakmor et al. 2014, 2017; Rieder & Teyssier 2017).

The increasing CPU and memory requirements of these simulations are confronted today by modern trends in computing archi-

* E-mail: thomas.guillet@h-its.org

tures. It is now well recognized that advances in computing power are more due to the power-efficient integration of multiple processor cores, rather than single-core performance growth. In addition, memory technology has not been keeping up with the steady gains in floating-point computing power, lagging behind both in terms of performance and capacity. As a result, codes have to turn towards increasingly parallel and compute-efficient numerical methods to face the expanding computing and memory requirements of the physical models they attempt to simulate.

In recent years, discontinuous Galerkin (DG) methods have gained traction in the computational fluid dynamics community in general, and in astrophysics in particular, as a promising framework for scalable and accurate high-order methods for hyperbolic problems. DG schemes lie at the crossroads of spectral element and finite volume methods: based on the expansion of the solution inside simulation volumes on a set of chosen basis functions (typically polynomials), DG schemes evolve the expansion coefficients (the so-called weights) in time based on a weak formulation of the governing equations. The popularity of DG stems from the fact that it provides a clean framework for discretizing hyperbolic problems at any order of spatial accuracy, with attractive data locality and reduced data exchanges between neighboring cells regardless of spatial order, especially when compared to reconstruction-based finite volume methods.

Because DG methods can easily be setup to operate at any spatial order, their numerical dissipation is controllable and can be reduced as required, which is of great interest to simulate the high Reynolds number flows of astrophysical plasmas. From a pure hydrodynamics point of view, [Bauer & Springel \(2012\)](#); [Nelson et al. \(2013\)](#) show that numerical dissipation can result in spurious heating of the intergalactic medium from turbulent motions, preventing cooling and accretion of the hot ambient gas. This result is corroborated by [Zhu et al. \(2013\)](#) in the context of higher-order reconstruction-based methods. In the context of decaying isothermal magnetized supersonic turbulence, the code comparison study of [Kritsuk et al. \(2011\)](#) has shown that higher-order codes provide a larger turbulence spectral bandwidth and increased effective Reynolds number compared to second-order schemes. Even at second order, recent studies by [Mocz et al. \(2014\)](#) and [Schaal et al. \(2015\)](#) show that DG methods demonstrate overall superior accuracy compared to finite volume methods, in particular due to lower advection errors and reduced angular momentum diffusion. Beyond second order, [Schaal et al. \(2015\)](#) find that increasing the DG order reduces advection errors, and decreases the total time to solution on smooth problems for a given target accuracy.

These properties make DG schemes attractive for fixed grid Eulerian astrophysical codes, where large structures such as galaxies may travel across the grid with large bulk advection velocities. As a result, there has been significant recent interest in DG methods in astrophysics, for hydrodynamics ([Schaal et al. 2015](#)), ideal magnetohydrodynamics ([Mocz et al. 2014](#); [Dumbser & Loubère 2016](#); [Karami Halashi & Luo 2016](#)), non-ideal MHD ([Boscheri & Dumbser 2017](#)), as well as special- and general-relativistic MHD ([Zanotti et al. 2015](#); [Kidder et al. 2017](#); [Anninos et al. 2017](#); [Zhao & Tang 2017](#); [Fambri et al. 2018](#)).

In this paper, we present AREPO-DG, our extension of the hydrodynamical DG code TENET of [Schaal et al. \(2015\)](#), to ideal MHD. Like its predecessor, this work leverages the infrastructure of the moving mesh code AREPO ([Springel 2010](#)), but instead of a moving Voronoi mesh, relies on AREPO's optional support for fixed Eulerian adaptive grids. The goal is to develop new parallel Eulerian solvers within this existing framework which explores dif-

ferent accuracy and scalability compromises than the moving mesh solution.

Our code adopts a modal Legendre polynomial basis for the expansion of hydrodynamical variables. Two different methods are implemented to impose the divergence-free condition of the magnetic field: in a first scheme, which we refer to as the Powell method, the magnetic field components are expanded onto a special basis of locally divergence-free (LDF) vector polynomials, and additional source terms are introduced following the formulation of [Powell et al. \(1999\)](#). The other implemented divergence control method relies on hyperbolic divergence cleaning as formulated by [Dedner et al. \(2002\)](#), for which we typically use the same Legendre basis both for hydrodynamical variables and for the magnetic field components. The DG weights are evolved in time through explicit time integration of the semi-discrete scheme, in the so-called Runge-Kutta DG (RKDG) framework introduced by [Cockburn et al. \(1989\)](#). In this paper, we discuss our implementation, detailing the numerical ingredients required for code stability and accuracy, with a special focus on the treatment of the divergence of the magnetic field. Our scheme introduces two new elements: different discretizations of the Powell term in the DG framework in [4.1.3](#), and a non-linear divergence-free slope limiting procedure for the magnetic field in [4.2.4](#). We also cover some important aspects of the method related to performance and computing efficiency.

The present paper is organised as follows. In [Section 2](#), we review the governing equations for ideal MHD in their conserved formulation. In [Section 3](#), we describe our DG discretization and time integration scheme, with a focus on aspects specific to MHD, in particular the divergence-free basis for the Powell scheme. [Section 4](#) details the numerical ingredients required to make the scheme accurate and stable with MHD; we discuss in particular the issues related to oscillation limiting, control of the divergence of the magnetic field, and enforcement of solution positivity. In [Section 5](#), we present results on a number of test problems, showing that the method achieves high order in smooth regions, while capturing shocks and discontinuities. We show that higher orders indeed reduce advection errors and dissipation even in case of strong MHD shocks, and help to efficiently capture barely resolved flow features. We also present test problems specifically aimed at testing the control of the magnetic field divergence. In [Section 6](#), we discuss prospects for this method and astrophysical simulations, also covering some aspects related to implementation efficiency and performance. We also consider some challenges and future prospects for our scheme—and DG methods in general—towards large-scale production astrophysical simulations.

2 GOVERNING EQUATIONS

The ideal MHD equations may be written as a system of conservation laws of the form:

$$\frac{\partial \mathbf{u}}{\partial t} + \sum_{\alpha=1}^3 \frac{\partial \mathbf{f}_{\alpha}(\mathbf{u})}{\partial x_{\alpha}} = 0, \quad (1)$$

where the index α runs over all space dimensions. The vector \mathbf{u} holds the conserved variables of the system:

$$\mathbf{u} = \begin{bmatrix} \rho \\ \rho \mathbf{v} \\ E \\ \mathbf{B} \end{bmatrix}, \quad E = \rho u + \frac{1}{2} \rho v^2 + \frac{1}{2} \mathbf{B}^2, \quad (2)$$

where ρ is the density, \mathbf{v} the fluid velocity, E the total energy density, \mathbf{B} the magnetic field vector, p the hydrodynamic (thermal) pressure, u is the specific internal energy density, which we assume to be related to the pressure p by the ideal gas equation of state with adiabatic index γ :

$$p = \rho u(\gamma - 1), \quad (3)$$

from which we will write the total energy as

$$E = \frac{p}{\gamma - 1} + \frac{1}{2}\rho v^2 + \frac{1}{2}\mathbf{B}^2. \quad (4)$$

The fluxes $\mathbf{f}_\alpha(\mathbf{u})$ for ideal MHD are given by

$$\mathbf{f} = \begin{bmatrix} \rho \mathbf{v} \\ \rho \mathbf{v} \otimes \mathbf{v} + p_T \mathcal{I}_3 - \mathbf{B} \otimes \mathbf{B} \\ (E + p_T)\mathbf{v} - (\mathbf{v} \cdot \mathbf{B})\mathbf{B} \\ \mathbf{B} \otimes \mathbf{v} - \mathbf{v} \otimes \mathbf{B} \end{bmatrix}, \quad (5)$$

where \mathcal{I}_3 is the 3×3 identity matrix, and p_T is the total pressure defined by

$$p_T = p + \frac{1}{2}\mathbf{B}^2. \quad (6)$$

For physical solutions, the Maxwell equations impose the additional constraint

$$\nabla \cdot \mathbf{B} = 0 \quad (7)$$

everywhere and at all times. The form of the flux (5) guarantees that initial divergence-free solutions will remain divergence-free under the dynamical evolution dictated by (1). However, this is not necessarily true for approximate numerical solutions, which require special divergence control measures.

3 DISCONTINUOUS GALERKIN SCHEME

We now describe our discontinuous Galerkin scheme, following [Schaal et al. \(2015\)](#) but adapted to MHD. In the following, we present the more general case of the Powell divergence control scheme, which relies on a basis of locally divergence-free polynomials for the magnetic field components. The other divergence control scheme, hyperbolic cleaning, requires only the use of the Legendre basis. We discuss both implementations of divergence control in Section 4.1.

3.1 Representation of the conserved state

The computational domain is partitioned into non-overlapping cubic cells, structured as an AMR octree. Within each cell K , the local conserved state vector $\mathbf{u}^K(\mathbf{x}, t)$ is expanded as a linear combination of N time-independent basis functions $\phi_l^K(\mathbf{x})$, defining time-dependent weights $w_l^K(t)$:

$$\mathbf{u}^K(\mathbf{x}, t) = \sum_{l=1}^N w_l^K(t) \phi_l^K(\mathbf{x}). \quad (8)$$

For each basis index l , ϕ_l^K is a vector of 8 time-independent scalar fields over K , one for each conserved component, and $w_l^K(t)$ is a time-dependent scalar function.

The weights w_l^K are defined independently in each cell, resulting in fields which are discontinuous at interfaces. Since all cells are cubic, we can define reference basis functions ϕ_l independent

of cell K by simple translation and rescaling of world coordinates \mathbf{x} to reference cell coordinates $\xi \in [-1, 1]^3$:

$$\phi_l^K(\mathbf{x}) = \phi_l(\xi), \quad \xi = \frac{2}{\Delta x^K}(\mathbf{x} - \mathbf{x}_c^K) \quad (9)$$

where Δx^K , \mathbf{x}_c^K are the cell size and cell center in world coordinates.

We further define the L^2 inner product on conserved vector fields in the reference cell:

$$\langle \mathbf{u}_1, \mathbf{u}_2 \rangle := \int_{[-1,1]^3} \mathbf{u}_1(\xi) \cdot \mathbf{u}_2(\xi) \, d\xi, \quad (10)$$

where the dot product on the right hand side is the usual dot product of \mathbb{R}^8 , i.e. across conserved components. By choosing a set of basis functions orthogonal with respect to this inner product, we can readily invert (8) by taking the inner product against each basis function:

$$w_l^K(t) = M_l^{-1} \langle \mathbf{u}^K, \phi_l \rangle, \quad (11)$$

where

$$M_l := \langle \phi_l, \phi_l \rangle \quad (12)$$

is the diagonal of the mass matrix $\langle \phi_l, \phi_{l'} \rangle$, all non-diagonal entries being zero. We now discuss the two types of basis functions making up our global DG basis $\{\phi_l\}$, $l \in \{1, \dots, N\}$.

3.2 Legendre basis for hydrodynamics

Following [Schaal et al. \(2015\)](#), we expand the hydrodynamical variables ρ , $\rho \mathbf{v}$ and E independently onto 5 scalar Legendre polynomials of degree $\leq d$. In 3 dimensions, scalar trivariate Legendre polynomials are indexed by 3 integers k_1, k_2, k_3 with $k_1 + k_2 + k_3 \leq d$, and take the form:

$$P_{k_1, k_2, k_3}(\xi) = \ell_{k_1}(\xi_1) \ell_{k_2}(\xi_2) \ell_{k_3}(\xi_3), \quad k_1 + k_2 + k_3 \leq d, \quad (13)$$

where ℓ_k is the univariate Legendre polynomial of degree k . The P_{k_1, k_2, k_3} form a basis of the space of scalar trivariate polynomials of degree $\leq d$, which has dimension

$$n(d) = \frac{1}{6}(d+1)(d+2)(d+3). \quad (14)$$

For brevity, we collectively label (k_1, k_2, k_3) with a single integer in $\{1, \dots, n(d)\}$. Note that the univariate Legendre polynomials ℓ_k are orthogonal for the L^2 inner product on $[-1, 1]$:

$$\int_{[-1,1]} \ell_k(x) \ell_{k'}(x) \, dx = 0 \Leftrightarrow k \neq k', \quad (15)$$

and as a result, the polynomials of (13) are orthogonal for the same inner product on the reference cell $[-1, 1]^3$.

Since we are expanding 5 conserved components independently, the Legendre subset of our basis functions will have dimension $N_L = 5n(d)$. We can choose the basis indexing l so that the first $n(d)$ components $l \in \{1, \dots, n(d)\}$ correspond to the Legendre expansion of ρ , then $l \in \{n(d)+1, \dots, 2n(d)\}$ correspond to ρv_1 , etc. To write the Legendre ϕ_l explicitly in terms of vectors of conserved components:

$$\phi_l(\xi) = \begin{bmatrix} P_l(\xi) \\ \mathbf{0} \\ 0 \\ \mathbf{0} \end{bmatrix}, \quad l \in \{1, \dots, n(d)\}, \quad \text{for } \rho \quad (16)$$

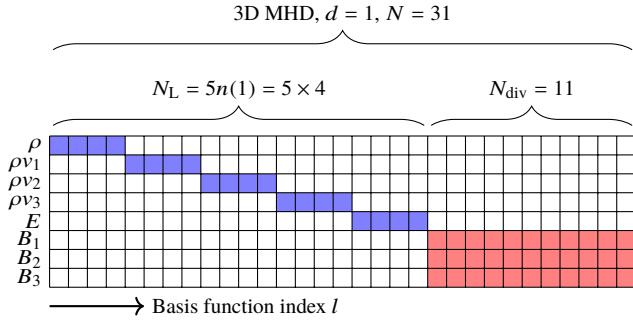


Figure 1. Structure of the DG basis functions for 3D MHD with degree $d = 1$ polynomials, showing nonzero contributions (colored cells) of each basis function ϕ_l (l -th column) to components of the conserved state vector \mathbf{u} (rows). The Legendre subspace for hydrodynamical variables has total dimension $N_L = 20$, whereas the divergence-free subspace for \mathbf{B} is of dimension $N_{\text{div}} = 11$. Cells in white correspond to zero. While the Legendre subspace has a tensor product structure, all 3 components of \mathbf{B} in the divergence-free basis are coupled.

up to

$$\phi_{4n(d)+l}(\boldsymbol{\xi}) = \begin{bmatrix} 0 \\ \mathbf{0} \\ P_l(\boldsymbol{\xi}) \\ \mathbf{0} \end{bmatrix}, \quad l \in \{1, \dots, n(d)\}, \quad \text{for } E. \quad (17)$$

Since the multivariate Legendre polynomials P are L^2 -orthogonal, the ϕ we just constructed are orthogonal for the inner product (10).

3.3 Divergence-free basis for the magnetic field

Locally divergence-free (LDF) basis functions for discontinuous Galerkin techniques were first introduced by Cockburn et al. (2004) for the linear Maxwell equations, and extended by Li & Shu (2005) to non-linear MHD in the context of RKDG methods. Divergence-free bases have also been used for DG in the context of relativistic MHD by Zhao & Tang (2017). The idea is to enforce the divergence-free condition (7) inside the cells by expanding the magnetic field \mathbf{B} in (8) onto a basis of divergence-free vector polynomials, which in 3D has dimension:

$$N_{\text{div}} = 3n(d) - n(d-1). \quad (18)$$

We will discuss the properties of this basis for divergence control in Section 4.1, and we refer the reader to Appendix A for more details related to properties and constructions of such bases in 2D and 3D. In two dimensions, suitable basis functions may be found for example in Cockburn et al. (2004); Li & Shu (2005) up to $d = 2$, or Zhao & Tang (2017) up to $d = 3$.

The resulting total dimension of the DG basis with its Legendre and divergence-free subspaces is

$$N = N_L + N_{\text{div}} = 8n(d) - n(d-1). \quad (19)$$

Its general structure is represented in Fig. 1. As detailed in Appendix A, the divergence-free basis is constructed so that its vectors are orthogonal with respect to the inner product (10).

The resulting combined Legendre and divergence-free basis is of dimension $N = N_L + N_{\text{div}}$, and is orthogonal:

$$\forall (l, l') \in \{1, \dots, N\}^2, \quad \langle \phi_l, \phi_{l'} \rangle = \delta_{ll'} M_l, \quad M_l > 0, \quad (20)$$

which results from the block structure shown in Fig. 1, together with

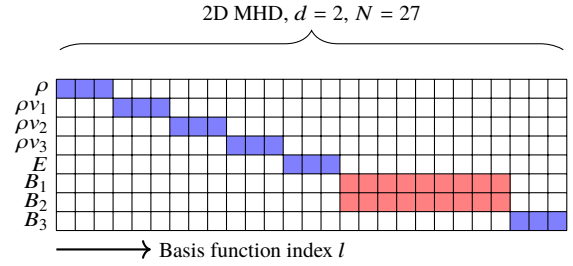


Figure 2. Structure of the DG basis functions for 2D MHD with degree $d = 2$ polynomials. See Fig. 1 for a detailed explanation of the 3D case. In the two-dimensional setup, $\partial/\partial x_3 \equiv 0$ and only B_1, B_2 are coupled by the divergence-free constraint. B_1, B_2 are therefore expanded on a 2D divergence-free basis, whereas B_3 is represented by an additional Legendre scalar component.

the orthogonality of the Legendre polynomials and divergence-free basis vectors. The constants M_l are fixed by the choice of basis polynomials.

LDF bases require slightly less storage per cell than expanding the components of \mathbf{B} as independent Legendre fields, due to the fewer degrees of freedom arising from the divergence constraint. However, we note that LDF bases prevent some optimizations that are possible with Legendre bases, thus requiring more floating point operations. They also require storing more precomputed data (such as the basis function values and gradients) compared to Legendre bases. We come back to these points in the discussion in Section 6.3.3.

3.4 The special case of 2D MHD

Two-dimensional MHD problems can be seen as being governed by the 3D equations, but with imposed translation invariance along the third dimension of space, i.e. $\partial/\partial x_3 = \partial/\partial \xi_3 \equiv 0$. In this case, the divergence-free condition on \mathbf{B} becomes $\nabla \cdot \mathbf{B} = \partial B_1/\partial x_1 + \partial B_2/\partial x_2 = 0$. The component B_3 is still present in the 2D equations, but may vary independently from B_1, B_2 .

In this case, we therefore follow Li & Shu (2005) and expand (B_1, B_2) on a 2D divergence-free basis, and treat B_3 as an independent Legendre scalar component. The resulting basis function structure is shown in Fig. 2.

3.5 Semi-discrete DG scheme

Under the decomposition (8), the time dependence of the solution is completely captured by the weights $w_l^K(t)$ for each cell K . To obtain the dynamics of the weights, we write the evolution equation in weak form, using our basis functions as test functions.

Within a given cell K , we can write the evolution equation (1) transformed to coordinates of the reference cell $[-1, 1]^3$:

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\Delta x^K}{2} \sum_{\alpha} \frac{\partial f_{\alpha}}{\partial \xi_{\alpha}} = s, \quad (21)$$

where we have added a generic source term s for generality. The $\Delta x^K/2$ factor arises from the mapping (9) from world coordinates \mathbf{x} to cell coordinates $\boldsymbol{\xi}$. For brevity, we will omit the superscripts indicating the local cell in the rest of the text, with the understanding that all quantities involved are always local to a given cell K unless specified otherwise. We then project this equation onto our space of DG functions by taking the inner product (10) with the basis

functions ϕ_l :

$$\frac{\partial}{\partial t} \langle \mathbf{u}, \phi_l \rangle + \frac{\Delta x}{2} \sum_{\alpha} \left\langle \frac{\partial f_{\alpha}}{\partial \xi_{\alpha}}, \phi_l \right\rangle = \langle s, \phi_l \rangle \quad (22)$$

which holds for all $l \in \{1, \dots, N\}$.

Using the expansion (8) of the solution and orthogonality (20) of the ϕ_l , the first term of the sum reduces to $M_l \dot{w}_l(t) = M_l dw_l/dt$. The second term involves a volume integral over the reference cell which can be integrated by parts, yielding

$$M_l \dot{w}_l(t) + \frac{\Delta x^K}{2} \left[\int_{\partial([-1,1]^3)} \sum_{\alpha=1}^3 (f_{\alpha} \cdot \phi_l) \mathbf{n}_{\alpha} dS - \int_{[-1,1]^3} \sum_{\alpha=1}^3 \left(f_{\alpha} \cdot \frac{\partial \phi_l}{\partial \xi_{\alpha}} \right) d\xi \right] = \int_{[-1,1]^3} (s \cdot \phi_l) d\xi, \quad (23)$$

where \mathbf{n}_{α} is the outwards pointing normal 3-vector to the face, and the dot product \cdot is the dot product of (10), i.e. the dot product of \mathbb{R}^8 across conserved components. Note that for the Legendre ϕ_l , this dot product effectively only selects one component at a time from f_{α} and s due to the structure of the ϕ_l represented in Fig. 1.

Note that in (23), the fluxes f_{α} appearing in the volume integral are evaluated using the analytical fluxes (5). The f_{α} at the faces, however, are computed using a numerical flux function which solves a 1D interface Riemann problem at the face, following the traditional finite volume technique. We discuss our choice of numerical flux in Section 4.1.2.

This equation directly yields the time evolution $w_l(t)$ of the weights for cell K , provided that we can compute the face and volume integrals. We use Gauss-Legendre quadrature with $d+1$ nodes per dimension to perform these integrals, as described in detail in Schaal et al. (2015). The resulting quadrature rule is exact for polynomials of degree $\leq 2d+1$, which allows to integrate products of two basis functions without any error, and would evaluate (22) exactly if the fluxes and source terms were linear in the conserved state.

3.6 Time integration

We integrate the weights in time using (23) with the traditional RKDG approach, using strong-stability preserving (SSP) explicit Runge-Kutta schemes of selectable order between 1 and 4 (see e.g. Gottlieb 2005). As described in Schaal et al. (2015), we use a *global* timestep Δt respecting the constraint of Cockburn & Shu (1989):

$$\Delta t = \min_K \frac{C}{2d+1} \left(\sum_{\alpha} \frac{|v_{\alpha}^K| + c^K}{\Delta x^K} \right)^{-1}, \quad (24)$$

where c^K is the maximum signal velocity in cell K (that is, $\sum_{\alpha} (|v_{\alpha}| + c)$ is an upper bound for the largest eigenvalue of the flux Jacobian). In absence of magnetic fields, c is the sound speed $c_S = \sqrt{\gamma p/\rho}$. In the MHD case, we take this upper bound to be $c = \sqrt{c_S^2 + c_A^2}$ where $c_A = |\mathbf{B}|/\sqrt{\rho}$ is the local Alfvén speed. With this choice, c is always greater or equal to the fast magnetosonic speed in the cell.

$C < 1$ is the chosen Courant number, which we typically set to $C = 0.8$. Note that the presence of source terms may require further reduction of the timestep. In particular we discuss the case of Powell source terms for divergence control in Section 4.1.4. Some RK schemes (typically at order 4 or more) may also require reducing the timestep (Gottlieb 2005).

For most of the test problems presented in Section 5 and in

particular the convergence tests presented in 5.2, we typically set the time integration order to match the spatial order of the scheme, to prevent time integration errors from dominating the total error of the solution. However, very high-order time integration schemes may not be required in practice for many science applications; we elaborate on this point in Section 6.1 in the discussion.

4 NUMERICAL INGREDIENTS

In this section, we detail the numerical components required to achieve stability and accuracy with the scheme.

4.1 Divergence control

As noted in Section 2, the Maxwell equations impose $\nabla \cdot \mathbf{B} = 0$ on all physical realizations of the magnetic field, everywhere and at all times. While the induction equation guarantees that an initially divergence-free magnetic field will remain so in time, truncation errors in discretized schemes can cause nonzero numerical divergence to appear. These errors can trigger a non-linear instability in the MHD equations and lead to blowup of the numerical solution (see Brackbill & Barnes 1980; Tóth 2000, and also Kemm 2013 for a detailed mathematical discussion of this instability). In addition, even if the numerical divergence stays bounded, divergence errors can still result in nonphysical perturbations to the flow, such as plasma acceleration along the magnetic field lines.

The issue of divergence control across cells is not specific to DG, and has received a lot of attention in the context of finite difference and finite volume MHD codes, resulting in the development of multiple techniques. Projection methods, introduced by Brackbill & Barnes (1980) and used in some recent schemes (e.g. Derigs et al. 2016) project the magnetic field onto a globally divergence-free representation at every time step. The main drawback of this technique is that it requires solving a global elliptic Poisson problem at each projection operation, which is expensive and less scalable than purely hyperbolic formulations as it requires global exchange of information. Another family of methods, constrained transport, keeps the magnetic field divergence-free to machine precision for some careful choice of discretization and update scheme for the induction equation (see e.g. Evans & Hawley 1988; Dai & Woodward 1998; Ryu et al. 1998; Balsara & Spicer 1999b; Gardiner & Stone 2005). Constrained transport methods have also been extended to non-staggered grids (Rossmannith 2006; Helzel et al. 2011). Constrained transport has been very popular with finite difference and finite volume grid codes in astrophysics (e.g. Stone & Norman 1992; Fromang et al. 2006; Stone et al. 2008; Collins et al. 2010; Mocz et al. 2016), because of its suitability for second-order mesh methods, exact divergence control, and lack of any tunable parameter in the scheme. Constrained transport schemes have also been extended to higher-order reconstruction methods (see e.g. Balsara 2009), and also to DG, involving either dual discretizations (Li et al. 2011; Li & Xu 2012; Xu & Liu 2016; Balsara & Käppeli 2017; Zhao & Tang 2017) or updating a vector potential with its own higher-order DG discretization (e.g. Rossmannith 2013). The main drawback of constrained transport in a DG setting is its implementation complexity and cost, requiring significantly more operations and storage to update the magnetic field in a divergence-free way.

For this work, we adopted two widespread divergence control techniques which allow working with cell-centered discretizations while preserving the hyperbolic character of the equations: the Powell scheme, based on the addition of a nonconservative source term

to the MHD equations, and hyperbolic divergence cleaning, which dynamically advects and dampens the numerical divergence using an additional scalar field. In the rest of this section, we describe both methods and detail our implementations.

4.1.1 Powell source terms

The so-called Powell scheme, after Powell et al. (1999), follows the insights of Godunov (1972), who pointed out that the conserved system (1) of ideal MHD does not formally conserve entropy and is also not Galilean invariant, unless a specific source term proportional to $\nabla \cdot \mathbf{B}$ is added. While identically zero on continuous physical solutions, this source term modifies the nature of the equations. The extra term may be obtained from deriving the local form of the conserved MHD equations based on integral conservation laws (Powell et al. 1999), or from requiring entropy stability (Godunov 1972; Chandrashekar & Klingenberg 2016; Winters & Gassner 2016; Liu et al. 2018). Defining the column vector

$$\mathbf{q}(\mathbf{u}) := (0, \mathbf{B}, \mathbf{v} \cdot \mathbf{B}, v)^T, \quad (25)$$

the method introduces an additional source term at the right hand side of the conserved equation (1):

$$s_{\text{Powell}} = -(\nabla \cdot \mathbf{B}) \mathbf{q}(\mathbf{u}). \quad (26)$$

Powell et al. (1999) derived the characteristics of the ideal MHD system with this source term, and showed that the addition of (26) results in an additional wave to the usual 7 waves, whose effect is to advect away $\nabla \cdot \mathbf{B} / \rho$ with the flow and restore Galilean invariance.

The main advantages of the Powell method are that it can be easily adapted to existing grid schemes, and does not require setting or tuning any free parameter. This scheme has been implemented in astrophysical MHD codes, both with adaptive mesh refinement (Mignone et al. 2012) or moving mesh (Pakmor & Springel 2013; Mocz et al. 2014) grids. In the DG context, it was also adopted by Warburton & Karniadakis (1999) for viscous MHD flows. Pakmor & Springel (2013) have found the Powell method to be more robust and stable than hyperbolic divergence cleaning when used with large dynamic ranges in time and space discretizations in the context of moving mesh simulations with local timestepping.

However, the Powell method comes with important limitations. Firstly, it does not completely eliminate the divergence, as it advects it away with the flow. It can therefore result in local accumulation of numerical divergence in the case of standing shocks, which are among the most challenging problems for static mesh MHD divergence control (Balsara & Spicer 1999b; Tóth 2000). Secondly, after adding the source term (26), the numerical scheme is not strictly conservative anymore: while the source term vanishes for exact physical solutions, it will locally inject conserved quantities whenever numerical divergence errors are present. As noted by Tóth (2000), this will result in wrong jump conditions across shock fronts. In this paper, we therefore will be carefully evaluating these effects in our implementation, including with a dedicated set of numerical tests in Section 5.4.

In the rest of this section, we describe the details of our Powell implementation, before covering the simpler hyperbolic cleaning scheme in Section 4.1.5.

4.1.2 Choice of numerical flux function

In this work, we consistently use the so-called HLLD fluxes of Miyoshi & Kusano (2005), which have gained widespread adoption

in astrophysics, largely because of their robustness, low numerical diffusion, and relative computational inexpensiveness. In particular, we use the same HLLD fluxes with both Powell and hyperbolic cleaning.

As noted by Powell et al. (1999), the addition of the source term (26) will modify the characteristics of the ideal MHD system by adding an 8-th so-called divergence wave. Formally, this would call for modifying the numerical flux function used at cell faces, since usual 1D Riemann solvers will not propagate jumps in the normal component of the magnetic field. Special Riemann solvers have been developed for 8-wave schemes (e.g. Powell 1994; Fuchs et al. 2011; Chandrashekar & Klingenberg 2016; Winters & Gassner 2016). We have experimented with a number of such numerical fluxes, including the 8-wave entropy-stable flux of Chandrashekar & Klingenberg (2016) and local Lax-Friedrichs 8-wave fluxes. Note that while our solver of choice, HLLD, is not an 8-wave Riemann solver, fluxes that do not incorporate the divergence wave have been used successfully with Powell schemes (Warburton & Karniadakis 1999; Mocz et al. 2014; Pakmor et al. 2017; Pakmor & Springel 2013), in which case the method reduces to adding a properly discretized source term.

Note however that, despite our HLLD flux not propagating the 8th divergence wave, we use the full 8-wave formalism when computing local characteristic eigensystems. We discuss characteristic decomposition in more detail in the description of slope limiting in Section 4.2.2.

4.1.3 LDF bases and discretization of the Powell term

Our Powell scheme requires expanding the magnetic field on locally divergence-free bases for stability. LDF basis functions have been found by several authors to significantly improve the stability of DG Maxwell and MHD schemes (Cockburn et al. 2004; Li & Shu 2005; Yakovlev et al. 2013; Zhao et al. 2014; Karami Halashi & Luo 2016), even though they are not sufficient by themselves for divergence control (Li & Shu 2005; Yakovlev et al. 2013). We now describe our choice of Powell term discretization with LDF basis functions, which our tests have found to be robust and accurate even with a non-diffusive solver like HLLD which does not propagate the divergence wave itself.

As noted by Waagan (2009); Waagan et al. (2011); Fuchs et al. (2011) in the context of finite volume methods, the exact discretization of the Powell term is critical to the stability of the scheme, and our experience with DG schemes can only support that statement. The problem is to find a consistent discretization of the contribution of the Powell source term (26) to the right hand side of the DG integral in (23):

$$\int_K s_{\text{Powell}} \cdot \phi_I d\xi = - \int_K (\nabla \cdot \mathbf{B}) (\mathbf{q} \cdot \phi_I) d\xi, \quad (27)$$

where \mathbf{q} is defined in (25). Although this term features a spatial derivative, it cannot be cast into conservative form as a flux, and integration by parts is not helpful. In addition, while $\nabla \cdot \mathbf{B}$ vanishes identically inside K because we use a LDF basis, the discontinuity of the normal components of \mathbf{B} will contribute surface terms to the integral, which mandates a careful discretization.

The face discontinuity suggests to split up the volume integral (27) into contributions in the interior (which will vanish by construction since $\nabla \cdot \mathbf{B} = 0$ for divergence-free bases), and contributions close to the faces. The overall situation is sketched in 1D in Fig. 3, from the point of view of the left cell L , close to a face F shared with its neighbor cell R . The gray area represents the volume

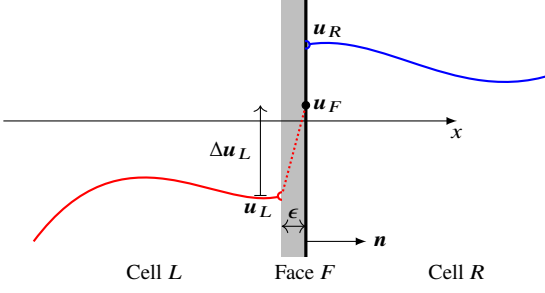


Figure 3. Discretization of the Powell term at a cell interface, from the point of view of the L cell (\mathbf{n} pointing outwards). \mathbf{u}_L and \mathbf{u}_R are the state vectors immediately left and right of the face F , obtained from the respective smooth DG state representations. \mathbf{u}_F is the state exactly at the face, which can be defined as the interface state returned by the Riemann solver. The total jump $\Delta \mathbf{u} = \mathbf{u}_R - \mathbf{u}_L$ at the face is split between the left and right cells into $\Delta \mathbf{u}_L$ and $\Delta \mathbf{u}_R$ using the face state.

of cell L “close” to the face, for some thin layer thickness $\epsilon \rightarrow 0$. The conserved state \mathbf{u}_L is obtained from the smooth DG state representation in cell L . Given some choice of state \mathbf{u}_F at the interface (which we will discuss below), the state in the left cell will jump by $\mathbf{u}_F - \mathbf{u}_L$ across the layer. This suggests that, in the notations of Fig. 3, the integral (27) may be computed as

$$\int_K s_{\text{Powell}} \cdot \phi_I = - \int_{\partial K} (\mathbf{q}^\dagger \cdot \phi_I) (\mathbf{B}_F - \mathbf{B}_L) \cdot \mathbf{n} dS \quad (28)$$

for some choice of \mathbf{q}^\dagger which determines how we average the non-linear product $(\nabla \cdot \mathbf{B}) \mathbf{q}$. If we choose some explicit form of conserved state \mathbf{u} which interpolates between \mathbf{u}_L and \mathbf{u}_F in the layer (represented for example by the dotted line in Fig. 3), we may in principle obtain (28) and \mathbf{q}^\dagger together directly by integrating (27) on the layer along the x direction, and taking the limit $\epsilon \rightarrow 0$. Note however that we may also choose \mathbf{q}^\dagger arbitrarily, independently from such considerations.

4.1.3.1 Interface state \mathbf{u}_F For HLL-type Riemann solvers, a natural choice for \mathbf{u}_F is to take the state from the Riemann fan which contains the interface. In particular, this choice guarantees that \mathbf{u}_F is properly upwinded, which Waagan (2009) argues to be critical to the stability of Powell schemes. In this work, we use the interface state from the HLLD solver.

4.1.3.2 Normal component B_F^n at the interface Let B^n denote the component of the magnetic field normal to the face. One-dimensional Riemann solvers such as HLLD usually assume that B^n is constant across the interface to satisfy the 1D divergence-free condition. These solvers therefore cannot prescribe any value B_F^n for B^n at the face whenever B^n is discontinuous. Consequently, before calling the Riemann solver to obtain the flux and face state, we decide on a face normal magnetic component B_F^n , and then assign it to the left and right states by setting $B_L^n = B_R^n = B_F^n$. A straightforward choice is to pick $B_F^n := \frac{1}{2}(B_L^n + B_R^n)$ (e.g. Dedner et al. 2002; Pakmor & Springel 2013). This choice results in a surface term proportional to the jump $\frac{1}{2}(B_R^n - B_L^n)$ which is also found in a number of published Powell term implementations (Waagan 2009; Chandrashekar & Klingenberg 2016; Liu et al. 2018). In this work, we found that we obtained slightly smaller divergence errors near shocks by instead averaging the Alfvén velocities of the left and

right states, and setting

$$B_F^n := \frac{B_L^n \sqrt{\rho_L}^{-1} + B_R^n \sqrt{\rho_R}^{-1}}{\sqrt{\rho_L}^{-1} + \sqrt{\rho_R}^{-1}}, \quad (29)$$

which we use in the rest of this work.

4.1.3.3 Choice of \mathbf{q}^\dagger Many prescriptions may be used for \mathbf{q}^\dagger , most of which we tested were found to be unstable in our DG scheme, including $\mathbf{q}^\dagger := \mathbf{q}(\mathbf{u}_L)$ or $\mathbf{q}^\dagger := \mathbf{q}(\mathbf{u}_F)$, despite the latter benefiting from upwinding from the Riemann solver. We describe two choices that we have found to be stable across all test problems.

First, as previously noted, we can construct a \mathbf{q}^\dagger by regularizing the integral (27) near the face and taking the limit $\epsilon \rightarrow 0$ for some choice of regularization (see Fig. 3). For a simple linear interpolation in primitive variables between the states \mathbf{u}_L and \mathbf{u}_F , this yields

$$\mathbf{q}^\dagger = \begin{bmatrix} 0 \\ \frac{1}{2}(\mathbf{B}_L + \mathbf{B}_F) \\ \frac{1}{6}(2\mathbf{B}_L \cdot \mathbf{v}_L + \mathbf{B}_L \cdot \mathbf{v}_F + \mathbf{B}_F \cdot \mathbf{v}_L + 2\mathbf{B}_F \cdot \mathbf{v}_F) \\ \frac{1}{2}(\mathbf{v}_L + \mathbf{v}_F) \end{bmatrix}. \quad (30)$$

This choice seems to be stable in our tests, and provides overall good results.

Second, we also found that we obtain a stable scheme by simply evaluating \mathbf{q}^\dagger at the average conserved state in the cell

$$\mathbf{q}^\dagger := \mathbf{q}(\langle \mathbf{u} \rangle_L), \quad \langle \mathbf{u} \rangle_L := \frac{1}{8} \int_{[-1,1]^3} \mathbf{u}_L(\xi) d\xi. \quad (31)$$

This prescription fully retains all the high order information about B^n at the face, which we find to be essential to preserve both the high-order property and stability of the scheme. However, it discards any local variation of the state within the cell to evaluate the \mathbf{q} part of the Powell term. For some test problems, the choice (31) yields slightly better results than (30). Because the cell average may be obtained from the lowest order DG weights without any additional computation, this choice is also computationally inexpensive. We therefore resort to using (31) by default.

For divergence control, the Orszag–Tang vortex problem described in Section 5.3.3 proved to be a particularly discriminatory test, as divergence issues will promptly cause distortions or ripples in smooth post-shock regions of the flow, or result in exponential divergence blowup. The problem is exacerbated by the use of non-diffusive Riemann solvers like HLLD.

For completeness, we remark that Janhunen (2000) proposed a combination of an HLL-inspired solver, together with a source term similar to (26) but with nonzero components in the induction equation only, which amounts to taking $\mathbf{B} = 0$ in (25). As a result, Janhunen’s source term preserves conservation of momentum and energy. In practice, while we find Janhunen’s term to work well with HLLD fluxes for some test problems, it seems to provide insufficient control of the magnetic field divergence in strong shock situations, which was also noted by other authors (e.g. Gaburov & Nitadori 2011).

4.1.4 Time step control for Powell term

In presence of the Powell source terms, we add an additional constraint to the timestep Δt to ensure that the time integration can stably resolve any local change in conserved quantities, in extreme cases where the Powell injection term would possibly become stiff.

Since the Powell term (26) contributes to the momentum, energy and magnetic components of the state vector, any of these may

in principle be used to limit the timestep. In practice, we found that using the total energy E provides a robust criterion to control relative changes, since the total energy is always a positive quantity, and the energy contribution is sensitive to all of \mathbf{v} , \mathbf{B} and $\nabla \cdot \mathbf{B}$. Therefore, we limit Δt so that the change ΔE in total energy E during a timestep Δt due to the Powell source term (26) respects:

$$\frac{\Delta E}{E} \approx \frac{1}{E} |\nabla \cdot \mathbf{B}| |\mathbf{v} \cdot \mathbf{B}| \Delta t < \eta \quad (32)$$

for some threshold η which we typically take to be $\eta = 0.2$.

4.1.5 Hyperbolic divergence cleaning

As a separate option for divergence control, we have also implemented the so-called hyperbolic divergence cleaning technique, proposed by Dedner et al. (2002), which introduces an additional scalar dynamical field which couples to $\nabla \cdot \mathbf{B}$ and results in hyperbolic advection and parabolic damping of the divergence. For some suitable choice of the advection velocity and damping time-scale parameters, divergence cleaning will efficiently attenuate and advect away divergence errors. Since it is straightforward to implement in existing schemes, it has been adopted by a number of MHD codes (e.g. Gaburov & Nitadori 2011; Mignone et al. 2012; Tricco & Price 2012; Hopkins & Raives 2016) including in a DG context (e.g. Boscheri et al. 2014; Zanotti et al. 2015; Dumbser & Loubère 2016; Kidder et al. 2017).

Our implementation closely follows the so-called generalized Lagrange multiplier (GLM) formulation of Dedner et al. (2002). An additional dynamical scalar field ψ is added to the 8 conserved variables of MHD. This field couples to the divergence of the magnetic field through a modified induction equation:

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla \cdot (\mathbf{B} \otimes \mathbf{v} - \mathbf{v} \otimes \mathbf{B}) + \nabla \psi = 0. \quad (33)$$

The field ψ is evolved according to an additional dynamical equation:

$$\frac{\partial \psi}{\partial t} + \frac{c_h^2}{c_p^2} \psi + c_h^2 \nabla \cdot \mathbf{B} = 0, \quad (34)$$

and the resulting coupled GLM system (33)–(34) makes the fluctuations of ψ propagate away from their sources at speed c_h while damping them with a time-scale $\tau = c_p^2/c_h^2$.

In practice, we introduce the extra variable ψ in the DG scheme as an additional scalar Legendre component. The analytical internal fluxes \mathbf{f} of Eq. (5) in the conserved formulation are modified: equation (33) amounts to an extra diagonal term $\psi \mathcal{I}_3$ in the fluxes of \mathbf{B} . Eq. (34) is broken down into two contributions: the $\nabla \cdot \mathbf{B}$ term yields $c_h^2 \mathbf{B}$ as the internal fluxes for ψ , whereas ψ/τ is treated as a damping source term on the right hand side of (21). The numerical fluxes at the faces are also modified: first, the interface values ψ^* and \mathbf{B}_x^* are computed using upwinding of the linear characteristics of the GLM system, following the 1D derivation of Dedner et al. (2002). The approximate MHD fluxes are then estimated using the HLLD Riemann solver. The magnetic field at the face may be set to \mathbf{B}_x^* above, or chosen to follow the prescription used for Powell terms in 4.1.3.2; in practice we found that this choice seems inconsequential and we use (29). Finally, the face fluxes for \mathbf{B}_x and ψ are corrected with the linear upwinded fluxes obtained from ψ^* and \mathbf{B}_x^* .

The parameters c_p and c_h also need to be set. In our implementation, on grids with uniform cell size Δx , c_h is chosen by adapting the procedure of Dedner et al. (2002) to DG and setting

$$c_h := \frac{C'}{2d+1} \frac{\Delta x}{D\Delta t}, \quad (35)$$

where D is the number of space dimensions, and C' is related to the CFL number C chosen in (24) and is such that $C < C' < 1$. The resulting c_h speed is faster than any other signal velocity appearing in the CFL constraint (24), but still compatible with this same stability condition for the explicit DG scheme.

For non-uniform grids however (i.e. with mesh refinement), we observed that a time-varying c_h will act as a local source of divergence, as was demonstrated and studied by Tricco et al. (2016) in the context of smoothed particle MHD. In this case, we resort to using a global and time-independent value of c_h manually adapted to each problem.

The parameter c_p is set following Dedner et al. (2002) by fixing $c_r := c_p^2/c_h = 0.18$. Note that in some cases, the resulting damping time-scale c_p^2/c_h^2 in (34) may be too small to be resolved by the timestep Δt , in which case c_p is set so that $c_p^2/c_h^2 \gtrsim 3\Delta t$. Unfortunately, all three parameters c_h, c_p, c_r above have dimensions, which makes this formulation only suitable for well-understood test problems. This can be reduced to one dimensional parameter c_h and one dimensionless parameter α following Mignone & Tzeferacos (2010), however these authors suggest that α is still slightly resolution-dependent.

To conclude on our divergence control implementations, note that in our test runs, we use exclusively either hyperbolic cleaning or Powell terms, by enabling only the corresponding additional fluxes and source terms of each selected method; in particular, our hyperbolic cleaning implementation does not use the “extended GLM” formulation of Dedner et al. (2002).

Finally, while hyperbolic divergence cleaning is implemented in a basis-independent way, we typically only use it together with the componentwise Legendre basis for the magnetic field: locally divergence-free bases are not required for hyperbolic cleaning, are slightly more computationally expensive than pure Legendre bases, and may in some cases result in slightly degraded convergence orders as discussed in Section 5.2.

4.2 Slope limiting

A major challenge for high-order codes is to limit the appearance of “ringing” artefacts around discontinuous solution features. This problem stems from Godunov’s theorem: any linear scheme of order 2 or above will introduce spurious extrema in the numerical solution. When applied to linear problems (i.e. for which the fluxes \mathbf{f}_α and source terms are linear) the RKDG scheme described so far is itself linear: the solution weights $w_l(t+\Delta t)$ are a linear combination of the weights $w_l(t)$. This problem is exacerbated for non-linear equations which can form shocks from smooth initial conditions, such as the MHD equations.

A typical workaround to this problem is to introduce in the scheme some non-linear procedure, whose role is to detect and attempt to control oscillations by locally modifying the solution. This so-called *limiting* process is recognized as a major challenge for high-order schemes (Qiu & Shu 2004; Balsara 2017). Many limiters have been proposed in the literature, first in the context of finite volume methods, but also specifically for discontinuous Galerkin schemes.

Conceptually, limiting proceeds in two successive steps (Qiu & Shu 2004): firstly, a detection procedure is used to identify “troubled cells” which are potentially subject to oscillations. Secondly, cells marked by this first procedure then have their weights w_l modified by the limiter. A high detection sensitivity (low false-negative rate) achieves scheme stability and oscillation reduction, whereas a high

specificity (low false-positive rate) ensures that the solution is not needlessly limited, which could result in effective order reduction and useless application of the potentially costly limiter procedure.

4.2.1 TVB slope limiter

In our current implementation, we use the widespread so-called total variation bounded (TVB) minmod slope limiter (Shu 1987; Cockburn et al. 1989). This limiter will detect troubled cells by comparing the slopes (linear components) of the solution in the cell with finite difference with averages in neighboring cell. When triggered, the limiting procedure downgrades the scheme to second and sometimes first order locally in the cell, altering the slopes so that the limited solution locally respects some upper bound on total variation.

We first illustrate this limiter for some scalar quantity u in 1D. In a 1D reference cell $[-1, 1]$ with coordinate ξ , u is expanded on a Legendre basis following (8). We may define the cell average \bar{u} and cell slope $u_{,\xi}$ of u by projecting u onto the space of piecewise linear functions of the form

$$\xi \mapsto \bar{u} + u_{,\xi} \cdot \xi. \quad (36)$$

Note that since $\xi \mapsto 1$ and $\xi \mapsto \xi$ are Legendre polynomials, the projection simply identifies \bar{u} and $u_{,\xi}$ with the first two weights in the basis expansion of u (up to scale factors depending only on the chosen normalization for Legendre polynomials).

The idea of the TVB detection procedure is to compare the slope $u_{,\xi}^i$ in cell i to finite difference approximations of the slope based on averages \bar{u} in the neighboring left and right cells $i - 1$ and $i + 1$:

$$\Delta_L u := (\bar{u}^{i+1} - \bar{u}^i) / \Delta\xi, \quad (37)$$

$$\Delta_R u := (\bar{u}^i - \bar{u}^{i-1}) / \Delta\xi, \quad (38)$$

where $\Delta\xi = 2$ is the size of the reference cell $[-1, 1]$. In the case of a linear solution $u = Ax + B$ in physical coordinates x , we have $u_{,\xi} = \Delta_L u = \Delta_R u = \frac{\Delta x}{2} A$ using (9). The TVB limiter first checks the absolute slope $|u_{,\xi}|$ to avoid limiting close to local extrema, where oscillations are unlikely to appear and higher order information needs to be retained. If u is smooth and admits an upper bound on its second derivative (i.e. $|\partial^2 u / \partial x^2| \leq 2M$ for some M) then within one cell of a local extremum, $|\partial u / \partial x| \leq 2M\Delta x$. Therefore, we avoid limiting cells for which $|\partial u / \partial x| \leq 2M\Delta x$, or equivalently using (9), $|u_{,\xi}| \approx |\partial u / \partial \xi| \leq M\Delta x^2$. For cells whose slope is above this threshold, TVB limiting applies a traditional total value diminishing limiter, based for example on the minmod function:

$$\text{minmod}(a, b, c) := \begin{cases} \min(a, b, c) & a, b, c > 0, \\ \max(a, b, c) & a, b, c < 0, \\ 0 & \text{otherwise.} \end{cases} \quad (39)$$

The new TVB limited slope σ is computed as

$$\sigma := \begin{cases} u_{,\xi} & \text{if } |u_{,\xi}| \leq M\Delta x^2, \\ \text{minmod}(u_{,\xi}, \beta\Delta_L u, \beta\Delta_R u) & \text{otherwise,} \end{cases} \quad (40)$$

where $\beta \in [1, 2]$ is a parameter that controls the aggressiveness of the total variation diminishing (TVD) part of the limiter (see e.g. LeVeque 2002). The detection procedure computes σ and compares it to the original solution slope $u_{,\xi}$. If $|\sigma - u_{,\xi}| > \epsilon$ for some small threshold ϵ , the limiter is marked as triggered.

For triggered cells, the limiting step assigns σ as the local solution slope by modifying the corresponding first-degree weight, and sets all second-degree or higher weights of the solution to

0. For limited cells, whenever all of $\Delta_L u$, $u_{,\xi}$ and $\Delta_R u$ have the same sign, the resulting limited weights have polynomial degree 1, and the scheme degrades to locally second-order accurate. However, whenever $\Delta_L u$, $u_{,\xi}$ or $\Delta_R u$ have conflicting signs (e.g. near extrema, or at oscillations around strong shocks), the minmod function will assign 0 to σ ; the limited solution becomes constant in the cell, and the scheme locally becomes only first-order. Note that the cell average \bar{u} is never modified by the limiter; this ensures that the limiting process stays conservative. In all of this work, we take $\epsilon = 10^{-8}$, and we use $\beta = 2$, which corresponds to a monotone central limiter¹.

Equation (40) prescribes a limiter whose minmod function gets applied almost everywhere, except very close to extrema as defined by the parameter M . For a fixed Δx , one can always find a suitable value of M , but across resolutions, we find that we obtain more consistent results if we let M scale with Δx , in a prescription similar to Schaal et al. (2015). We therefore define $\tilde{M} := M\Delta x$, and choose to keep \tilde{M} a constant instead of M in (40). This has the effect of making the limiter weaker at higher resolutions, which we found to be important for divergence control in the Powell scheme. We come back to this non-intuitive aspect of the limiter in Section 6.3.2 in the discussion.

In 2D and 3D, the limiter is simply applied in each space direction independently, acting only on one of the 2 or 3 directional slopes at a time.

4.2.2 Characteristic limiting

To apply this scalar slope limiter to systems of equations, we can in principle apply the limiter to each scalar conserved variable successively, downgrading the local cell accuracy to second order or less whenever one of the components triggered the limiter.

Another option is to compute a local decomposition into characteristics, applying the limiter on characteristic variables instead. In this approach, the slopes $\Delta_L u$, $u_{,\xi}$, $\Delta_R u$ are first transformed to characteristic slopes, limited using the TVB procedure, and the resulting limited slope σ is transformed back to conserved variables. For a given spatial direction along which to perform the 1D limiting process, the characteristic variables Δa are obtained from the conserved differences (or slopes) Δu with

$$\Delta a = \mathbf{L}(\bar{u}) \cdot \Delta u, \quad (41)$$

where $\mathbf{L}(\bar{u})$ is the matrix of left eigenvectors of the conserved 1D MHD equations along the chosen direction, linearized around the average state \bar{u} in the central cell. After limiting Δa as described in 4.2.1, we obtain the limited conserved variables using the matrix $\mathbf{R}(u) = \mathbf{L}(u)^{-1}$ of right MHD eigenvectors:

$$\Delta u = \mathbf{R}(\bar{u}) \cdot \Delta a. \quad (42)$$

Suitable expressions for numerical evaluation of \mathbf{L} and \mathbf{R} may be found e.g. in Powell et al. (1999) or Stone et al. (2008). Jiang & Wu (1999) conveniently summarize how to obtain eigenvectors for the divergence wave of the Powell scheme from conserved variables,

¹ Note that using $\beta = 1$ with the TVB limiter is not a suitable choice in practice. As resolution is increased, the $M\Delta x^2$ threshold will protect only $O(1)$ cells around local extrema from the minmod function, and as a result the latter will end up being invoked almost everywhere in the domain. If $\beta = 1$, then $\text{minmod}(a, b, c) \neq a$ almost everywhere, which will end up triggering the limiter on almost all cells of the domain, degrading the scheme to second order at best.

and [Dedner et al. \(2002\)](#) cover the extension of the characteristic matrices for hyperbolic divergence cleaning.

Characteristic limiting is well motivated by the description of local variations as a superposition of local linearized physical waves, and is generally recognized as yielding better results than conserved variable limiting ([Qiu & Shu 2004](#); [Balsara 2017](#)). Our numerical experiments are consistent with these earlier results, and we therefore systematically resort to limiting characteristic variables. In particular, we find that conserved variable limiting can result in post-shock oscillations and noise, readily visible for example on the Orszag–Tang vortex test problem, whereas characteristic limiting results both in sharp shocks and noise-free solutions in smooth regions of the flow.

For each independent limiting space direction α , we compute the 1D characteristics along direction α with the appropriate 1D matrices \mathbf{L} and \mathbf{R} , and apply the scalar limiting described in 4.2.1 to each characteristic variable independently.

The main drawback of characteristic limiting is the expensive construction and application of the matrices \mathbf{L} and \mathbf{R} for each cell; in practice the cost can be amortized by processing multiple characteristic matrices within a same inner loop, which allows the efficient use of vector CPU instructions.

4.2.3 Choice of limiter threshold

We conclude by noting that suitably choosing the parameters M or \tilde{M} is central to the good performance of this limiter. Unfortunately, irrespective of the choice of limiter threshold scaling prescription, neither M nor \tilde{M} are dimensionless, and their optimal value depends on the initial conditions, spatial resolution, and choice of units.

For simple scalar problems with smooth initial conditions, M may be interpreted as an upper bound on second derivatives in the initial conditions ([Cockburn et al. 1989](#)). Finding good a priori choices of M for systems is more complicated ([Qiu & Shu 2004](#)). In the case of conserved variable limiting, each variable formally has different units, and it is unclear whether a single numerical value of M may apply to all components meaningfully. In the case of characteristic limiting, the normalization of characteristic variables depends on the arbitrary normalization of the eigenvectors, making M dependent on this choice as well. \tilde{M} is related to admissible gradients in the solution, and is subject to the exact same shortcomings of units and normalization.

Despite these issues, we use this simple limiter based on \tilde{M} as a starting point, and discuss some possible promising alternatives in Section 6.4.1.

4.2.4 Slope limiting with LDF bases

At the end of the limiting process, whenever the slope limiter was triggered and has modified the weights in the cell, we are about to reset the slopes of the fields in the cell, and clear the higher-order moments in the weights. When using LDF bases, some additional caution is required: after the TVB slope limiter step operates in each space direction β independently, the limited slopes $\mathbf{B}_{\alpha,\beta}$ of each component \mathbf{B}_α of the magnetic field may not be divergence-free, i.e. we may have

$$\delta := \sum_{\alpha} \mathbf{B}_{\alpha,\alpha} \neq 0. \quad (43)$$

If $\delta \neq 0$, then the resulting slopes are not representable on a LDF basis. One could simply think of projecting back new slopes onto the divergence-free basis using L^2 projection; however, this procedure

has a number of flaws in the general case where $\delta \neq 0$. First, this L^2 projection is not total variation diminishing, and may reintroduce local extrema in the magnetic field that were just taken away by the limiting process. In particular, it may generate nonzero slopes of \mathbf{B} in directions along which the magnetic field was initially uniform, thereby breaking symmetries by coupling of spatial directions. Finally, L^2 projection may create higher-order contributions (degree 3 and above) in the divergence-free weights, which is not desirable in a limited cell; this particular issue is discussed in more detail in Appendix B.

To solve this problem, we use a simple non-linear procedure applicable to any number of dimensions. We start from limited slopes $\mathbf{B}_{\alpha,\beta}$ obtained from any slope limiting procedure; in general those will have $\delta \neq 0$. Note that the off-diagonal slopes $\mathbf{B}_{\alpha,\beta}$ with $\alpha \neq \beta$ do not contribute to δ , and therefore do not require any correction. Let $\sigma_\alpha := \mathbf{B}_{\alpha,\alpha}$ be the limited diagonal slopes before divergence correction. Our goal is to derive corrected slopes $\tilde{\sigma}_\alpha$ verifying $\tilde{\delta} = \sum_{\alpha} \tilde{\sigma}_\alpha = 0$.

We start by separating the divergence δ into positive and negative contributions δ^+ and δ^- :

$$\delta^\pm := \pm \sum_{\alpha} \max(\pm \sigma_\alpha, 0), \quad (44)$$

such that $\delta = \delta^+ - \delta^-$, with $\delta^\pm \geq 0$.

Suppose the slopes $\mathbf{B}_{\alpha,\beta}$ have divergence $\delta > 0$; then δ^+ is in excess compared to δ^- and we have $\delta^+ > \delta^- \geq 0$. In this case, we rescale the slopes which contribute to δ^+ (i.e. diagonal slopes which are positive) by the appropriate factor to exactly cancel out the divergence. We set:

$$\text{For } \delta > 0, \quad \tilde{\sigma}_\alpha := \begin{cases} \sigma_\alpha & \text{if } \sigma_\alpha \leq 0, \\ (\delta^-/\delta^+) \sigma_\alpha & \text{if } \sigma_\alpha > 0, \end{cases} \quad (45)$$

so that we obtain $\tilde{\delta}^+ = \tilde{\delta}^-$. Similarly, in case of negative divergence, we set:

$$\text{For } \delta < 0, \quad \tilde{\sigma}_\alpha := \begin{cases} \sigma_\alpha & \text{if } \sigma_\alpha \geq 0, \\ (\delta^+/\delta^-) \sigma_\alpha & \text{if } \sigma_\alpha < 0. \end{cases} \quad (46)$$

It is easy to check that with this prescription, for any initial value of δ , we obtain $\tilde{\delta} = 0$. Because $\delta > 0 \Rightarrow 0 \leq \delta^-/\delta^+ < 1$ and $\delta < 0 \Rightarrow 0 \leq \delta^+/\delta^- < 1$, we always have $|\tilde{\sigma}_\alpha| \leq |\sigma_\alpha|$, and the procedure is total variation diminishing.

Finally, we assign the newly obtained divergence-free slopes $\tilde{\sigma}_\alpha$ to the $\mathbf{B}_{\alpha,\alpha}$, and we obtain the limited magnetic field weights by L^2 projection of the resulting second-order solution $\mathbf{B}_\alpha(\xi) = \tilde{\mathbf{B}}_\alpha + \sum_{\beta} \mathbf{B}_{\alpha,\beta} \xi_\beta$ onto the LDF basis functions. Since this second-order solution is now divergence-free, this last projection is exact and does not suffer from any of the issues mentioned at the beginning of this section.

4.3 Enforcing density and pressure positivity

In the conserved variables formulation, the thermal pressure is derived from the total energy by subtracting the kinetic and magnetic energy terms. Under strong shock conditions, this can result in unphysical negative thermal pressure p at quadrature points. This is also true for the density which, although represented exactly in the conserved variables, can still suffer from high-order oscillations in rarefied regions.

Schemes have been developed to try to maintain positivity of pressure and density in the context of higher-order finite volume methods. A possible solution revolves around rewriting the conserved system as a conservation law for some appropriately defined

entropy variables. This solution was adopted by [Ryu et al. \(1993\)](#) in the context of the Euler equations for cosmological simulations, and later extended to MHD by [Balsara & Spicer \(1999a\)](#). Combining this idea with appropriate Riemann solvers for the numerical fluxes, one can then construct finite volume schemes with some positivity properties. [Chandrashekar & Klingenberg \(2016\)](#) developed an entropy-stable finite volume scheme for MHD with a prescribed numerical flux and Powell term, and other authors have further developed this approach, see e.g. [Winters & Gassner \(2016\)](#); [Derigs et al. \(2016\)](#).

In practice, positivity can however generally only be proven for a limited set of schemes, in the absence of general source terms and relying on selective numerical flux prescriptions. In addition, as noted by [Balsara & Spicer \(1999a\)](#), discretization or round-off errors can still contribute to causing negative states even with positivity-preserving strategies in place.

For this work, we follow the approach used for hydrodynamics in [Schaal et al. \(2015\)](#) and adopt the general framework of [Zhang & Shu \(2010\)](#) for positivity limiting in a DG context.

Our positivity limiter proceeds in two steps to modify the cell weights w . In a first step, the cell averages (represented by 0th-degree weights \bar{w}) are checked for positive density and pressure. If positivity is satisfied, then the cell weights w are left unchanged. If the cell averages violate positivity, then w is modified by setting the average density and/or pressure to predefined floor values ϵ_ρ and ϵ_p . This operation is not conservative: it modifies the total amount of mass and energy in the simulation, and should therefore be viewed as a last resort effort to keep the simulation running. We keep track of the amount of conserved quantities injected by this stage. We find that this criterion is never triggered in practice with the Powell scheme in any of the test problems presented in this paper, but can in some cases be triggered using hyperbolic divergence cleaning with strong shocks in very low plasma- β situations.

Once positivity of the cell average state \bar{w} is guaranteed, we follow the general idea of [Zhang & Shu \(2010\)](#) by computing a scaling factor $\tau \in [0, 1]$ such that the cell state defined by the weights

$$\tilde{w}(\tau) := \tau w + (1 - \tau)\bar{w} \quad (47)$$

has positive density and pressure at all *cell* and *face* quadrature points. This prescription effectively scales the amplitude of the spatial variations of the cell state between $\tau = 1$ (which yields $\tilde{w} = w$ and recovers the full unlimited state) and $\tau = 0$ (corresponding to $\tilde{w} = \bar{w}$, i.e. a piecewise-constant solution in the cell, which is positive everywhere by construction). Ideally, we would like to find the maximal (i.e. least impacting) value of τ such that we have both:

$$\rho(\tilde{w}(\tau)) \geq \epsilon_\rho \quad (48)$$

$$p(\tilde{w}(\tau)) \geq \epsilon_p \quad (49)$$

at each individual quadrature point of the cell.

Since the density is a linear function of the weights, solving (48) for τ at a given quadrature point q with local density ρ_q simply yields:

$$\tau_q = \frac{\bar{\rho} - \epsilon_\rho}{\bar{\rho} - \rho_q}. \quad (50)$$

Taking τ to be the smallest τ_q over all quadrature points q which require positivity limiting (i.e. where $\rho_q < \epsilon_\rho$) guarantees that (48) is fulfilled.

The situation for the pressure positivity (49) is more complicated, because p is not a linear function of the conserved quantities \mathbf{u} and therefore of the weights w . In the case of hydrodynamics,

solving (49) requires finding the roots of a quadratic polynomial at each non-positive quadrature point, which is the method adopted by [Zhang & Shu \(2010\)](#); [Schaal et al. \(2015\)](#). For MHD however, this approach requires cubic root finding due to the magnetic energy term in the total energy, which is both costly and challenging to implement in a numerically robust way. [Wang et al. \(2012\)](#) noted that we can easily obtain a value of τ which satisfies (49) by exploiting the fact that $p(\mathbf{u})$ is a concave function of the conserved state \mathbf{u} , i.e.:

$$\forall \tau \in [0, 1], \quad p(\tau \mathbf{u} + (1 - \tau)\bar{\mathbf{u}}) \geq \tau p(\mathbf{u}) + (1 - \tau)p(\bar{\mathbf{u}}), \quad (51)$$

which can be easily checked by noting that the eigenvalues of the Hessian of $\mathbf{u} \mapsto p(\mathbf{u})$ are all negative. Therefore, choosing

$$\tau_q = \frac{\bar{p} - \epsilon_p}{\bar{p} - p_q} \quad (52)$$

guarantees that (49) holds at quadrature point q . Even though this τ is not necessarily maximal, this prescription retains the space-varying properties of the solution while ensuring positivity, and we found it to be very efficient and much more stable numerically than iterative root finding.

We can now pick a single value of τ which satisfies simultaneously (48) and (49) at all quadrature points, by taking the smallest of all the τ_q prescribed by both (50) and (52). In the case of a cell which does not need any limiting because its conserved state is positive at all quadrature points, we set $\tau = 1$. Finally, we use this single τ to update the weights in the cell by setting $w \leftarrow \tilde{w}(\tau)$ using (47) whenever we end up with $\tau < 1$.

Finally, note that our practical implementation differs from [Zhang & Shu \(2010\)](#); [Schaal et al. \(2015\)](#) in two additional important details. First, instead of introducing a separate set of Gauss-Legendre-Lobatto points for positivity limiting, we enforce positivity over all face and volume Gauss-Legendre quadrature points—the exact same points used in the computation of the face and volume terms in the DG scheme. This avoids issues related to round-off errors, which may arise from enforcing positivity and evaluating the conserved state at different quad points, to which MHD seems particularly prone. In addition, [Zhang & Shu \(2010\)](#) suggest reducing the Courant number whenever the positivity limiting procedure is enabled. In practice, we find that we do not need to modify the timestep criterion for our test problems, as long as the limiting is applied consistently at each of the Runge-Kutta substeps before computing any conserved state at quadrature points.

4.4 Adaptive mesh refinement

In order to capture a large dynamic range for astrophysical applications, our code provides tree-based spatial adaptive mesh refinement, in which each cell may be refined or de-refined independently based on local criteria. A refinement operation consists in a local splitting of the parent cell into 8 children cells (in 3D), which results in an octree-structured grid.

The design of the AMR algorithm broadly follows the adaptive mesh implementation of RAMSES described in [Teyssier \(2002\)](#). The DG framework provides a clean setting for defining prolongation (refinement) and restriction (de-refinement) operations, through the use of local L^2 projections onto basis functions. Our MHD implementation is identical to the pure hydrodynamics implementation discussed in detail in [Schaal et al. \(2015\)](#), except for the change in basis functions when using LDF bases. In this paper, we do not particularly focus on adaptive mesh refinement, but provide an illustration in Fig. 12 of the Orszag–Tang vortex run with AMR, to

demonstrate that the method works with this demanding test problem.

5 RESULTS

5.1 General test problem setup

In the following section, we present test problems run using our DG schemes at various orders. Specific care has been taken to ensure that the test problems are run with consistent and homogeneous settings, without problem-specific fine tuning,

In all test problems, we use HLLD as the approximate Riemann solver. Except for the convergence order tests, positivity limiting is enabled with density and pressure floors $\epsilon_\rho = \epsilon_p = 10^{-12}$, and we limit the slopes using the characteristic slope limiter, for which the modified threshold parameter \tilde{M} is always set to $\tilde{M} = 5$. While this sometimes results in sub-optimal limiting, we feel that it achieves a good compromise and presents an honest picture of the capabilities of the code across a range of test problems without any limiter fine-tuning.

In the following, “DG- $(d + 1)$ ” designates the DG method with degree $\leq d$ basis polynomials—whose spatial convergence order is typically $d + 1$. Unless otherwise specified, the results are presented for degree $d = 2$ polynomials, i.e. the 3rd order scheme DG-3, and using the Powell scheme for divergence control. By default, we match the order of the Runge-Kutta time integration to the spatial scheme order $d + 1$, up to a maximal order of RK4. Comparisons with hyperbolic cleaning are shown whenever they are informative or show relevant differences. Details of the divergence control schemes are discussed in Section 4.1.

Most of the test problems shown are computed on 2D or 3D Cartesian grids, for which the resolution level ℓ corresponds to 2^ℓ grid points per dimension. We also illustrate the use of our scheme with adaptive mesh refinement in Fig. 12.

5.2 Convergence order tests

We first present tests of the spatial convergence order of the code. Carefully measuring convergence of higher-order MHD codes is surprisingly challenging, as smooth MHD test problems present numerical subtleties that are revealed by the very low numerical diffusion of higher-order methods. We rely on widely-used smooth MHD test problems for convergence assessment: the so-called isodensity vortex in 2D, and non-linear circularly polarized Alfvén waves in 2D and 3D.

For the smooth convergence order tests of this Section 5.2, we disable the slope and positivity limiters, to ensure that the limiters do not interfere with effective order measurement. Note that given a problem with a smooth solution and a desired resolution, the slope limiter setting M (or equivalently \tilde{M}) may always be set in such a way that the limiter will never trigger.

5.2.1 Computing errors against analytical solution

To study the convergence properties of the code, we compute the L^p error between a numerical scalar function f and its reference solution f_{ref} over the simulation volume V as

$$\|f - f_{\text{ref}}\|_p = \left[\frac{1}{V} \int_V |f(\mathbf{x}) - f_{\text{ref}}(\mathbf{x})|^p d\mathbf{x} \right]^{\frac{1}{p}}. \quad (53)$$

The integral is computed using Gaussian quadrature over each cell. As described in Schaal et al. (2015), we use a higher number of points for this quadrature rule ($d + 3$ instead of $d + 1$ used in the numerical scheme) to account for the fact that $|f(\mathbf{x}) - f_{\text{ref}}(\mathbf{x})|^p$ will generally not be a polynomial of degree $\leq 2d$. This simple prescription ensures that the error calculated from (53) in our convergence tests will not be dominated by errors from the quadrature in the norm itself.

5.2.2 Isodensity MHD vortex in 2D

We first consider the so-called isodensity MHD vortex of Balsara (2004). This problem follows the evolution of a stationary magnetized vortex crossing a periodic domain $[-10, 10]^2$, advected with a background flow until final time $t_f = 20$, after which it will have returned to its initial location at the center of the domain. We broadly follow the setup of Li & Shu (2005). We set $\gamma = 5/3$, and initialize the unperturbed background flow with $\rho = 1$, $p = 1$, velocity $\mathbf{v} = \mathbf{v}_0 = \sqrt{2}(\cos \alpha, \sin \alpha, 0)$ and $\mathbf{B} = 0$. Most authors including Li & Shu (2005) set $\alpha = 45^\circ$, however we choose to test both $\alpha = 45^\circ$ and $\alpha = 30^\circ$ to test 2D configurations not aligned with the grid. Letting $r^2 = x^2 + y^2$, the velocity and magnetic field are perturbed according to

$$\delta \mathbf{v} = \frac{\eta}{2\pi} \nabla \times \left(0, 0, e^{(1-r^2)/2} \right), \quad (54)$$

$$\delta \mathbf{B} = \frac{\xi}{2\pi} \nabla \times \left(0, 0, e^{(1-r^2)/2} \right). \quad (55)$$

Dynamical equilibrium is achieved by correcting the pressure following

$$\delta p = \left[\xi^2(1 - r^2) - \eta^2 \right] \frac{1}{8\pi^2} e^{1-r^2}. \quad (56)$$

We use $\eta = \xi = 1$. The vortex is simply advected in an equilibrium configuration, so the solution at any time t is found by $\mathbf{u}(\mathbf{x}, t) = \mathbf{u}(\mathbf{x} - \mathbf{v}_0 t, 0)$.

Note that it is important to take a large enough domain, so that the perturbations are negligible at the domain border when setting up initial conditions, otherwise waves will appear at the periodic boundaries. This issue was studied in the context of higher-order Euler codes by Spiegel et al. (2015) for the similar isentropic vortex test.

Fig. 4 presents the L^2 solution errors after the vortex has crossed the box at time $t = 20$. Results are shown for the x component of the magnetic field and pressure (top and bottom rows), and for the Powell scheme with LDF basis and Legendre basis with hyperbolic cleaning (left and right columns). Solid and dashed lines correspond to errors for an advection angle $\alpha = 45^\circ$ and $\alpha = 30^\circ$ respectively. The gray shaded area shows the approximate range of problem resolutions across which the vortex structure is resolved with at least 2 cells (lower resolution limit), but not over-resolved, in the sense that the pressure fluctuation δp changes by 1% or more over at least one cell (upper resolution limit). This is the regime most interesting for science applications, where the spatial resolution is dynamically adapted to the feature size of interest.

We find that the code achieves the expected convergence order overall. When using LDF basis functions however, the effective order of convergence is degraded with $\alpha = 30^\circ$ in the case of over-resolved solutions. This effect is mostly visible on the pressure, in the lower left panel of Fig. 4. We speculate that this effect is caused by some instability related to the LDF bases, whose discussion we postpone to the end of this section.

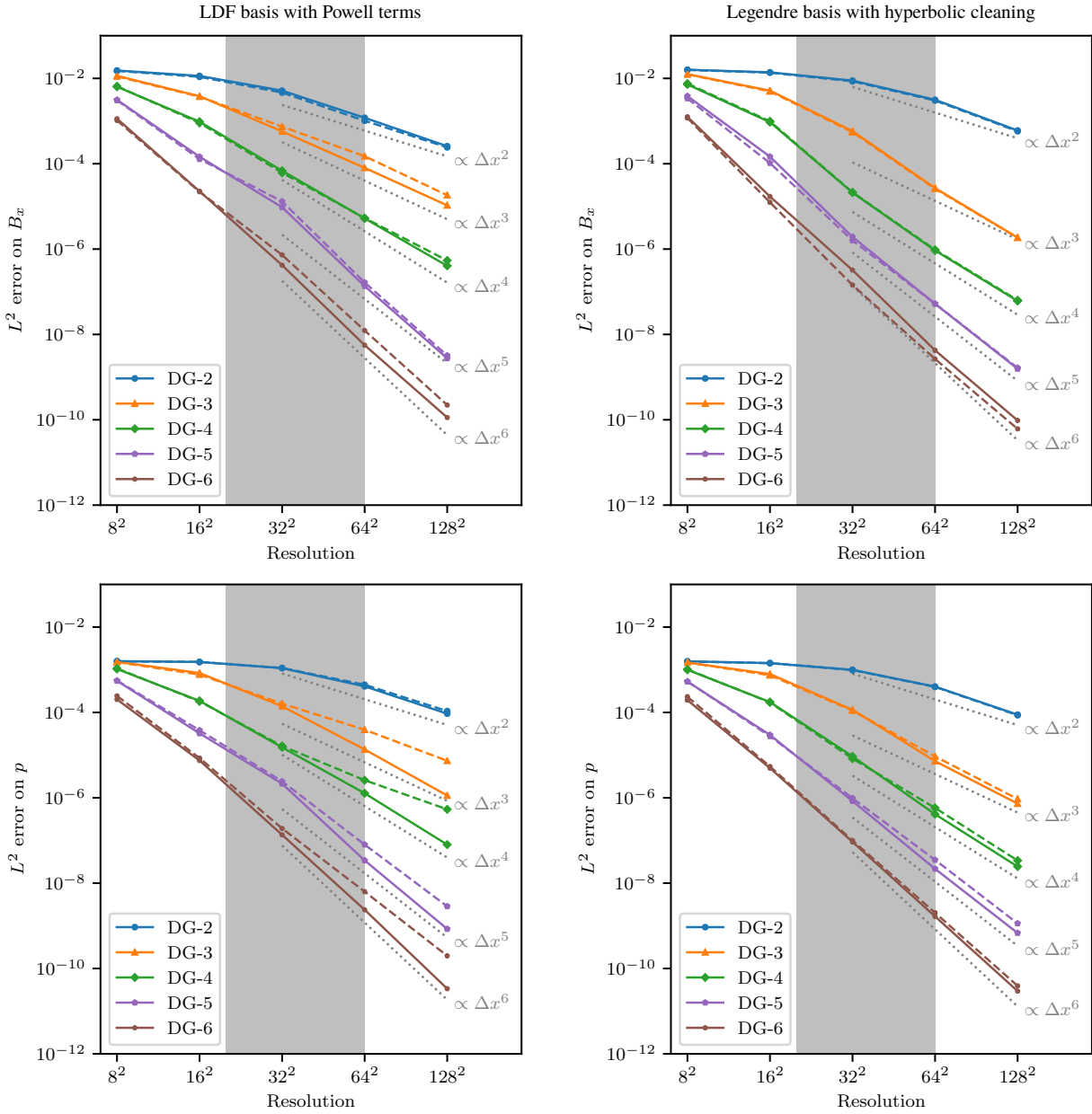


Figure 4. Convergence of errors in the MHD isodensity vortex problem. Solution L^2 errors are plotted for B_x (top row) and pressure (bottom row), for both the LDF basis with Powell terms (left column), and Legendre basis with hyperbolic cleaning (right column). Errors are measured at time $t = 20$ after the vortex has crossed the whole computational domain. Dotted lines show theoretical slopes for convergence orders 2 to 6. Solid and dashed lines correspond to errors for an advection angle $\alpha = 45^\circ$ and $\alpha = 30^\circ$ respectively. With the Legendre basis, the errors are insensitive to the vortex advection angle. With the LDF basis, the effective convergence order generally degrades slightly with $\alpha = 30^\circ$, in particular for the pressure. The shaded area corresponds to a range of resolutions for which the vortex is resolved but not over-resolved (see discussion in the text).

We first comment on the evolution of the global numerical divergence present in the solution as a function of scheme order and grid resolution. Fig. 5 presents the L^2 norm of the magnetic field divergence for the smooth vortex problem at final time $t = 20$, for both the Powell and hyperbolic cleaning schemes. Numerical divergence is computed as described in Appendix C. We find that the divergence indeed asymptotically converges to zero with resolution at approximately the expected order, however, the effect of resolution is very sensitive to the resolved character of the solution for this smooth problem. For resolved smooth solutions (right of the gray band in Fig. 5), higher order schemes provide a more accurate

representation of the solution, thereby reducing the discontinuity of the normal component of the magnetic field and allowing the global divergence of \mathbf{B} to converge to zero with order and resolution. For unresolved solutions on the other hand (left of the gray band in Fig. 5), additional resolution does not translate into reduction of the divergence, particularly at low scheme orders. We note that both Powell and hyperbolic cleaning techniques exhibit very similar convergence rates of $\nabla \cdot \mathbf{B}$ on this problem, however, hyperbolic cleaning consistently results in global divergence levels about one order of magnitude below the Powell scheme.

To conclude with this test problem, we return to the conver-

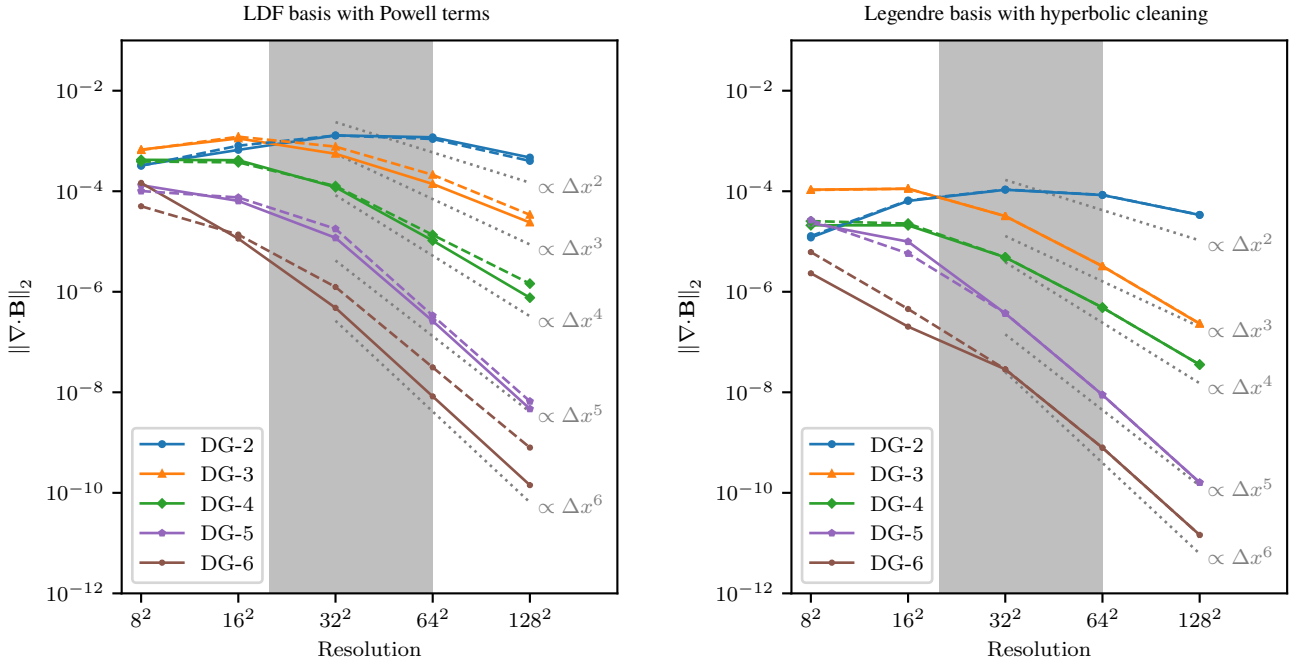


Figure 5. Convergence of $\nabla \cdot \mathbf{B}$ in the MHD isodensity vortex problem. The L^2 norm of the divergence of the magnetic field is shown at $t = 20$ for the LDF basis with Powell terms (left column), and Legendre basis with hyperbolic cleaning (right column). Similarly to Fig. 4, dotted lines show theoretical slopes for convergence orders 2 to 6, solid and dashed lines correspond to $\alpha = 45^\circ$ and $\alpha = 30^\circ$ respectively, and the shaded area corresponds to a range of resolutions for which the vortex is barely resolved.

gence order degradation observed in Fig. 4 with the Powell scheme for $\alpha = 30^\circ$. It is tempting to point at the lack of divergence damping with the Powell scheme as a possible culprit, because for such schemes, divergence errors are known not to converge away with resolution for discontinuous solutions (Tóth 2000). Dumbser et al. (2008) also suggested that high-order methods alone are not enough to achieve optimal convergence without some form of divergence cleaning. However, for this problem, Fig. 5 shows no significant difference between the Powell and hyperbolic cleaning schemes in terms of convergence rate of the divergence. In addition, for our experiments, the convergence degradation also appears with hyperbolic cleaning when using LDF bases—whereas hyperbolic cleaning with Legendre bases is immune, as Fig. 4 demonstrates. This suggests that the effect is connected to the LDF bases, and not caused by the Powell treatment alone.

Instead, we suspect that this deterioration is caused by the joint use of locally divergence-free basis functions, together with the continuous treatment of the normal component of \mathbf{B} in the chosen HLLD Riemann solver. While the detailed mechanics of this effect remain unclear at this point, we speculate that projection of source terms (in particular at the faces) onto LDF bases could in some cases contaminate high order modes (see Appendix B). Potential ways to alleviate this issue with HLLD fluxes could be to rely on an 8-wave version of this Riemann solver, for example the one developed by Fuchs et al. (2011), or to combine different flux functions based on local smoothness of the solution (see e.g. Derigs et al. 2016). We leave these investigations to future work.

At this point, we also wish to stress that this convergence order degradation does not appear to be of great practical importance at this stage, because the errors are still well behaved and the beneficial effects of order convergence are still present, especially in the shaded area where the problem is marginally resolved: at resolutions

around 32^2 , order convergence helps capture the vortex much more efficiently than spatial convergence, with both choices of divergence control. It is of course possible that this effect is only the early-time manifestation of a more serious long-term instability, however we have seen no indication of such problems in our test runs so far. We believe these results demonstrate the necessity and value of testing multidimensional schemes in non-grid-aligned configurations (i.e. $\alpha = 30^\circ$ in this case).

5.2.3 Circularly polarized Alfvén wave problem in 2D

Circularly polarized Alfvén waves are simple, exact smooth analytic solutions of the MHD equations for any wave perturbation amplitude. They are therefore particularly suitable to study the convergence order of the scheme, as well as its amount of numerical dispersion and dissipation. They consist of plane waves in which the magnetic field and velocity oscillate in phase in a circular polarization perpendicular to the propagation direction.

We broadly follow the traveling wave setup proposed by Tóth (2000). Using periodic boundary conditions on a square domain $[0, L]^2$, we construct an Alfvén plane wave of unit wavelength, propagating at angle $\alpha > 0$ with the x axis. In order to respect periodicity, we pick α and L such that the box extends for exactly 1 wavelength in the y direction, and m wavelengths in the x direction (with $m > 0$ an integer). This implies $\alpha = \arctan(1/m)$ and $L = 1/\sin \alpha$. We pick $m = 2$ which yields $\alpha \approx 26.6^\circ$, $L \approx 2.24$. This ensures that the problem is truly two-dimensional. Calling (\parallel, \perp, z) the rotated (x, y, z) frame so that the wave propagates along the \parallel direction, the phase of the wave at any point (x, y) is given by $\phi = 2\pi x_{\parallel}$, with $x_{\parallel} = x \cos \alpha + y \sin \alpha$. The magnetic field is initialized as $(B_{\parallel}, B_{\perp}, B_z) = (1, \epsilon \sin \phi, \epsilon \cos \phi)$, and the initial velocity is set to $(v_{\parallel}, v_{\perp}, v_z) = (v_0, \epsilon \sin \phi, \epsilon \cos \phi)$. We set $\epsilon = 0.1$ and take

$v_0 = 0$, which produces waves travelling along the \parallel direction towards negative x_{\parallel} at the Alfvén velocity $v_A = B_{\parallel}/\sqrt{\rho} = 1$. Density and pressure are uniform with $\rho = 1, p = 0.1$, and we take $\gamma = 5/3$. Note that the magnetic pressure is uniform thanks to the B_{\parallel} and B_{\perp} components being in quadrature, and exact pressure equilibrium is achieved.

The analytic solution to this problem at any time t for the complete state vector \mathbf{u} is simply $\mathbf{u}(x, y, t) = \mathbf{u}_0(x - (v_0 - v_A)t \cos \alpha, y - (v_0 - v_A)t \sin \alpha)$ where the signs of $-v_A$ reflect the fact that the wave travels towards negative x_{\parallel} .

Unfortunately, although it is an exact solution of non-linear MHD, this setup is subject to parametric instabilities, which are easily triggered by low dissipation high-order schemes (see e.g. the discussion in Balsara et al. 2009, and references therein). In practice, this restricts convergence order tests to low resolutions, before the instability starts picking up and dominating the very low errors of high order schemes. We present results with up to 64 grid points per dimension, which is comparable to the maximum resolution shown by Balsara et al. (2009).

Fig. 6 presents the L^2 solution errors for the x component of the magnetic field, after the wave has crossed one wavelength at time $t = 1$. The convergence order generally follows the theoretical order of convergence for both divergence control schemes. Note that at 64^2 resolution, the error of the 5th order scheme starts leveling off at around 10^{-8} , which we attribute to the triggering of the aforementioned parametric instability; Balsara et al. (2009) found evidence for this effect in their 4th order code at comparable resolutions in their 3D Alfvén wave problem. Note also that our Powell scheme—which relies on LDF bases—suffers from a slightly degraded convergence rate for the fourth-order DG-4 scheme compared to hyperbolic cleaning. We deem that this effect could be related to the observed deterioration of the convergence order for the isodensity MHD vortex of Section 5.2.2.

5.2.4 Circularly polarized Alfvén wave problem in 3D

To test convergence order of the code in 3D, we run a comparable Alfvén wave setup, partially based on Balsara et al. (2009). For this problem, the domain is $[0, 1]^3$ with periodic boundary conditions in all three directions. We take a uniform background density $\rho = 1$ and thermal pressure $p = 100$.

A rotated coordinate system (x', y', z') is setup so that the x' direction is aligned with the cube diagonal, following the exact transformation described in Balsara et al. (2009). Letting ϕ be the phase angle of the wave, the velocity field in the primed reference frame is $\mathbf{v}' = (1, \epsilon \cos \phi, \epsilon \sin \phi)$, while the magnetic field is $\mathbf{B}' = (1, -\epsilon \cos \phi, -\epsilon \sin \phi)$, for which we pick $\epsilon = 0.02$. The unprimed velocity and magnetic fields are obtained by transforming back to unprimed coordinates.

The resulting wave travels at speed 2 along x' , since it has Alfvén velocity 1 and is also advected with the flow with velocity 1 along x' due to our choice of \mathbf{v}' . The phase angle ϕ is therefore given by:

$$\phi = \frac{2\pi}{\lambda}(x' - 2t). \quad (57)$$

Choosing the wavelength $\lambda = 1/\sqrt{3}$ ensures that the problem is periodic in all 3 directions x, y and z , since whenever we cross one box length in one of these directions, x' shifts by $1/\sqrt{3}$. With this choice of λ , the cube diagonal crosses 3 wave periods. The problem is run until time $t_f = \sqrt{3}/2$, after which the 3 wave fronts will have crossed the whole domain once.

Fig. 7 shows the convergence of the errors on the B_x component with the Powell and hyperbolic cleaning schemes for scheme orders 2 to 4. Both schemes follow overall the expected order of convergence.

5.3 Shock problems

We now turn to test problems in 1D, 2D and 3D designed to test the shock capturing properties of the scheme.

5.3.1 Brio-Wu shock tube

The famous shock tube problem introduced by Brio & Wu (1988) has now become a classic of shock tests for MHD codes. For this test, we take the computational domain to be $[0, 1]$ with fixed boundary conditions at $x = 0$ and $x = 1$. In the whole domain, the flow is initially at rest ($\mathbf{v} = 0$) and $(B_x, B_z) = (0.75, 0)$. The initial primitive variables are discontinuous at $x = 0.5$, with the left and right states given by $(\rho, p, B_y)_L = (1, 1, 1)$ and $(\rho, p, B_y)_R = (0.125, 0.1, -1)$ respectively. We set $\gamma = 2$, and run the simulation until final time $t_f = 0.1$. Note that we actually run our 1D test problems in 2D, with perfectly y -independent initial conditions and periodic boundary conditions in y , in order to test the numerical stability of the 1D setup. We check that no significant y dependence of the solution has developed at the final time.

Fig. 8 presents the density, pressure, and y component of the magnetic field in the Brio-Wu shock tube test problem at final time $t_f = 0.1$, using the third-order DG-3 scheme at resolution level $\ell = 9$ (corresponding to 512 cells). The reference solution was obtained using Athena (Stone et al. 2008) with the third-order Roe solver on 10^4 mesh points. The DG-3 scheme captures all the MHD waves correctly, and the shocks are resolved with 1–2 cells, whereas the contact discontinuity is resolved within about 4 cells. The limiter is efficient at preventing overshoots around shocks, although some oscillations are visible around the contact discontinuity and after the right rarefaction wave, probably due to a rather lenient global limiter setting used across all of our tests. A more finely tuned or more sophisticated limiter could help reducing these oscillations. We note however that we observed identical oscillations of similar amplitude and wavelength with Athena with HLLD and third-order reconstruction at the same resolution of 512 cells.

5.3.2 1D MHD Shu-Osher test problem

The so-called hydrodynamical Shu-Osher test problem, introduced by Shu & Osher (1989), follows the interaction of a supersonic shockwave with smooth density perturbations. It tests the scheme's ability to resolve small-scale features in the presence of strong shocks. An MHD version of the test was proposed by Susanto (2014), and used in particular by Derigs et al. (2016) to test their MHD scheme in FLASH. We follow the setup of these two papers. The computational domain is $[-5, 5]$ with reflective boundary conditions. At $t = 0$, the shock interface is located at $x_0 = -4$. In the region $x \leq x_0$, a smooth supersonic flow is initialized with primitive state given by $(\rho, v_x, v_y, v_z, p, \mathbf{B}_x, \mathbf{B}_y, \mathbf{B}_z) = (3.5, 5.8846, 1.1198, 0, 42.0267, 1, 3.6359, 0)$. In the rest of the domain $x > x_0$, smooth stationary perturbations are setup with primitive state $(1 + 0.2 \sin 5x, 0, 0, 0, 1, 1, 1, 0)$. The flow is evolved until final time $t_f = 0.7$.

The density profile at t_f is shown in Fig. 9, for resolution levels 7 (128 cells) and 8 (256 cells). To compare the results across

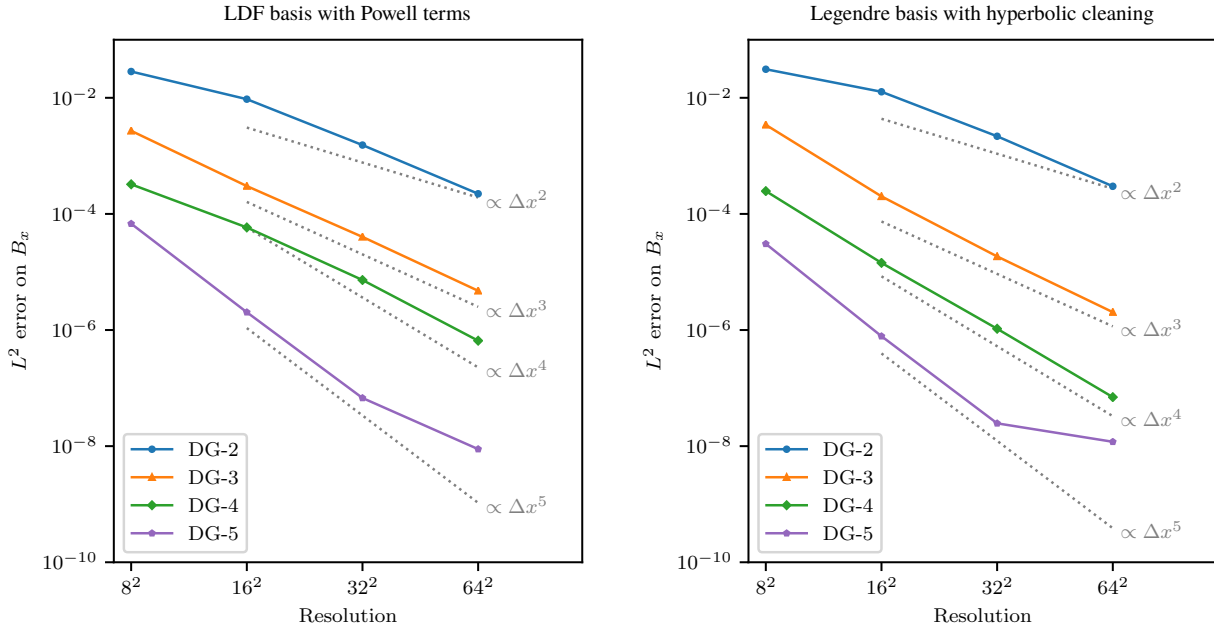


Figure 6. Convergence of errors in the 2D Alfvén wave problem. Solution L^2 errors on the x component of the magnetic field are shown for the LDF basis with Powell terms (left), and Legendre basis with hyperbolic cleaning (right). Errors are computed after the wave has crossed five wavelengths, at time $t = 5$. At the lowest resolutions, the DG-2 and DG-3 schemes do not accurately capture the wave fronts, which explains their apparent superconvergence. At error levels $\sim 10^{-8}$, the solution is polluted by the parametric instability of Alfvén waves discussed in the text. The effective order of convergence is degraded for the DG-4 scheme with LDF bases, probably because of projection effects. We refer to the text for a discussion of these last two effects.

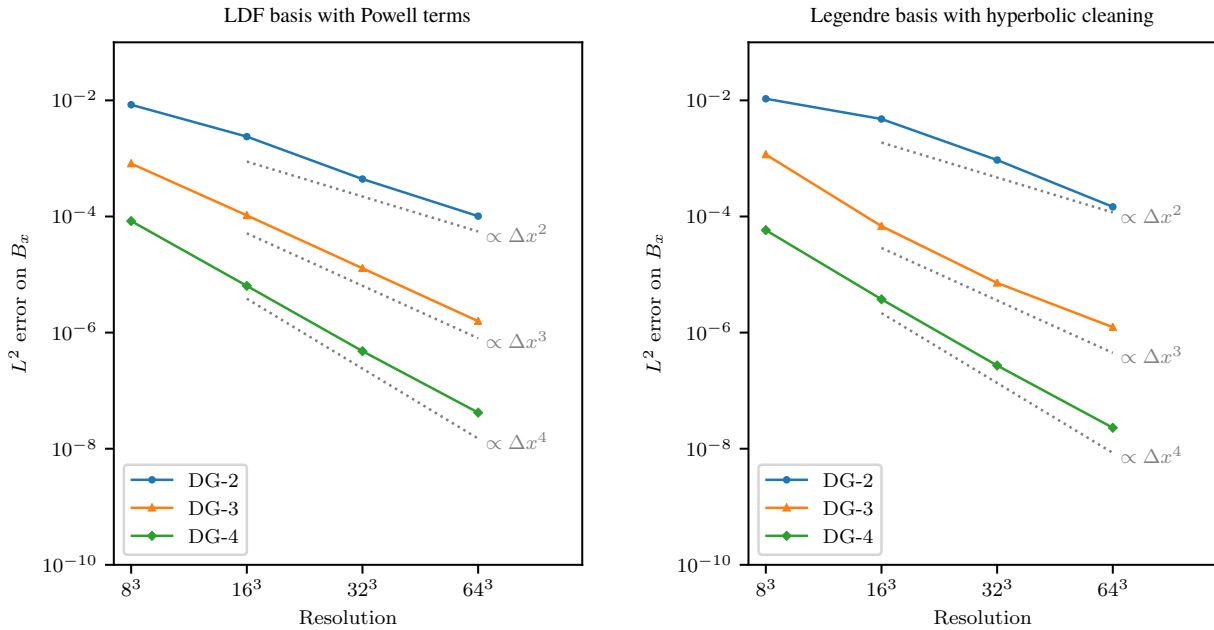


Figure 7. Convergence of errors in the 3D Alfvén wave problem. Solution L^2 errors on the x component of the magnetic field are shown for the LDF basis with Powell terms (left), and Legendre basis with hyperbolic cleaning (right). Errors are computed after 3 wave fronts have crossed the box domain, at time $t_f = \sqrt{3}/2$.

different orders, we use the RK3 SSP time integrator for all runs of this test problem. The zoom-in panel on the right shows that going from second to third order greatly improves the capturing of the strong oscillations in the interaction region, in particular at the lower resolution level 7. This test problem demonstrates the ability of higher-order methods to capture finer features in the flow with

lower numerical diffusion. In particular, our third-order scheme is able to resolve the major oscillating features of the interaction region with only 128 cells, which is very competitive for this test problem (Derigs et al. 2016; Chakravarthy et al. 2015).

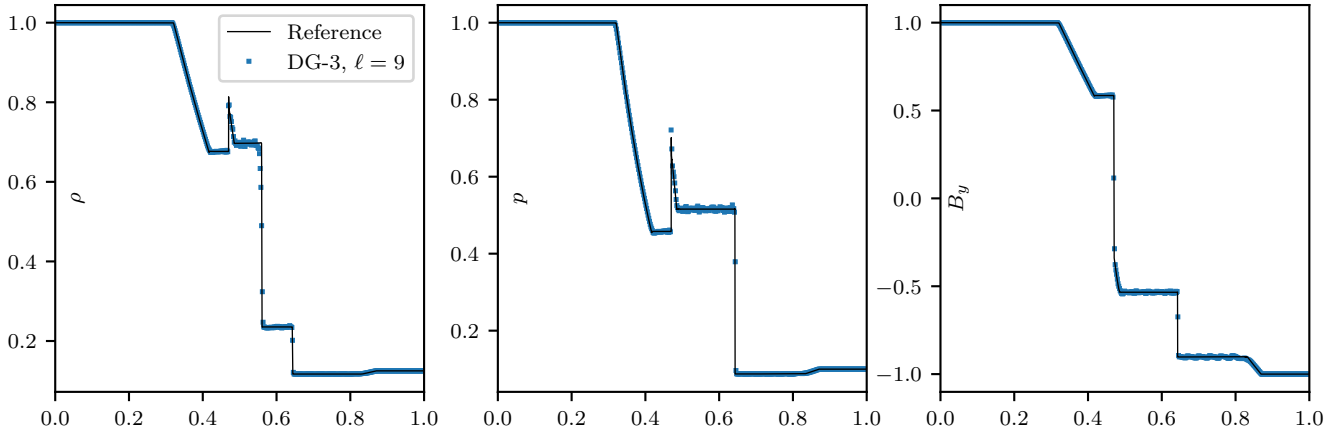


Figure 8. *Brio-Wu MHD shock tube problem.* The density, pressure and y component of the magnetic field are shown at $t = 0.1$ for a grid with 512 cells (resolution level $\ell = 9$). The reference solution was computed using Athena with the third-order Roe solver on 10^4 mesh points. All the MHD waves are correctly captured and shocks are captured within 1–2 cells without overshooting. Some oscillations are visible, probably due to the simple limiting procedure employed (see text).

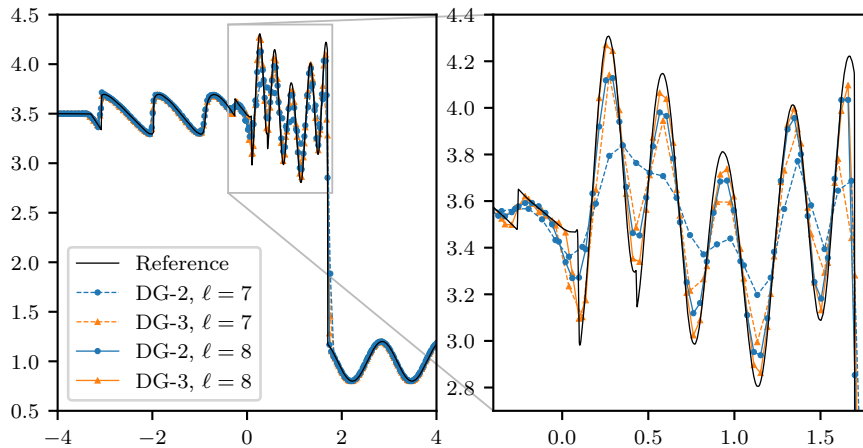


Figure 9. *MHD Shu-Osher shock tube test problem.* The density profile is shown at final time $t_f = 0.7$. This figure can be directly compared to Derigs et al. (2016). The reference solution was computed using Athena with the third-order Roe solver on 10^4 mesh points. The right panel is a zoom on the region shown framed in the left panel. The numerical solution for second-order (DG-2) and third-order (DG-3) schemes are shown, for resolution levels $\ell = 7$ (128 grid points) and $\ell = 8$ (256 points).

5.3.3 Orszag–Tang vortex problem

We now consider the Orszag–Tang vortex problem, a widely-used test problem for MHD. The vortex starts from a smooth initial field configuration, and quickly forms shocks before transitioning to 2D MHD turbulence.

For this problem, our computational domain is $[0, 1]^2$ and we use $\gamma = 5/3$. The initial density and pressure are uniform, with $\rho = \frac{25}{36\pi}$ and $p = \frac{5}{12\pi}$. The initial gas velocity field is $\mathbf{v} = (-\sin(2\pi y), \sin(2\pi x), 0)$, and the initial magnetic field is $\mathbf{B} = (-B_0 \sin(2\pi y), B_0 \sin(4\pi x), 0)$, with $B_0 = 1/\sqrt{4\pi}$.

The solution at $t = 0.5$ is presented in Fig. 10. This is the well-recognizable picture of the Orszag–Tang vortex, presented by many authors discussing MHD codes. Note that we obtain both sharp shocks and smooth, noise-free flow with resolved features between the shocks. During our experiments we found that details of both the limiting procedure and the Powell term discretization could strongly impact the shape of the low-density smooth regions.

While the Orszag–Tang vortex is a classic MHD test, many

authors do not discuss how their codes capture the evolution beyond $t = 0.5$. During our experiments, we found that while many limiters and Powell term prescriptions work equivalently well until $t = 0.5$, long time integration of this problem can be challenging, in particular for divergence errors, as noted by Balsara (1998). We found that running the problem for longer times could discriminate well between otherwise seemingly equivalent discretizations. In particular, at times $t \in [0.75, 0.85]$, complex shock interactions occur near the center of the box which can easily cause divergence blowups.

Because the Orszag–Tang test gives rise to many complex flow features including regions of potentially severe numerical divergence, we also use this test problem to assess the impact of non-conservative Powell terms on scheme conservation. We find that in general the total energy is the conserved quantity most directly impacted by the Powell term. Fig. 11 shows the time evolution of the magnetic, kinetic, thermal and total energies (E_{mag} , E_{kin} , E_{th} , E_{tot} respectively) for the DG-3 scheme over $0 \leq t \leq 1$. In the bottom

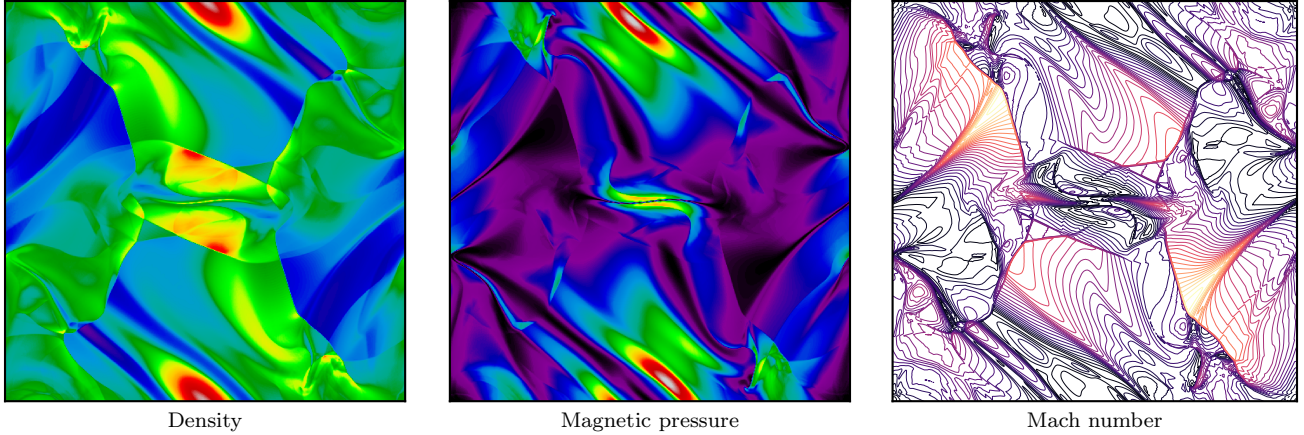


Figure 10. Orszag–Tang vortex test problem. The density, pressure and Mach number are shown at $t = 0.5$ on a 512^2 grid, computed using the third-order DG scheme with the Powell method.

panel, the energies are shown for resolution levels $\ell = 7$ (128^2) and $\ell = 9$ (512^2) and for both divergence control schemes. The energy histories are in good overall agreement across methods and resolutions, although differences develop over time. The top panel shows the slight deviation of the total energy E_{tot} from its initial value due to non-conservative source terms with the Powell scheme. Over the whole time evolution, E_{tot} has deviated by at most 0.6% for $\ell = 7$, and 0.3% for $\ell = 9$. Increasing the resolution seems to contribute to reduce the deviation of E_{tot} , even though it is known that divergence errors in general do not locally converge away with resolution (Tóth 2000). Note that the deviation is not evolving monotonically, as such, it is difficult to appreciate the long-term impact of non-conservative source terms on the solution. Note also that the effect of the divergence control scheme on the magnetic energy (lower inset) is comparable in magnitude to the effect of resolution, so the detailed differences in energies may not all be attributed to the difference in divergence treatment.

We point out that total conservation is not a sufficient test of the quality of the solution against divergence errors, because energy can be created and destroyed *locally* in the flow with different signs, depending on the sign of various quantities in the source term (26). While the conservation results are encouraging overall, we intend to perform more careful comparisons when applying both schemes to physical problems. Driven turbulence problems in particular may turn out to be more sensitive to non-conservation issues, because energy is continually injected in the simulation domain.

Finally, we conclude this section with Fig. 12, which shows the same Orszag–Tang problem run up to $t = 0.5$ with adaptive mesh refinement using the DG-3 Powell scheme, with resolution varying from 64^2 to 1024^2 from the coarsest to finest cells. The refinement is based on the density gradient, and follows the implementation discussed in more detail in Schaal et al. (2015).

5.3.4 Advected Orszag–Tang vortex

In this problem, we test the code’s Galilean invariance by running the Orszag–Tang vortex test case up to $t_f = 0.5$, giving the fluid an additional uniform global bulk velocity $v_0 = (10, 10, 0)$. This corresponds to a hypersonic advection with initial Mach number ≈ 14 . At time t_f , the flow will have crossed the box 5 times in the

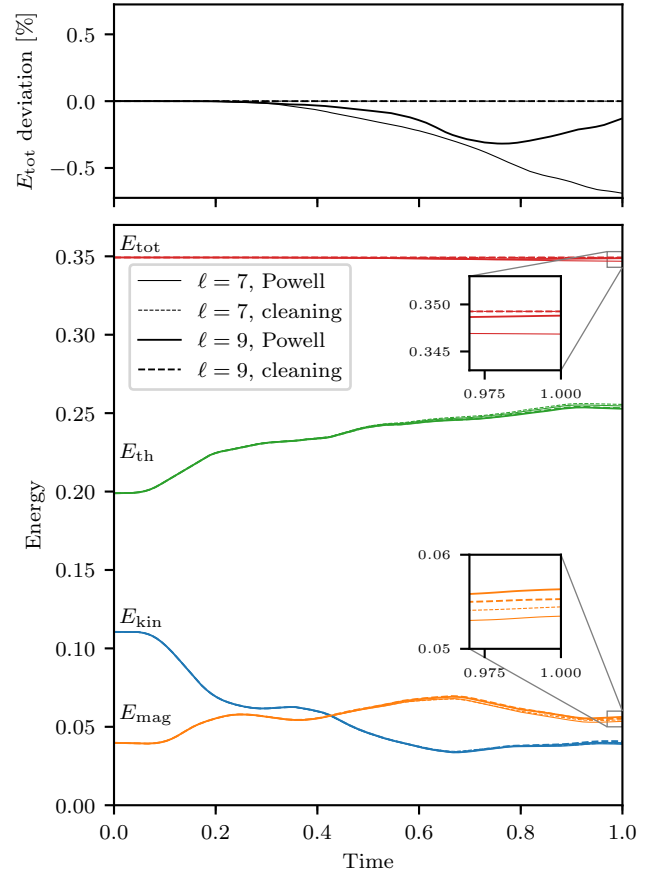


Figure 11. Evolution of energies in the Orszag–Tang vortex problem, for the 3rd order DG-3 scheme, at resolution levels 7 (128^2) and 9 (512^2) using the Powell and hyperbolic cleaning schemes. The bottom panel shows the time evolution of the different energies integrated over the whole domain, while the top panel shows the deviation in the total energy E_{tot} from its initial value due to non-conservative terms for the Powell scheme.

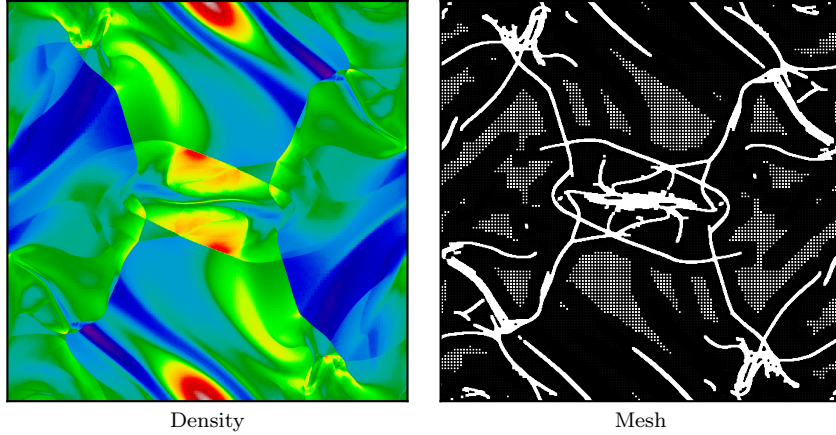


Figure 12. Orszag–Tang vortex problem with adaptive mesh refinement at $t = 0.5$, from resolution level 7 (128^2) to 10 (1024^2) using the DG-3 scheme with hyperbolic divergence cleaning. The left panel shows the density field, while the right panel shows the cells of the AMR grid up to level 9 (level 10 cells are not drawn for clarity). The refinement criterion is set by the local density gradient, determined from the DG weights in each cell.

x and y directions, and end up in a configuration identical to the non-advected problem at the same time.

We present the results for advected and non-advected problems for second-order DG-2 and third-order DG-3 in Fig. 13. For this test, all runs use the same RK3 SSP time integrator. Comparing the density contours of the non-advected and advected DG-2 solutions at resolution 128^2 (first row), we notice that the supersonic bulk advection velocity increases numerical diffusion significantly. In the advected solution, the shocks are smoothed out, the density valleys are shallower, and some of the complex structures, e.g. in the lower-right corner of the domain, are eroded. This diffusion is clearly visible on the difference map on the right column, which shows the deviation between the advected and the non-advected maps. For each run configuration, the total number of degrees of freedom N_{DoF} is indicated at the left of the corresponding row.

The middle row of Fig. 13 shows the same second-order scheme, this time with a spatial resolution of 256^2 . This increased resolution results in four times as many degrees of freedom, and is able to significantly improve the advection errors in smooth regions of the flow, although it does not eliminate them.

In the bottom row, the resolution was taken back to 128^2 , but the order was increased to DG-3. Compared to DG-2 at the same resolution, this represents only a $1.95\times$ increase in the number of degrees of freedom. With the 3rd order algorithm, advection errors in smooth regions of the flow are almost completely eliminated, but errors are also reduced close to shocks, resulting in the best overall preservation of Galilean invariance across the three runs.

This test shows that the reduction of advection errors offered by higher orders, which was discussed by e.g. Robertson et al. (2010) and demonstrated with DG by Schaal et al. (2015) for Euler hydrodynamics problems, carries over to complex MHD flows in presence of shocks, even with our relatively simple slope limiting prescription. It also shows that in some cases, increasing the order can result in notable improvements, and at a lesser expense in terms of degrees of freedom than increasing the resolution (in this case, DG-3 128^2 requires only about half as many degrees of freedom as DG-2 256^2).

5.3.5 MHD rotor problem

We now look at the so-called 2D MHD rotor problem introduced by Balsara & Spicer (1999b), for which we use the more stringent “first rotor problem” variant of Tóth (2000). In this setup, a dense disk of fluid rotates within a static fluid background, with a gradual velocity tapering layer between the disk edge and the ambient fluid. An initially uniform magnetic field is present, which winds up with the disk rotation and contains the dense rotating region through magnetic field tension. The computational domain is $[0, 1]^2$. Initial pressure and magnetic fields are uniform in the whole domain, with $p = 1$ and $\mathbf{B} = (5/\sqrt{4\pi}, 0, 0)$. The central rotating disk is defined by $r < r_0$ where $r^2 = (x - 0.5)^2 + (y - 0.5)^2$, and $r_0 = 0.1$. Inside the disk, $\rho = 10$, and the disk rotates rigidly with $(v_x, v_y) = (0.5 - y, x - 0.5)v_0/r_0$ with $v_0 = 2$. Beyond $r > r_1$ lies the background fluid, which has density $\rho = 1$ and is at rest: $v_x = v_y = 0$. In the annulus $r_0 \leq r \leq r_1 = 0.115$, the taper region linearly interpolates between the disk and the background, with $(v_x, v_y) = (0.5 - y, x - 0.5)v_0 f/r_0$ and $\rho = 1 + 9f$, where $f = (r_1 - r)/(r_1 - r_0)$ is the tapering function. The simulation runs until $t_f = 0.15$. We use periodic boundary conditions, but note that the perturbations will not have reached the domain boundaries at t_f , so reflective or inflow boundary conditions are also suitable.

The density, magnetic pressure and Mach number contours are presented in Fig. 14 for the Powell scheme. We note that the central contours show a very good conservation of the circular rotation pattern, which was found by Tóth (2000) to be challenging for some MHD schemes. Also, there are no distortions at the outskirts of the central almond-shaped disk region, which suggests that the magnetic field divergence is well controlled (see Li & Shu 2005). Fig. 15 presents slices² of the magnetic field at t_f across horizontal and vertical cuts through the center of the box, and can be compared to fig. 26 of Stone et al. (2008) with which they show good agreement.

² Note that since we use an even number of grid points per dimension, we here simply approximate the slices as the average of the two rows (or columns) of cells immediately adjacent to the box center line.

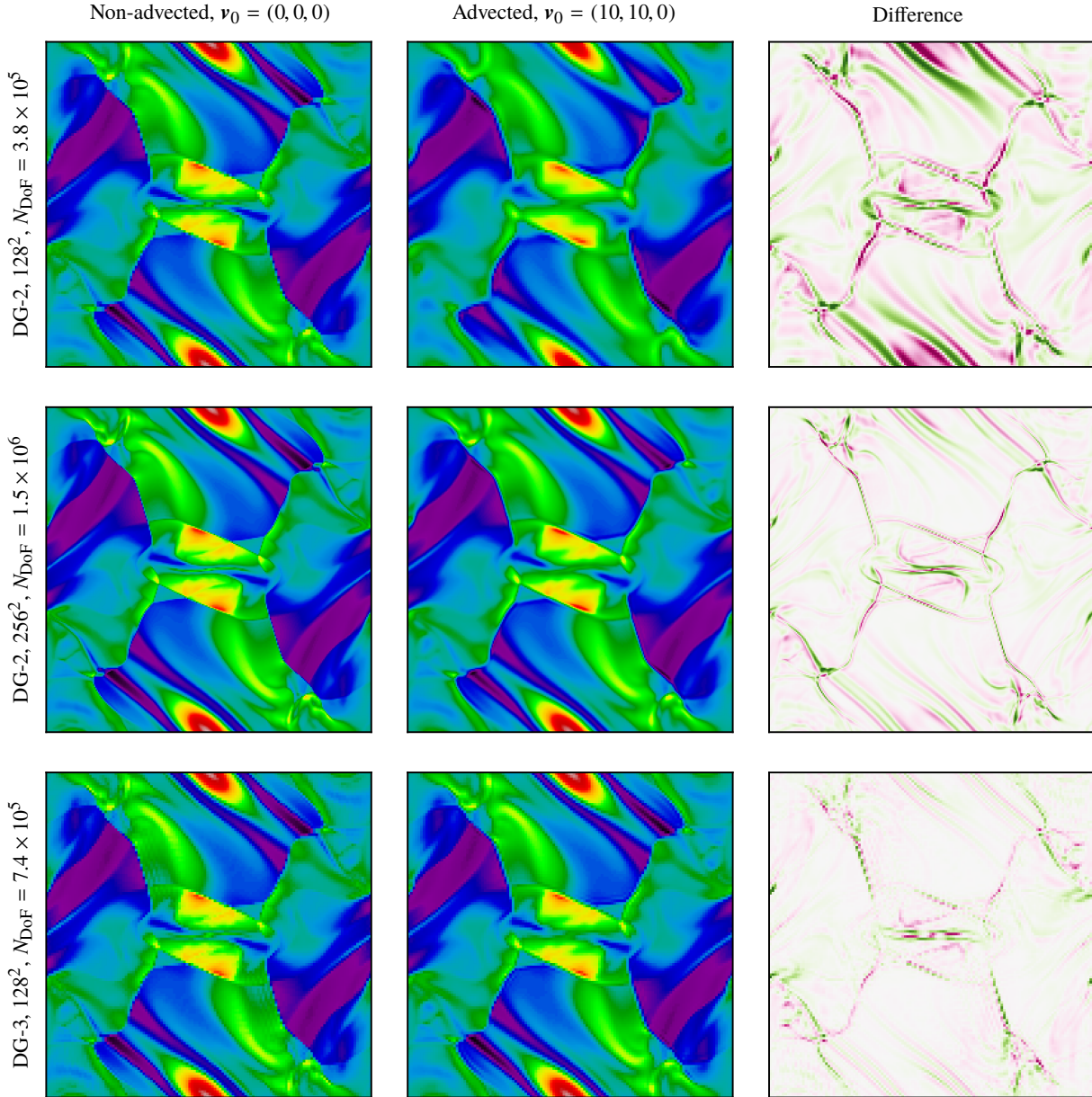


Figure 13. *Advected Orszag–Tang vortex problem.* We show the density cell averages at $t = 0.5$ for the non-advected Orszag–Tang vortex problem (left column), the advected problem (center column), and the difference between the two (right column), for various combinations of scheme order and resolutions (rows). The top row is run using second-order DG at 128^2 , the middle row increases the resolution of the second-order method to 256^2 , while the bottom row uses the third-order scheme at the lower 128^2 resolution. The color scales are identical across all density maps and all difference maps. In the row labels, N_{DoF} describes the total number of scalar degrees of freedom in the problem. This test shows the ability of higher order methods to reduce advection errors and preserve finer structures in the flow, even with limiting in the presence of MHD shocks, at a smaller cost in terms of degrees of freedom.

5.3.6 MHD blast wave in 2D

This test follows the evolution of MHD discontinuities in a strongly magnetized configuration. For this test, we adapt the version of Stone et al. (2008) to a square periodic domain $[0, 1]^2$. The background fluid has zero velocity, $\rho = 1, p = 0.1$, and magnetic field $\mathbf{B}_0 = (B_0 \cos \alpha, B_0 \sin \alpha, 0)$ with $B_0 = 1, \alpha = 45^\circ$. Within a radius $r_0 = 0.1$ of the center of the box, we set $p = 10$, keeping all other primitive variables at their background values. We use $\gamma = 5/3$, and choose $\alpha = 45$ degrees. The simulation is run until a final time of $t_f = 0.2$.

Note that when running this test with \mathbf{B}_0 aligned with the

grid with our scheme, small localized finger-like features develop at the shock front around the axis of symmetry parallel to \mathbf{B}_0 , which we attribute to the so-called “carbuncle” instability (Robinet et al. 2000). This instability manifests itself in the case of strong shocks exactly aligned with the grid directions, when using non-diffusive Riemann solvers. In few highly specific simulation configurations where this problem may appear, it may be resolved by a modification of the numerical fluxes (see e.g. Robinet et al. 2000; Stone et al. 2008); however here we do not modify the scheme to counter this peculiar instability.

We also successfully run this test with $B_0 = 10$, with an initial

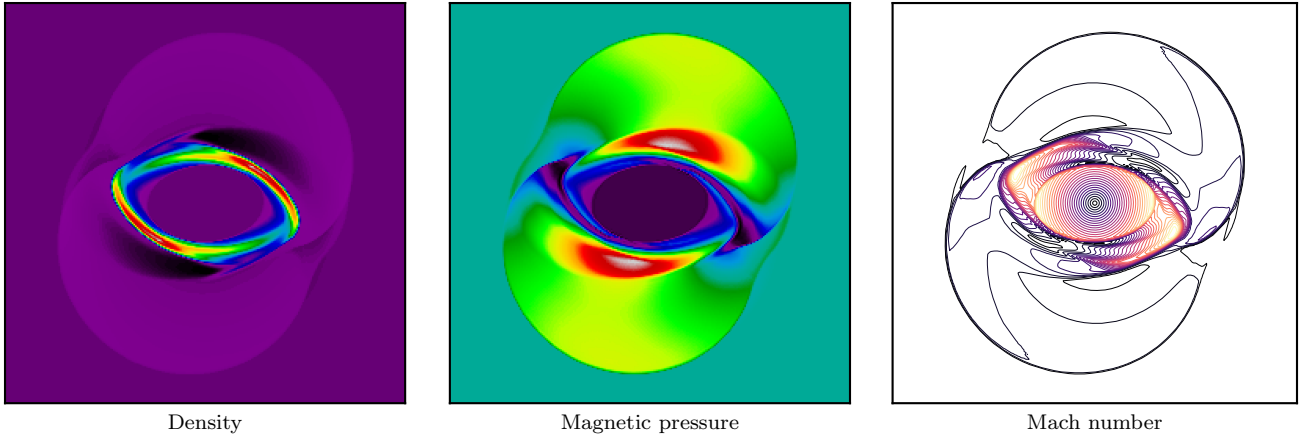


Figure 14. *Magnetic rotor test problem.* The density, pressure and Mach number contours in the 2D magnetic adiabatic rotor test are shown, on a 256^2 grid using the third-order Powell scheme.

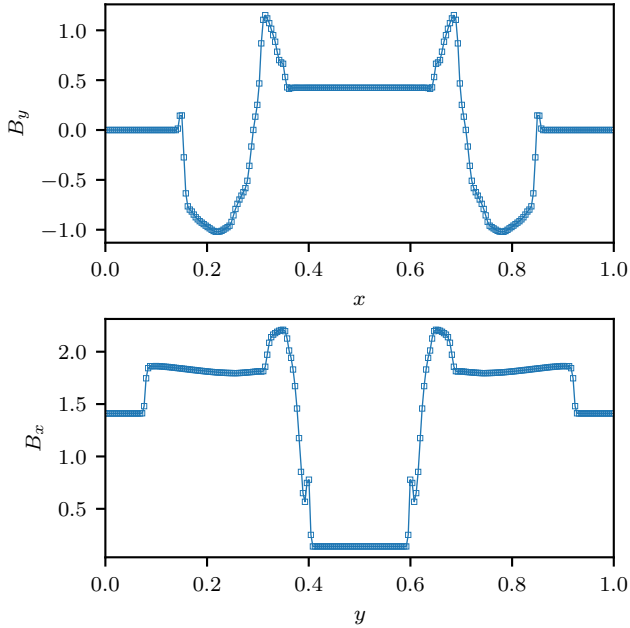


Figure 15. *Slices of the magnetic field in the MHD rotor test,* along $y = 0.5$ (top) and $x = 0.5$ (bottom), computed using DG-3 with Powell terms. The slices can be compared to Stone et al. (2008).

plasma β of 2×10^{-2} , which demonstrates the robustness of our implementation in highly magnetized shock configurations. In all cases, we find that the symmetry of the problem is very well preserved, as can be seen on the Mach number contours in Fig. 16. The shocks are sharp, while post-shock regions remain smooth.

To investigate the potential impact of the Powell method on magnetic field divergence on this test problem, we also plot slices of the magnetic field in Fig. 17 following Hopkins & Raives (2016). Unlike these authors, we find no evidence of errors in the components of the magnetic field with the Powell scheme for this test problem in our scheme.

5.3.7 MHD blast wave in 3D

To test the shock-capturing and positivity preserving performance of the code in 3D, we adopt the 3D blast wave setup of Balsara et al. (2009). The computation domain is $[0, 1]^3$ with periodic boundary conditions. The background fluid is initially at rest with respect to the grid, with $\mathbf{v} = 0$, $\rho = 1$, and a uniform magnetic field $\mathbf{B} = (B_0, B_0, B_0)$. To match the setup for the slices presented in Balsara et al. (2009), we take $B_0 = 100\sqrt{3}^{-1}\sqrt{4\pi}^{-1}$ in our system of units³. The pressure is set to $p = 0.1$ in the background, and within a central ball of radius $r_0 = 0.1$ we set $p = 1000$ to initialize the blast. This creates a near-infinite shock strength with a pressure ratio of 10^4 , in a strongly magnetized background with a plasma- β of 7.5×10^{-4} . We take $\gamma = 1.4$, and run the simulation until final time $t_f = 0.01$.

Fig. 18 shows slices at constant z through the center of the blast at t_f , for the third-order DG-3 method at resolution 128^3 . Note that for slicing purposes, care must be taken to place the blast center exactly at the center of a cell, which we achieve at 128^3 resolution by shifting the center of the high pressure ball by half a cell in every direction. The slices of Fig. 18 may be compared to fig. 11 in Balsara et al. (2009), which uses a higher resolution of 151^3 .

The DG Powell scheme is able to maintain positivity of the pressure and density in the whole domain, while finely capturing the very strong discontinuities and resolving the complex structures in the velocity. Note that no oscillations are visible around discontinuities. Some numerical noise is present in the post-shock regions for the density and pressure fields, but disappears with a more aggressive limiting threshold. This test shows the robustness and shock-capturing behavior of the Powell scheme for three-dimensional problems involving very strong magnetized shocks.

This problem proves particularly challenging for our implementation of hyperbolic cleaning. In such a low plasma- β setup, changes in the magnetic field caused by hyperbolic cleaning propagate fast in the ambient medium at the cleaning speed c_h , and result in fluctuations of the magnetic pressure which locally cause negative thermal pressures far away and ahead of the MHD blast wave

³ Note that Balsara et al. (2009) mention $1000\sqrt{3}^{-1}$ for B_0 in the text, but their figure seems to correspond to $100\sqrt{3}^{-1}$ instead.

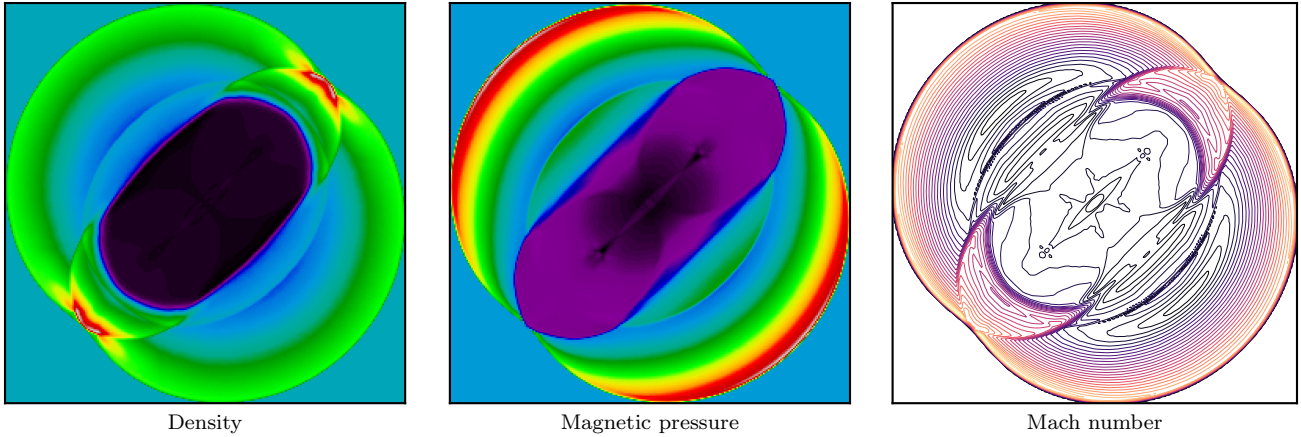


Figure 16. Two-dimensional MHD blast test problem. The density, magnetic pressure and Mach number contours are shown on a 256^2 grid using the third-order Powell scheme.

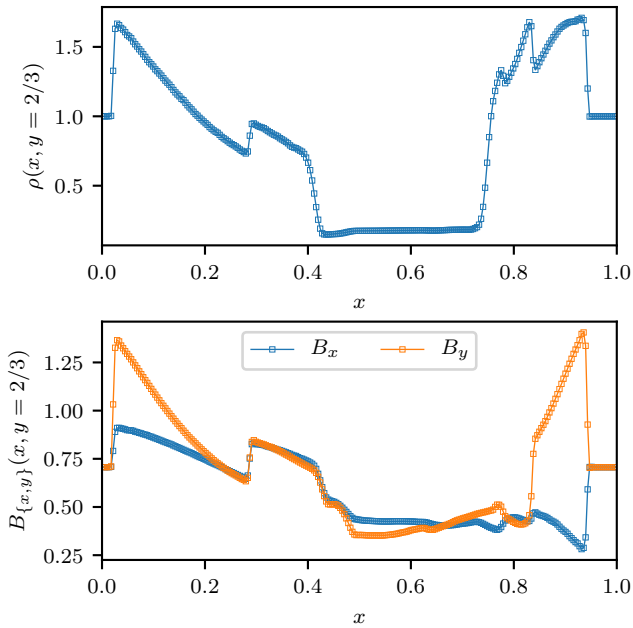


Figure 17. Slices across $y = 2/3$ in the MHD blast wave test problem, following Hopkins & Raives (2016), shown here for the DG-3 Powell scheme. We do not observe any of the dramatic errors noted by these authors on their tests with Powell terms.

front. These negative pressures are successfully but aggressively corrected by our positivity limiter, and although the computation does not crash, the resulting fields show damage. Similar issues have been noted and worked on by a number of authors (e.g. Mignone & Tzeferacos 2010; Tricco & Price 2012; Susanto 2014; Tricco et al. 2016); we come back to this issue and possible solutions in the discussion.

5.4 Divergence control problems

We now turn to test problems more specifically aimed at evaluating the efficiency of the divergence control, with a focus on the Powell scheme. We have already discussed some aspects related to $\nabla \cdot \mathbf{B}$ in

some of the previous test problems; we now show the stability of the Powell scheme and some consequences of its non-conservative source terms.

On a general note, like Balsara & Spicer (1999b) we found that problems with strong moving shocks, such as the blast or rotor problems, are not necessarily the most stringent tests of divergence control. Instead, colliding shocks whose convergence front is at rest with respect to the grid—such as the rotated shock tube of Tóth (2000) described in 5.4.3, or some shock-shock interactions in the Orszag–Tang vortex of Section 5.3.3—proved to be much more challenging tests of the divergence control scheme and its stability in particular.

In addition to shock interactions, we found that smooth problems such as the simple advection of a magnetic field loop can also be unstable with inappropriate discretizations of the Powell term, and their smooth character makes it easier to follow the development of the divergence instability. Without Powell source terms, the instability usually grows faster with scheme order, as the numerical diffusion at lower orders helps slow down the divergence runaway.

5.4.1 Loop advection

This test follows the advection of a magnetic field loop after Gardiner & Stone (2005). On the periodic domain $[-1, 1]^2$, the background fluid has $\rho = 1, p = 1$, and a global advection velocity $(v_x, v_y) = (2, 1)$ so that the ambient flow is not aligned with grid directions. Letting r be the radial distance to the center of the box, the magnetic field is initialized from a vector potential $\mathbf{A} = (0, 0, A_z(r))$ with $\mathbf{B} = \nabla \times \mathbf{A}$. To define a magnetic field loop of radius $r_0 = 0.3$, we set $A_z(r) = \max(0, A_0(r_0 - r))$. Taking $A_0 = 10^{-3}$, we obtain a very weakly magnetized configuration with a plasma β of order 10^6 , in which the magnetic field is essentially a passive scalar. For this field configuration, the MHD current vanishes everywhere, except at $r = 0$, and $r = r_0$ where the corresponding current line and return current tube are singular.

The aim of the test is to verify that the current loop is advected without deformation or noise, and to monitor the time evolution and dissipation rate of the total magnetic energy, following Gardiner & Stone (2005); Stone et al. (2008).

As discussed in Gardiner & Stone (2005), the linearized dynamics of the magnetic field involves the diagonal derivatives

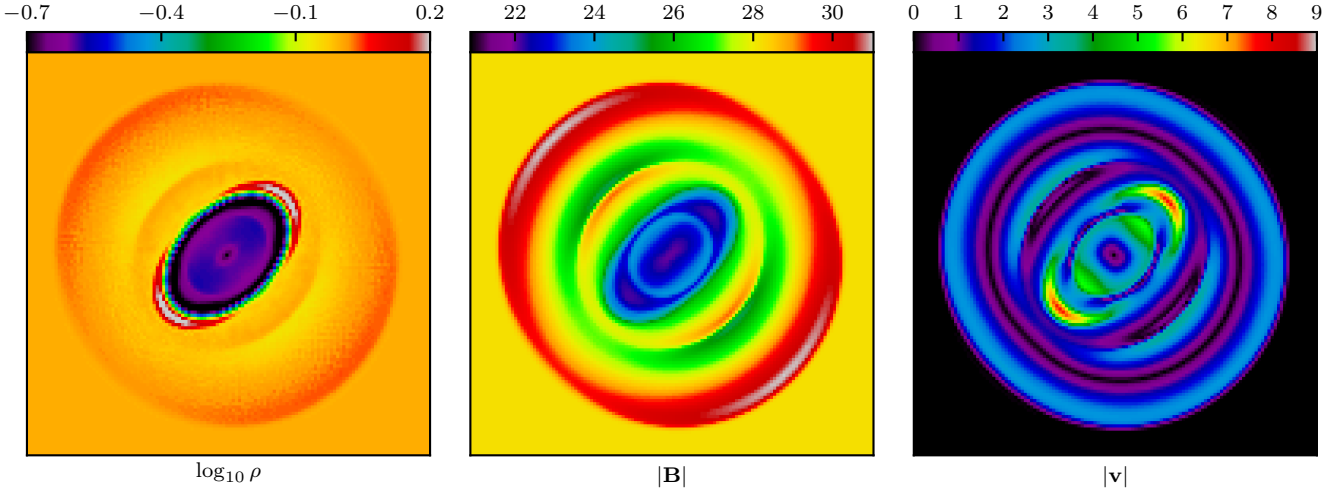


Figure 18. MHD blast wave in 3D. Slices in the (x, y) plane showing log density, magnitude of velocity, and magnitude of magnetic field at resolution 128^3 with the third-order DG method.

$\partial B_x / \partial x$, and this test is therefore sensitive to the $\nabla \cdot \mathbf{B}$ treatment, although this is mostly an issue for directionally split methods. In any case however, the linearized loop advection setup constitutes a so-called resonant hyperbolic problem, as discussed in Kemm (2013), and therefore constitutes a good test for the growth of divergence instabilities. We found this test to be particularly unstable without divergence control, even with a smooth non-singular loop configuration.

Fig. 19 shows the z component of the cell average of the current density $\mathbf{j} = \nabla \times \mathbf{B}$ at final time $t = 2$ after two horizontal domain crossings, at resolution 128^2 , for orders 2, 3 and 4. For this comparison, we use the RK3 SSP time integrator across all spatial orders. The current density is a stringent diagnostic since, being a derivative of the magnetic field, it is very sensitive to noise and local fluctuations. The scheme preserves the exact circular shape of the current loop at all orders, with very little noise and oscillations in the current at all orders. In addition, the reduction in numerical diffusion and advection errors is clearly noticeable as order is increased. We also see that the current loop gets increasingly resolved at higher orders, showing an increase in effective resolution even for singular features.

The plot of Fig. 20 follows the time evolution of the normalized total magnetic energy during two horizontal domain crossings, at resolution 128^2 . The results are shown for the Powell and hyperbolic cleaning schemes. The hyperbolic cleaning scheme seems slightly more diffusive, especially at lower orders. The corresponding decay time-scales may be fitted with the power law functional form proposed in Gardiner & Stone (2005): $E_{\text{mag}}(t) = A(1 - (t/\tau)^\alpha)$. For the Powell scheme, we measure $\tau \approx 1.8 \times 10^4$ for DG-2, $\tau \approx 6.1 \times 10^7$ for DG-3, and $\tau \approx 6.0 \times 10^{11}$ for DG-4, which quantifies the strong reduction of the decay rate τ^{-1} with increasing scheme order.

5.4.2 Current sheet problem

A very stringent test of the capability of the code to handle numerical divergence and maintain positivity is the so-called current sheet problem (Hawley & Stone 1995; Gardiner & Stone 2005), which follows the evolution of sharp interfaces between regions with magnetic fields of opposing directions. The chosen fluid velocities will

push and pull the interfaces along their normal direction, letting the magnetic tension act as a restoring force which results in Alfvén oscillations. At the interfaces—where the tangential component of the magnetic field flips sign—numerical reconnection will occur, forming growing isolated islands of reconnected magnetic regions. Similar configurations have also been used in the context of non-ideal MHD to measure reconnection rates (see e.g. Marinacci et al. 2018, and references therein).

In our implementation of this test, the computational domain is $[0, 1]^2$ with periodic boundary conditions. The density and pressure are uniform with $\rho = 1$, $p = \frac{1}{2} B_0^2 \beta$. We pick $B_0 = 1/\sqrt{4\pi}$ for our units, and we set the plasma- β to $\beta = 0.1$, so that the configuration is strongly magnetized. The domain is split into three regions forming vertical bands, separated by two vertical interfaces located at $x = 0.25$ and $x = 0.75$. In the left ($x < 0.25$) and right ($x > 0.75$) bands, we initially set $\mathbf{B} = (0, B_0, 0)$. In between the left and right interfaces, we initialize $\mathbf{B} = (0, -B_0, 0)$, so that the vertical (tangential) component of the magnetic field flips across each interface. The initial velocity field is chosen as $\mathbf{v} = (A \sin(2\pi y), 0, 0)$ with $A = 0.1$. The adiabatic constant is set to $\gamma = 5/3$, and the evolution of the interfaces is followed until $t_f = 10$.

Note that this setup corresponds to the results presented in the Athena code tests web page⁴, and is more stringent than the version of Gardiner & Stone (2005): the plasma- β is twice as low, and the oscillations are chosen with a larger amplitude.

The code is able to stably follow the evolution of the current sheet up to the final time $t = 10$ with both the Powell and hyperbolic cleaning schemes. Fig. 21 shows the evolution of the kinetic, thermal, magnetic and total energies in the current sheet problem, computed with both divergence control methods using the third-order DG-3 scheme, at resolutions levels 7 and 8.

The evolution of the kinetic and magnetic energies are similar for both divergence control schemes: while they exhibit some differences, these are comparable in magnitude to the effect of resolution,

⁴ <https://www.astro.princeton.edu/~jstone/Athena/tests/current-sheet/current-sheet.html>.

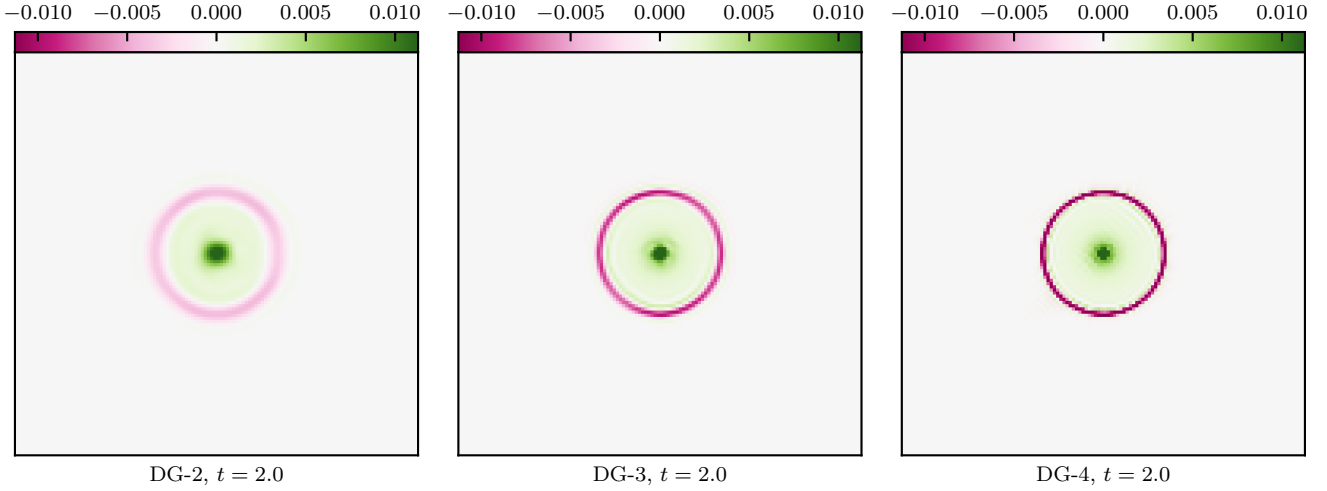


Figure 19. MHD current in the loop advection problem. The average z component of current density $\mathbf{j} = \nabla \times \mathbf{B}$ in each cell is shown for the Powell DG-2, DG-3 and DG-4 schemes at resolution 128^2 , at $t = 2$ after two horizontal domain crossings. The code preserves the shape of the current loop very well at all orders, and the test demonstrates the qualities or higher order schemes for reducing numerical diffusion.

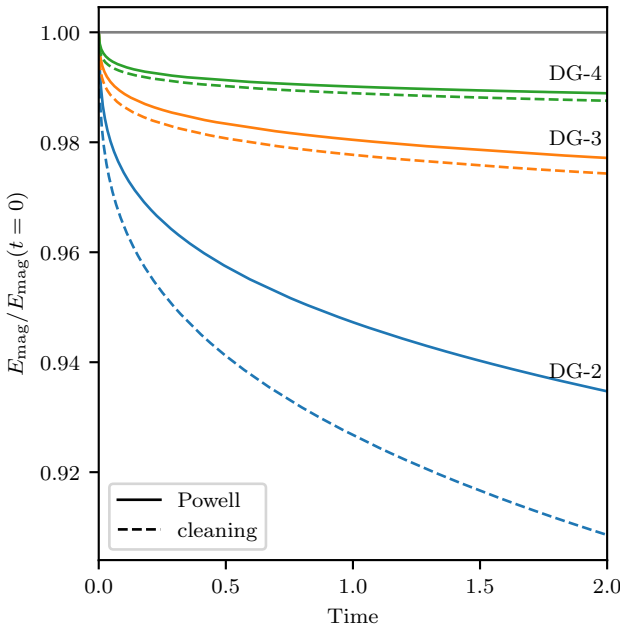


Figure 20. Time evolution of the magnetic energy in the field loop advection test, for the second, third, and fourth order schemes, at resolution 128^2 . The solid lines correspond to the Powell scheme, whereas the dashed lines use hyperbolic divergence cleaning.

and it is therefore difficult to attribute them to the divergence control scheme.

However, the Powell scheme introduces noticeable non-conservation of the total energy (red curve), of the order of 2–3%, which does not seem to disappear with increasing resolution. This, in turn, translates into a deviation of the thermal energy, whose relative magnitude can be severe for such strongly magnetized configurations. In this case, the Powell scheme results in artificial cooling of the plasma. This test is very challenging for the Powell method:

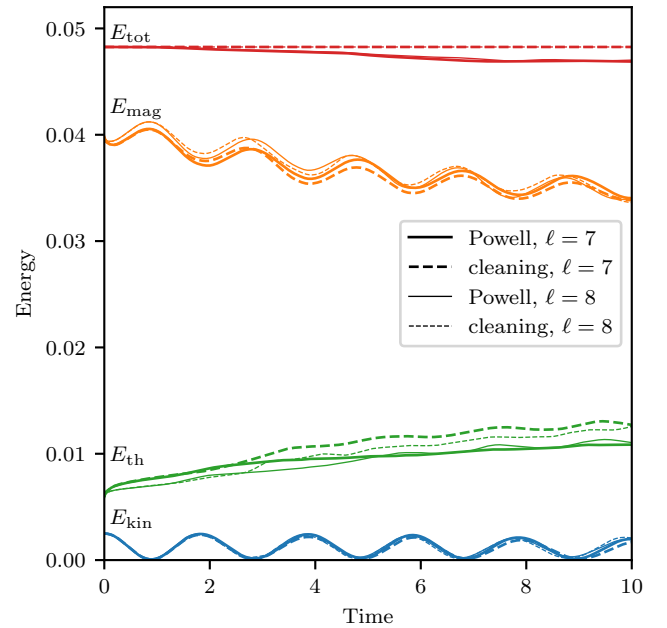


Figure 21. Current sheet test. The evolution of energies is shown for third-order DG scheme, computed with the Powell method (solid lines) and hyperbolic divergence cleaning (dashed lines), at resolution levels 7 (128^2 grid, thick lines) and 8 (256^2 grid, thin lines). The transfer of magnetic energy E_{mag} into thermal energy E_{th} is caused by numerical reconnection.

while initially the velocity and magnetic fields are orthogonal everywhere, as soon as reconnection islands start to form, regions of collinear \mathbf{v} and \mathbf{B} will quickly develop. In such regions, the energy injection component $\mathbf{v} \cdot \mathbf{B}$ of the Powell source term (25) will reach maximal magnitude, and will be very sensitive to any local $\nabla \cdot \mathbf{B}$ introduced in the numerical solution.

Finally, we note that the slope limiter has a marked overall impact on this problem. First, the amount of energy injected by the

Powell scheme seems to strongly depend on the choice of limiter threshold: making the limiter more aggressive by decreasing $\tilde{M} = 5$ down to $\tilde{M} = 1$ reduces the deviation by more than one order of magnitude. We advise that this complicates the comparison of the energy deviations at the two resolution levels, as our choice of scaling of M or \tilde{M} with resolution may impact the results. In addition, we see very little difference between the second and third-order DG-2 and DG-3 schemes for this problem. This could be explained by the slope limiter triggering locally on the sharp discontinuities at the interfaces, thereby degrading the scheme to at most second order precisely in places where numerical reconnection occurs. A more sophisticated limiter could therefore improve both energy conservation with the Powell scheme, and provide clearer benefits of higher-order methods for this particular test configuration.

5.4.3 Rotated shock tube

This problem, adapted from Tóth (2000), is designed specifically to test the impact of Powell terms on jump conditions in the situation of a strong shock at an angle with the grid. On the computational domain $[-1, 1]^2$ with periodic boundary conditions, we define a rotated frame at an angle $\theta = \arctan(2)$ with the lab frame (x, y) , and whose origin lies at the box center $(0, 0)$. We label coordinates and vector components in the rotated frame with (\parallel, \perp) . In this coordinate system, we define an interface at $x_{\parallel} = 0$, where we initialize a shock. For $x_{\parallel} < 0$, we set the velocity $(v_{\parallel}, v_{\perp}, v_z) = (10, 0, 0)$ and pressure $p = 20$, whereas for $x_{\parallel} \geq 0$ we initialize $(v_{\parallel}, v_{\perp}, v_z) = (-10, 0, 0)$ and $p = 1$. This sets up two fast converging flows colliding at $x_{\parallel} = 0$. The density is set to $\rho = 1$ everywhere, and the initial magnetic field is uniform with $(B_{\parallel}, B_{\perp}, B_z) = (5, 5, 0)/\sqrt{4\pi}$. We take $\gamma = 5/3$, and run the simulation until final time $t_f = 0.08/\sqrt{5}$.

Note that in comparison to Tóth (2000), we run the problem on a square domain with periodic boundary conditions for simplicity. This will create shocks at domain boundaries, so we use a bigger domain of size 2 to avoid contamination of the central region. Correspondingly, we run at a higher spatial resolution of 512^2 , focusing only on the central 256^2 region of the domain which by time t_f is not yet impacted by the boundary shocks. As such, our results are directly comparable to Tóth (2000).

In the configuration of this test problem, the $\nabla \cdot \mathbf{B} = 0$ constraint imposes that the component B_{\parallel} of the magnetic field, which is parallel to the shock propagation direction and normal to the interface, should remain exactly at its initial constant value. Achieving this is particularly challenging for Powell schemes, because the angled shock will introduce jumps in both normal components of \mathbf{B} , and the resulting numerical divergence will act as sources and sinks for conserved quantities through the Powell terms, thereby impacting the jump conditions. In addition, the initial shock at the convergence of the two regions is close to stationary with respect to the grid, and as a result, the magnetic field divergence cannot be efficiently advected away by the Powell terms.

Fig. 22 shows the errors on B_{\parallel} across a slice along x cutting through the shock region, for the DG-3 scheme. Both the Powell and hyperbolic cleaning schemes are shown.

For the Powell method, we find systematic errors in the 5% range in the inter-shock region, consistent with the results of Tóth (2000). Our measured errors seem however lower than those reported by Mignone & Tzeferacos (2010) in their similar test, where they measured relative errors greater than 10%. The Powell error may potentially be reduced by using a Riemann solver which propagates the divergence wave—for instance, the 8-wave Rusanov flux

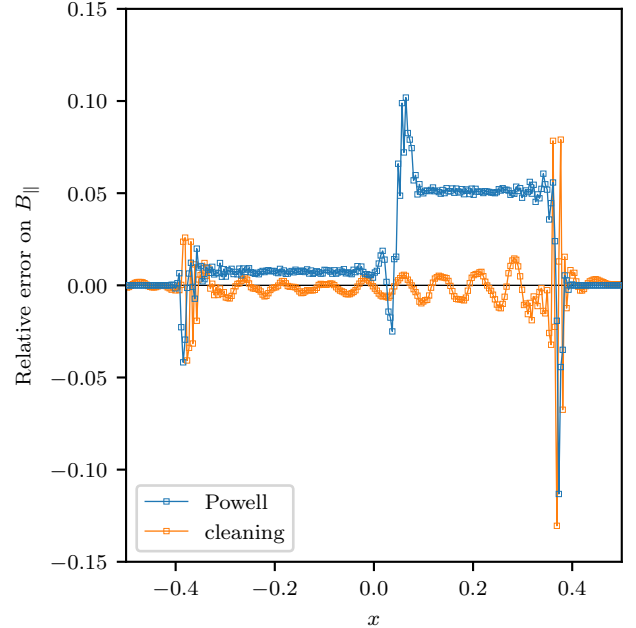


Figure 22. Rotated shock tube test. The relative error on the parallel magnetic field B_{\parallel} in the rotated shock tube test of Tóth (2000) is shown for the DG-3 method with the Powell and hyperbolic cleaning schemes. For the Powell method, between the left and right fast shocks ($x = \mp 0.4$), the value of B_{\parallel} deviates with respect to the exact solution due to the nonconservative source terms. Apart from oscillations at the discontinuities, the Powell scheme results in systematic offsets of B_{\parallel} of order $\lesssim 5\%$. The hyperbolic cleaning scheme features no such systematic offsets, but produces damped oscillations which overtake the fast shocks as the divergence gets advected away.

produces much smaller errors, but it is also extremely diffusive. We also noted in our experiments that the choice of face flux (see 4.1.3.2) plays a role in determining the location of the plateaus visible in Fig. 22, and it is possible that B_F^n may be chosen optimally as to minimize the absolute deviations in this particular test. Overall, this shows that DG cannot work around some of the pitfalls of nonconservative Powell schemes, for the fundamental reason that shocks are essentially first-order features.

The solution obtained with hyperbolic cleaning is also represented in Fig. 22. Unlike the Powell solution, it does not feature systematic shifts in the value of B_{\parallel} , however, large-scale oscillations due to the ongoing advection and damping of the divergence are visible. More importantly, these oscillations also occur for $|x_{\parallel}| \gtrsim 0.4$, outside of the region between the two shock fronts, which is an unphysical effect, and illustrates the fact that hyperbolic cleaning will propagate information faster than any physical MHD wave.

6 DISCUSSION

6.1 High-order schemes for astrophysics

Our tests have shown some promising results for higher-order schemes in astrophysical simulations, and for MHD in particular. We now discuss how the combination of DG with adaptive mesh refinement could allow very efficient computations in smooth regions of the flow with high order convergence, while finely capturing the shocks and discontinuities with spatial refinement.

We can interpret order/resolution convergence plots such as

Fig. 4 from two complementary points of view: looking at a given spatial resolution on the x axis, we can increase the order to reduce the solution error. But at a fixed error on the y axis, we may also increase the order and correspondingly reduce the spatial resolution. We argue that this second vision is more relevant to many types of simulations in astrophysics, as the spatial truncation errors need only be smaller or comparable to other types of errors, stemming from uncertainties in the physical models, missing physics, subgrid recipes, etc. We can therefore see order convergence as a way of getting away with fewer cells in smooth problems, to the extent that we can efficiently “patch” smooth regions of the flow with coarser cells.

Cell-based adaptive mesh refinement provides a suitable framework to do this, as it allows cell-by-cell resolution adaptivity to match the local feature size. In a number of our test problems, we found higher-order to better resolve features close to the grid resolution. The vortex problem of Fig. 4 illustrates the advantage of higher orders for capturing features which are barely resolved by the grid (shaded area): at 32^2 resolution, the 2^{nd} order scheme is not yet resolving the vortex, whereas the DG-4 method has already achieved its theoretical 4^{th} order convergence. The same conclusion holds in the presence of shocks, as illustrated by the Shu-Osher MHD shock tube in Fig. 9 where we see a significant improvement from DG-2 to DG-3. The loop advection problem further shows that sharp features (such as singular field derivatives appearing in the MHD current) can also be captured within one cell by moderately increasing the spatial order: going from 2^{nd} to 3^{rd} or 4^{th} order dramatically improves the loop sharpness, while reducing the dissipation of magnetic energy.

Compared to Lagrangian methods, a major issue with AMR Eulerian grid codes is that they require sufficient grid resolution to avoid dissipation due to bulk flow velocities; i.e. they are only Galilean invariant for solutions sufficiently resolved to make advection errors negligible. Because spatial resolution translates into tighter CFL constraints on the timestep, a compromise has to be reached between advection errors and compute time in practice. The advected Orszag–Tang test of Section 5.3.4 demonstrates that not only do higher order schemes help reduce advection errors and restore Galilean invariance, but for smooth regions of the flow, it can actually be beneficial to increase the order while reducing spatial resolution. Note that these test problems present MHD shocks, and it is encouraging to see that these positive features remain, even though we find that they can be sensitive to the details of the limiter settings.

The combination of higher order methods with adaptive mesh refinement therefore seems particularly powerful. We note that for most astrophysical situations, spatial refinement will likely be required, because of the presence of shocks which are inherently first-order features, but also whenever the fluid is self-gravitating. One may therefore ask what scheme order will turn out to be the optimal choice for a given problem. While we discussed positive effects of higher orders, in practice we expect diminishing returns. From the above discussion, it is clear the optimal global scheme order will depend on the volume filling fraction and geometry of shocks and other discontinuities, as well as the acceptable truncation error, both of which are very problem-dependent. Higher orders will only be helpful to the extent that we can efficiently patch smooth regions of the flow with coarser and coarser cells. In addition, the computational cost of DG becomes prohibitive for large orders, in part because of the expensive quadrature operations, but also because of the more restrictive CFL condition (24). We note that this CFL constraint can be relaxed within the DG framework, for example

using so-called PNPM schemes (Dumbser et al. 2008), where N moments are evolved dynamically as in DG, whereas high-order spatial reconstruction is used up to order $M \geq N$ to recover the remaining moments; however this comes at the cost of a more extended pattern of ghost cells, as with purely reconstruction-based schemes.

Time integration for RKDG schemes is also both computationally and memory-expensive at high temporal orders, as Runge-Kutta methods require multiple steps with intermediate storage. These memory and computational requirements may be reduced using other time integration schemes such as ADER (see 6.4.2). Note that even though we match the RK time integration order to the spatial order of the scheme (up to RK4) in most of our runs, this is only done to ensure that the time integration errors do not contaminate the convergence tests of Section 5.2. In practice, high-order RK time integration may be unnecessary, for example when the CFL criterion enforces very small time steps due to high plasma temperatures, or in presence of shorter time-scales in other physical processes.

For these reasons, based on the test problems presented in this work, we expect third or fourth-order schemes (DG-3, DG-4) to achieve a good overall compromise when combined with adaptive mesh refinement. Anninos et al. (2017) have reported similar practical limits for their astrophysical DG scheme, based on compute time and floating point machine precision considerations.

6.2 Discontinuous Galerkin for high-order

Explicit DG schemes are often touted as a promising family of methods for high performance computing, because they are both very local (in the sense that information about all derivatives can be obtained locally within a cell, instead of relying on multiple neighbouring cells as with reconstruction methods), and compute-intensive. Both properties are regarded as desirable in combination, as they shift the load away from the communication and memory subsystems towards CPU cores, which translates into greater parallel scalability. In addition, most of the compute operations in DG involve dense linear algebra and can be readily optimized to make use of the powerful floating point units of modern CPUs.

However, in practice, exploiting the full compute capabilities of modern CPUs, such as vector units and fused multiply-add instructions, requires exposing operations to the compiler at the right level of loop granularity, which mandates careful organization of the code. In some places, we resort to temporary data structures to pack and unpack data elements to benefit from vectorization. We believe such optimization efforts are necessary if one intends to reap the benefits of DG, because of the large additional computational costs compared to reconstruction methods.

In addition to being an asset for achieving high compute efficiency, the local character of derivatives in DG could also have implications for the implementation of physical models. We find that we can obtain a very sharp and clean signal for the derivatives within a cell, as illustrated by the loop advection problem of Section 5.4.1: the MHD current is readily evaluated locally in each cell from the DG weights, and higher orders significantly improve the capturing of singular features in the current (Fig. 19). We believe this property of DG schemes could have important applications for the modeling of physical processes which require accurate derivatives of the fields, such as for example shock detection (Schaal 2016), or cosmic ray streaming in magnetic fields (e.g. Wiener et al. 2013; Jiang & Oh 2018).

6.3 Central divergence control methods

In this paper, we have explored the implementation of cell-centered divergence control techniques, in the form of the 8-wave formalism with Powell terms, and hyperbolic divergence cleaning. We find that although these techniques can be applied successfully to MHD test problems with DG, they are no silver bullet against all divergence-related effects, in particular if we insist on using the well-tested and low-diffusion HLLD fluxes.

6.3.1 Hyperbolic divergence cleaning

We find that the hyperbolic cleaning method of [Dedner et al. \(2002\)](#) lends itself very well to a straightforward implementation with DG. The scheme is very effective at eliminating divergence in test problems where a single spatial and temporal scale can be readily identified. In addition, a major advantage is that we find the scheme to be equally stable with LDF but also Legendre bases for the magnetic field. As a result, hyperbolic cleaning has been a popular implementation of cell-based divergence control in DG-MHD codes (e.g. [Zanotti et al. 2015](#); [Dumbser & Loubère 2016](#); [Kidder et al. 2017](#)). The scheme has a number of drawbacks however, which impact its suitability for large dynamic range simulations.

First, the scheme requires at least one⁵ dimensional parameter to be set (the cleaning speed c_h), which should be chosen faster than the fastest MHD wave at all points and all times in the simulation. In the formulation of [Dedner et al. \(2002\)](#), c_h is taken to be uniform in space and constant in time, which is in practice intractable for simulations with large dynamic range because of the resulting global CFL constraint. [Tricco et al. \(2016\)](#) showed that using the original divergence cleaning formulation with time-varying c_h could result in locally creating divergence, which we have observed in our test simulations in presence of adaptive mesh refinement. To overcome this issue, they developed locally varying cleaning speeds in both space and time in the context of Lagrangian SPH methods. Adapting this work to Eulerian grids could help alleviate the problem of choosing hyperbolic cleaning speeds for problems with large dynamic range.

Beyond the question of parameter tuning, hyperbolic cleaning can also produce nonphysical effects, because it will propagate disturbances in the magnetic field at velocity c_h , i.e. faster than any physical wave of ideal MHD. We have seen this effect with the rotated shock tube setup (Fig. 22). In our test problems, the effect is most impacting on the 3D blast wave problem, where the issue is compounded by the very low ambient plasma β of this particular setup: hyperbolic cleaning will cause fluctuations in the magnetic energy to propagate outwards ahead of the shock into the background medium, resulting in a loss of positivity of the thermal pressure. In science applications, this has been found to produce spurious bending of field lines ahead of bow shocks by e.g. [Susanto \(2014\)](#). Approaches have been developed to address positivity issues with divergence cleaning, either by ensuring that the magnetic energy can only be decreased by the scheme ([Tricco & Price 2012](#)), or through an entropy-stable formulation with entropy variables ([Derigs et al. 2018](#)). We have not investigated these directions in this work, but leave them as potential future improvements.

⁵ As discussed in Section 4.1.5, our current implementation requires three dimensional parameters, but they can be reduced to only one using the parameterization of [Mignone & Tzeferacos \(2010\)](#).

6.3.2 Powell scheme

In comparison to hyperbolic divergence cleaning, the Powell source term approach does not have any free parameter, and is therefore intrinsically scale-free. We find that for most test problems, the Powell approach produces results close to hyperbolic cleaning, and in particular our implementation seems to perform markedly better than similar pure source term results in other published works ([Hopkins 2016](#); [Hopkins & Raives 2016](#); [Derigs et al. 2018](#)). The method handles strong shocks and low plasma- β situations very robustly, as demonstrated in the 3D blast problem. Interestingly, like [Balsara & Spicer \(1999b\)](#), we find that isolated moving shocks are not the strongest source of divergence with the Powell scheme. Rather, interacting or standing shocks, and stagnation points in smooth flows, are most at risk of local corruption of the solution due to divergence, because the Powell scheme cannot advect it efficiently away. Typical test setups exhibiting these issues are the rotated shock tube and the Orszag–Tang vortex, the former featuring a standing shock, and the latter smooth stagnating flows, “pinch” points, and interacting shocks.

An obvious drawback of the Powell method is its nonconservative character due to the source term which injects conserved quantities. It is difficult to comprehensively assess the exact impact of this source term on the quality of the solution. In terms of *global* conservation, we found only a small impact of the source term in the Orszag–Tang vortex problem (Fig. 11), but we stress that the impact could get more severe over long time-scales. The rotated shock tube provides an even more stringent test of *local* conservation properties, and we measure systematic deviations of about 5% in the magnetic field strength along the problematic direction. We note that this error does not converge away with scheme order, since the shock is intrinsically a first-order feature. In addition, we also observe that this error is constant with resolution as noted by [Tóth \(2000\)](#), because it is caused by the effect of the *integrated* Powell term which is singular at the jump, and for which the cell size only acts as a regularization scale. However, outside of shocks where the solution is smooth (possibly with steep gradients), additional resolution produces smaller jumps at interfaces, thereby reducing the overall amount of divergence. This is consistent with the findings of other studies (e.g. [Pakmor & Springel 2013](#); [Zhu et al. 2015](#)). In addition to resolution, we showed that increasing the scheme order can very efficiently reduce the divergence in smooth regions of the flow, as soon as the flow features are at least partially resolved (Fig. 5).

These non-conservative properties are a known weak point of Powell schemes, and our study shows that DG and higher-order methods are not immune to them. Our Powell scheme seems however very robust in high-Mach, low- β environments, as illustrated by the 3D blast test. We note that nonconservative source terms seem to be a recurring ingredient of cell-centered schemes with good positivity properties. [Mignone & Tzeferacos \(2010\)](#) noticed that the behaviour of hyperbolic cleaning in the presence of shocks is improved by the addition of source terms (the co-called EGLM approach of [Dedner et al. 2002](#)). Interestingly, entropy stability—which is related to positivity and robust behaviour at shocks—seems to call for non-conservative source terms, which are themselves at odds with the underlying foundational formulation as conservation laws ([Chandrashekar & Klingenberg 2016](#); [Derigs et al. 2018](#)).

The divergence control provided by the Powell scheme with our DG method is unfortunately not a turnkey solution. While we did not experience divergence runaways with LDF bases and the Powell term prescriptions discussed in 4.1.3 across all of our test runs, the

resulting solutions can, for some limiter settings, show signs of divergence damage when compared to runs with hyperbolic cleaning. Generally speaking, if the characteristic slope limiter threshold \tilde{M} is set too high (i.e. if the limiter is too lenient), spurious extrema will contaminate all waves in the solution, a problem which is of course not specific to the magnetic field divergence. The current sheet problem of Section 5.4.2 illustrates a case where the Powell method suffers from too permissive limiter settings. We also noticed, however, that \tilde{M} may conversely be set *too low*, so that the limiter becomes too aggressive and restricts the propagation of the 8th divergence wave, thereby reducing the efficiency of divergence control. This can paradoxically result in divergence damage being greater with a more aggressive limiter setting. This effect seems particularly noticeable on the Orszag–Tang vortex problem. This does not happen with other variables or MHD waves, because they do not rely on one of the characteristics for stabilization against a runaway process.

Another important and likely related aspect of the Powell scheme in our DG implementation is that it seems to require LDF bases to achieve stability. Expanding the magnetic field components on a Legendre basis, and accounting for a volume contribution of the Powell term to the DG integrals is not sufficient to ensure stability, and can result in uncontrolled divergence blowup.

These effects highlight the complex interactions between different ingredients of the Powell scheme, which we attribute to the non-linear nature of the Powell source term. We note that the slope limiter used in this work is rather primitive, and other limiters could well help resolve this issue. We review some possible limiter improvements in the dedicated discussion Section 6.4.1.

6.3.3 Locally divergence-free bases

As discussed above, we find that the LDF basis functions discussed in this paper are necessary for the stability of our Powell scheme. They are, however, in no way sufficient as soon as a non-diffusive Riemann solver such as HLLD is employed. More diffusive Riemann solvers, such as Rusanov fluxes, or entropy-stable fluxes of Chandrashekar & Klingenberg (2016), can achieve stability with LDF bases and no other divergence treatment on simple problems: these fluxes are generally more diffusive than HLLD, and in particular include a normal component flux for the magnetic field which will dampen the normal jump of \mathbf{B} at the faces. However, in the presence of strong shocks, we find that the HLLD solver provides sharper solutions, and therefore constitutes a more sensible choice for high-order codes attempting to resolve features close to the grid resolution.

LDF basis functions require significant additional complexity in the code and in the scheme itself, from basis function generation (see Appendix A) to special limiting procedures (such as the one we introduced in 4.2.4). Because LDF bases make use of the divergence-free condition, they require somewhat less storage than a component-wise Legendre basis for \mathbf{B} .

However, LDF bases turn out to be potentially more computationally expensive than Legendre bases for high orders: the Legendre basis (13) possesses a tensor product structure (component \times basis function) which allows optimized computation of conserved states \mathbf{u} and Gaussian quadrature using sum factorization (see Kronbichler et al. 2017). These optimizations cannot be readily used with LDF bases, because the coupling between components of \mathbf{B} breaks the tensor product structure. In addition, precomputing and storing basis function values and gradients at quadrature points requires more memory for the LDF basis than the Legendre basis, because

all 3 components of \mathbf{B} have to be stored (in addition to the Legendre basis, which is still required for hydro variables). The memory footprint of the precomputed basis function data can exceed the size of the L1 processor cache, and for orders ≥ 4 in 3D, the quadrature data for the LDF+Powell scheme will not even fit in a 256 kB L2 cache, resulting in a strong performance degradation. From a few practical tests, we find that sum factorization optimizations and careful memory footprint management can make a significant difference for orders ≥ 4 , so users of very high order schemes may want to reconsider the use of divergence-free bases on computational efficiency grounds.

Finally, the convergence tests of Section 5.2 have shed light on a subtle instability with LDF bases that reveals itself with HLLD fluxes in non-grid-aligned configurations. We speculate that this may be related to the projection effects discussed in Appendix B. While we do not see this as a source of concern for actual simulations, we note that the effective convergence order with LDF bases may in some cases be lower than expected.

6.3.4 Conclusion on divergence control

A natural way around the shortcomings of both schemes would be to rely on exactly globally divergence-free schemes. Such methods have been proposed for DG, for example based on vector potential techniques (e.g. Rossmannith 2013). High-order versions of constrained transport have also been developed, where the magnetic field is expanded on the interfaces, reconstructed within the cell using a divergence-free reconstruction, and evolved using a distinct evolution equation (see e.g. Li et al. 2011; Balsara & Käppeli 2017). These methods involve complicated ingredients: they require more complex mesh structures, multidimensional Riemann solvers to obtain the electric fields, and higher-order expansions for the magnetic field inside the cells. In addition, because the dynamical degrees of freedom are split between cell- and face-centered discretizations, it is less clear how to define a proper limiting procedure for high-order modes.

Although we find that both hyperbolic cleaning and the Powell methods have shortcomings which are a source for concern for large dynamic range simulations, we note that these methods have been successfully used in practice in production in a finite volume context. The extensive code comparison study of Kritsuk et al. (2011), focusing on decaying isothermal supersonic turbulence, showed no significant systematic effect of the divergence control method on a number of physically relevant quantities; the authors demonstrate that other aspects of codes are more important for accurate turbulence simulations. In particular, they find that high-order codes achieve increased spectral bandwidth and effective Reynolds number. In the context of the magneto-rotational instability (MRI), Flock et al. (2010) see little difference between Powell and upwinded constrained transport schemes on MRI growth rates and evolution of the magnetic energy.

In any case, we plan to apply both hyperbolic divergence cleaning and the Powell scheme to MHD turbulence simulations, for which the impact of divergence control can be readily tested against published results.

6.4 Future improvements

Our current DG scheme relies on a number of simple ingredients which will likely require improvement for use in production astrophysical science applications. The main current directions will be

to replace the simple DG limiter described in 4.2 by more sophisticated alternatives, add support for local timestepping, and improve $\nabla \cdot \mathbf{B}$ control, even within the framework of cell-centered divergence control schemes. Fortunately, solutions to some of these issues have already been developed in the DG context, and we plan to implement them as future improvements.

6.4.1 Limiters

Our limiting procedure discussed in 4.2 relies on the choice of a dimensional parameter \tilde{M} (or equivalently M), which, although sufficient for smooth test problems where a single spatial scale dominates, is not convenient for problems involving large dynamic range, or for which the spatial variation of the solution is not well controlled. In addition, acting as a slope limiter, it can only ever retain second-order information in the cell whenever the limiter is triggered.

A number of more sophisticated limiters have been developed and successfully applied in the DG literature. The WENO approach has given rise to a number of DG limiters which preserve and correct higher-order moments of the solution (see e.g. Qiu & Shu 2004; Balsara et al. 2007; Zhong & Shu 2013). Divergence-free WENO reconstruction schemes have been developed for MHD (Balsara & Dumbser 2015; Zhao & Tang 2017), and could form a suitable basis for DG limiters with LDF bases.

More recently, so-called *a posteriori* subcell limiters have been developed by Clain et al. (2011); Sonntag & Munz (2014); Dumbser et al. (2014); Dumbser & Loubère (2016). They rely on after-the-fact detection of solution defects by first taking a timestep and projecting the resulting tentative solution in each cell onto a finer subcell grid. The subcell grid is then inspected for loss of positivity or spurious oscillations. In case any subcell is marked as troubled, the solution in the parent cell is rolled back to the previous time point, its subcell values are recomputed and evolved in time using a lower-order but more robust scheme, and the updated subcell grid is used to reconstruct the DG moments of the parent cell at the final time. This technique has been successfully used for DG implementations of MHD (Zanotti et al. 2015; Fambri et al. 2018). In particular, simple criteria for solution admissibility have been developed, and subcell oscillation detection can be performed without dimensional parameters (Dumbser et al. 2014). We intend to replace our current limiter with this promising technique in the near future.

6.4.2 Local timestepping with ADER

While some astrophysical problems only span a limited range of spatial and temporal scales (for example, simulations of idealized isotropic isothermal turbulence in absence of self-gravity), many simulation setups require spatial resolution with AMR, or feature strongly locally-varying signal speeds. In the latter case, the CFL constraint (24) will impose a maximum Δt which can vary significantly across cells, and it becomes beneficial to update each cell at its own pace according to its maximal Δt using local timestepping.

The RKDG method used in our scheme is not easily amenable to local timestepping, because it treats the whole hyperbolic problem as a set of coupled ordinary differential equations through the semidiscrete scheme (23), in which the *whole* vector of all cell weights is integrated forward in time using a Runge-Kutta method.

In the DG context, local timestepping may be achieved using the so-called ADER methods, introduced by Titarev & Toro (2002); Dumbser & Munz (2006); Dumbser et al. (2007), and more

recently reformulated by Dumbser et al. (2008). In its modern variant, ADER exploits a DG-like weak formulation on a high-order spacetime basis to first obtain a high-order predictor state local to each cell. This state is then used within the DG framework to perform time integration as a predictor-corrector scheme. This formulation has been used by e.g. Zanotti et al. (2015); Fambri et al. (2017) to develop ADER-DG codes with adaptive mesh refinement and local timestepping. Very recently, Charrier & Weinzierl (2018) have shown that this ADER-DG predictor-corrector time integration scheme can be made efficient on HPC architectures, with a reduced memory footprint, increased compute intensity and optimized distributed memory data exchanges. The above properties make ADER an attractive time integration algorithm for large dynamic range astrophysical applications, as an improvement over our current RKDG scheme.

6.4.3 Divergence control

We have discussed divergence control in detail in 6.3. Short of implementing a full-fledged high-order version of constrained transport, we would like to try to adapt improvements of the hyperbolic cleaning methodology (following Tricco & Price 2012; Mignone & Tzeferacos 2010; Tricco et al. 2016) to a Eulerian DG framework, to allow fully adaptive cleaning speeds, and to reduce positivity issues with hyperbolic cleaning for very strong shocks. For the Powell approach, a promising direction would be to experiment with an 8-wave version of the HLLD flux, such as proposed by Fuchs et al. (2011).

7 CONCLUSIONS

In this paper, we have described our implementation of a discontinuous Galerkin scheme with adaptive mesh refinement for magnetohydrodynamics, implemented within the framework of the astrophysical code AREPO. Our scheme relies on the Runge-Kutta DG framework, and can use two different types of cell-centered divergence control techniques: locally divergence-free bases for the magnetic field together with Powell terms to control the global divergence, and hyperbolic cleaning, which may be used with the same Legendre basis functions used for hydrodynamical variables. We have introduced two main new numerical ingredients: a non-linear limiting procedure for the magnetic field components, and different discretizations of the Powell source term, which we found to be a key ingredient for the stability and accuracy of the scheme. We have also discussed some subtle numerical and performance-related properties of locally divergence-free bases, which we encountered during our development of the scheme.

We have shown that the resulting method is accurate across a wide range of typical MHD test problems, and can achieve the expected theoretical order of convergence. Like previous authors, we found that higher-order schemes can efficiently reduce advection errors in Eulerian grid codes, and we further showed that these results hold even with a simple limiting procedure and in presence of MHD shocks.

We showed that increasing the DG order of the scheme can benefit solutions which are barely resolved by the grid size, by reducing not only numerical diffusion but also magnetic field divergence more efficiently than can be achieved by increasing grid resolution. We argue that this makes higher order methods very attractive for resolution-limited astrophysical simulations. In addition, DG provides instant access to local derivatives within the cell

without smearing across neighbour stencils, which can be important for astrophysical applications where a clean derivative signal is required for the physics, such as shock detection or models of cosmic ray streaming.

We discussed the divergence control performance of the Powell scheme and hyperbolic cleaning, and found that while both methods perform well across most test problems, neither of them in our current implementation was fully satisfactory across all flow regimes.

Finally, we covered a few of the future improvements planned for our current DG implementation, which we see as a base development platform for increasingly sophisticated high-order solvers in the AREPO code, in which to integrate more and more physical models while developing the solvers into highly scalable implementations.

We are now planning to further test and improve the scheme with high resolution simulations of MHD turbulence, which will provide insights on the effectiveness of our numerical ingredients—in particular the two divergence control techniques—and more generally contribute to shedding light on the potential of DG methods for astrophysics.

ACKNOWLEDGEMENTS

TG would like to thank Markus Zenk, Philipp Girichidis, Christoph Pfrommer, Matthew Bate and Pascal Tremblin for very helpful discussions. The authors gratefully acknowledge the support of the Klaus Tschira Foundation. We acknowledge financial support through subproject EXAMAG of the Priority Programme 1648 “SPPEXA” of the German Science Foundation, and through the European Research Council through ERC-StG grant EXAGAL-308037. PC acknowledges support from the Airbus Foundation Chair on Mathematics of Complex Systems at TIFR-CAM, Bangalore.

REFERENCES

Anninos P., Bryant C., Fragile P. C., Holgado A. M., Lau C., Nemergut D., 2017, *Astrophys. J. Suppl. Ser.*, 231, 17

Balsara D. S., 1998, *Astrophys. J. Suppl. Ser.*, 116, 133

Balsara D. S., 2004, *Astrophys. J. Suppl. Ser.*, 151, 149

Balsara D. S., 2009, *J. Comput. Phys.*, 228, 5040

Balsara D. S., 2017, *Living Rev. Comput. Astrophys.*, 3, 2

Balsara D. S., Dumbser M., 2015, *J. Comput. Phys.*, 299, 687

Balsara D. S., Käppeli R., 2017, *J. Comput. Phys.*, 336, 104

Balsara D. S., Spicer D., 1999a, *J. Comput. Phys.*, 148, 133

Balsara D. S., Spicer D. S., 1999b, *J. Comput. Phys.*, 149, 270

Balsara D. S., Altmann C., Munz C.-D., Dumbser M., 2007, *J. Comput. Phys.*, 226, 586

Balsara D. S., Rumpf T., Dumbser M., Munz C.-D., 2009, *J. Comput. Phys.*, 228, 2480

Bauer A., Springel V., 2012, *Mon. Not. R. Astron. Soc.*, 423, 2558

Boscheri W., Dumbser M., 2017, *J. Comput. Phys.*, 346, 449

Boscheri W., Dumbser M., Balsara D., 2014, *Int. J. Numer. Methods Fluids*, 76, 737

Brackbill J. U., Barnes D. C., 1980, *J. Comput. Phys.*, 35, 426

Brio M., Wu C. C., 1988, *J. Comput. Phys.*, 75, 400

Chakravarthy V. K., Arora K., Chakraborty D., 2015, *Int. J. Numer. Methods Fluids*, 77, 707

Chandrashekar P., Klingenberg C., 2016, *SIAM J. Numer. Anal.*, 54, 1313

Charrier D. E., Weinzierl T., 2018, arXiv:1801.08682 [cs]

Clain S., Diot S., Loubère R., 2011, *J. Comput. Phys.*, 230, 4028

Cockburn B., Shu C.-W., 1989, *Math. Comput.*, 52, 411

Cockburn B., Lin S.-Y., Shu C.-W., 1989, *J. Comput. Phys.*, 84, 90

Cockburn B., Li F., Shu C.-W., 2004, *J. Comput. Phys.*, 194, 588

Collins D. C., Xu H., Norman M. L., Li H., Li S., 2010, *Astrophys. J. Suppl. Ser.*, 186, 308

Dai W., Woodward P. R., 1998, *Astrophys. J.*, 494, 317

Dedner A., Kemm F., Kröner D., Munz C.-D., Schnitzer T., Wesenberg M., 2002, *J. Comput. Phys.*, 175, 645

Derigs D., Winters A. R., Gassner G. J., Walch S., 2016, *J. Comput. Phys.*, 317, 223

Derigs D., Winters A. R., Gassner G. J., Walch S., Böhm M., 2018, *Journal of Computational Physics*, 364, 420

Dumbser M., Loubère R., 2016, *J. Comput. Phys.*, 319, 163

Dumbser M., Munz C.-D., 2006, *J. Sci. Comput.*, 27, 215

Dumbser M., Käser M., Toro E. F., 2007, *Geophys. J. Int.*, 171, 695

Dumbser M., Balsara D. S., Toro E. F., Munz C.-D., 2008, *J. Comput. Phys.*, 227, 8209

Dumbser M., Zanotti O., Loubère R., Diot S., 2014, *J. Comput. Phys.*, 278, 47

Evans C. R., Hawley J. F., 1988, *Astrophys. J.*, 332, 659

Fambri F., Dumbser M., Zanotti O., 2017, *Comput. Phys. Commun.*, 220, 297

Fambri F., Dumbser M., Köppel S., Rezzolla L., Zanotti O., 2018, *Mon. Not. R. Astron. Soc.*, 477, 4543

Federrath C., Roman-Duval J., Klessen R. S., Schmidt W., Mac Low M.-M., 2010, *Astron. Astrophys.*, 512, A81

Flock M., Dzyurkevich N., Klahr H., Mignone A., 2010, *Astron. Astrophys.*, 516, A26

Fromang S., Hennebelle P., Teyssier R., 2006, *Astron. Astrophys.*, 457, 371

Fryxell B., et al., 2000, *Astrophys. J. Suppl. Ser.*, 131, 273

Fuchs F. G., McMurry A. D., Mishra S., Risebro N. H., Waagan K., 2011, *Commun. Comput. Phys.*, 9, 324

Gaburov E., Nitadori K., 2011, *Mon. Not. R. Astron. Soc.*, 414, 129

Gardiner T. A., Stone J. M., 2005, *J. Comput. Phys.*, 205, 509

Godunov S. K., 1972, *Numer. Methods Mech. Contin. Medium*, 1, 26

Gottlieb S., 2005, *J. Sci. Comput.*, 25, 105

Hawley J. F., Stone J. M., 1995, *Comput. Phys. Commun.*, 89, 127

Helzel C., Rossmanith J. A., Taetz B., 2011, *J. Comput. Phys.*, 230, 3803

Hopkins P. F., 2016, *Mon. Not. R. Astron. Soc.*, 462, 576

Hopkins P. F., Raives M. J., 2016, *Mon. Not. R. Astron. Soc.*, 455, 51

Janhunen P., 2000, *J. Comput. Phys.*, 160, 649

Jiang Y.-F., Oh S. P., 2018, *Astrophys. J.*, 854, 5

Jiang G.-S., Wu C.-c., 1999, *J. Comput. Phys.*, 150, 561

Karami Halashi B., Luo H., 2016, *J. Comput. Phys.*, 326, 258

Kemm F., 2013, *Commun. Appl. Math. Comput. Sci.*, 8, 1

Kidder L. E., et al., 2017, *J. Comput. Phys.*, 335, 84

Kritsuk A. G., et al., 2011, *Astrophys. J.*, 737, 13

Kronbichler M., Kormann K., Pasichnyk I., Allalen M., 2017, in *High Performance Computing. Lecture Notes in Computer Science*. Springer, Cham, pp 237–255, doi:10.1007/978-3-319-58667-0_13

LeVeque R. J., 2002, *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press

Li F., Shu C.-W., 2005, *J. Sci. Comput.*, 22-23, 413

Li F., Xu L., 2012, *J. Comput. Phys.*, 231, 2655

Li F., Xu L., Yakovlev S., 2011, *J. Comput. Phys.*, 230, 4828

Liu Y., Shu C.-W., Zhang M., 2018, *J. Comput. Phys.*, 354, 163

Mac Low M.-M., 1999, *Astrophys. J.*, 524, 169

Marinacci F., Vogelsberger M., Kannan R., Mocz P., Pakmor R., Springel V., 2018, *Mon. Not. R. Astron. Soc.*, 476, 2476

Meurer A., et al., 2017, *PeerJ Comput. Sci.*, 3, e103

Mignone A., Tzeferacos P., 2010, *J. Comput. Phys.*, 229, 2117

Mignone A., Bodo G., Massaglia S., Matsakos T., Tesileanu O., Zanni C., Ferrari A., 2007, *Astrophys. J. Suppl. Ser.*, 170, 228

Mignone A., Zanni C., Tzeferacos P., van Straalen B., Colella P., Bodo G., 2012, *Astrophys. J. Suppl. Ser.*, 198, 7

Miyoshi T., Kusano K., 2005, *J. Comput. Phys.*, 208, 315

Mocz P., Vogelsberger M., Sijacki D., Pakmor R., Hernquist L., 2014, *Mon. Not. R. Astron. Soc.*, 437, 397

Mocz P., Pakmor R., Springel V., Vogelsberger M., Marinacci F., Hernquist L., 2016, *Mon. Not. R. Astron. Soc.*, 463, 477

Nelson D., Vogelsberger M., Genel S., Sijacki D., Kereš D., Springel V., Hernquist L., 2013, *Mon. Not. R. Astron. Soc.*, 429, 3353
 Pakmor R., Springel V., 2013, *Mon. Not. R. Astron. Soc.*, 432, 176
 Pakmor R., Bauer A., Springel V., 2011, *Mon. Not. R. Astron. Soc.*, 418, 1392
 Pakmor R., Marinacci F., Springel V., 2014, *Astrophys. J. Lett.*, 783, L20
 Pakmor R., et al., 2017, arXiv:1701.07028 [astro-ph]
 Powell K. G., 1994, Technical report, Approximate Riemann Solver for Magnetohydrodynamics (That Works in More than One Dimension). NASA ICASE
 Powell K. G., Roe P. L., Linde T. J., Gombosi T. I., De Zeeuw D. L., 1999, *J. Comput. Phys.*, 154, 284
 Qiu J., Shu C.-w., 2004, *SIAM J. Sci. Comput.*, 26, 907
 Rieder M., Teyssier R., 2016, *Mon. Not. R. Astron. Soc.*, 457, 1722
 Rieder M., Teyssier R., 2017, *Mon. Not. R. Astron. Soc.*, 472, 4368
 Robertson B. E., Kravtsov A. V., Gnedin N. Y., Abel T., Rudd D. H., 2010, *Mon. Not. R. Astron. Soc.*, 401, 2463
 Robinet J.-C., Gressier J., Casalis G., Moschetta J.-M., 2000, *J. Fluid Mech.*, 417, 237
 Rossmannith J., 2006, *SIAM J. Sci. Comput.*, 28, 1766
 Rossmannith J. A., 2013, arXiv:1310.4251 [math.NA]
 Ryu D., Ostriker J. P., Kang H., Cen R., 1993, *Astrophys. J.*, 414, 1
 Ryu D., Miniati F., Jones T. W., Frank A., 1998, *Astrophys. J.*, 509, 244
 Schaal K., 2016, PhD thesis, Universität Heidelberg
 Schaal K., Bauer A., Chandrashekar P., Pakmor R., Klingenberg C., Springel V., 2015, *Mon. Not. R. Astron. Soc.*, 453, 4279
 Schekochihin A. A., Cowley S. C., Taylor S. F., Maron J. L., McWilliams J. C., 2004, *Astrophys. J.*, 612, 276
 Shu C.-W., 1987, *Math. Comput.*, 49, 105
 Shu C.-W., Osher S., 1989, in , Upwind and High-Resolution Schemes. Springer, Berlin, Heidelberg, pp 328–374, doi:10.1007/978-3-642-60543-7_14
 Sonntag M., Munz C.-D., 2014, in Springer Proceedings in Mathematics & Statistics, Finite Volumes for Complex Applications VII-Elliptic, Parabolic and Hyperbolic Problems. Springer, Cham, pp 945–953, doi:10.1007/978-3-319-05591-6_96
 Spiegel S. C., Huynh H. T., DeBonis J. R., 2015, in , 22nd AIAA Computational Fluid Dynamics Conference. American Institute of Aeronautics and Astronautics, doi:10.2514/6.2015-2444
 Springel V., 2010, *Mon. Not. R. Astron. Soc.*, 401, 791
 Stone J. M., Norman M. L., 1992, *Astrophys. J. Suppl. Ser.*, 80, 791
 Stone J. M., Ostriker E. C., Gammie C. F., 1998, *Astrophys. J. Lett.*, 508, L99
 Stone J. M., Gardiner T. A., Teuben P., Hawley J. F., Simon J. B., 2008, *Astrophys. J. Suppl. Ser.*, 178, 137
 Susanto A., 2014, PhD thesis, University of Waterloo
 Teyssier R., 2002, *Astron. Astrophys.*, 385, 337
 Titarev V. A., Toro E. F., 2002, *J. Sci. Comput.*, 17, 609
 Tóth G., 2000, *J. Comput. Phys.*, 161, 605
 Tricco T. S., Price D. J., 2012, *J. Comput. Phys.*, 231, 7214
 Tricco T. S., Price D. J., Bate M. R., 2016, *J. Comput. Phys.*, 322, 326
 Waagan K., 2009, *J. Comput. Phys.*, 228, 8609
 Waagan K., Federrath C., Klingenberg C., 2011, *J. Comput. Phys.*, 230, 3331
 Wang P., Abel T., 2009, *Astrophys. J.*, 696, 96
 Wang C., Zhang X., Shu C.-W., Ning J., 2012, *J. Comput. Phys.*, 231, 653
 Warburton T. C., Karniadakis G. E., 1999, *J. Comput. Phys.*, 152, 608
 Wiener J., Oh S. P., Guo F., 2013, *Mon. Not. R. Astron. Soc.*, 434, 2209
 Winters A. R., Gassner G. J., 2016, *J. Comput. Phys.*, 304, 72
 Xu Z., Liu Y., 2016, *J. Comput. Phys.*, 327, 203
 Yakovlev S., Xu L., Li F., 2013, *J. Comput. Sci.*, 4, 80
 Zanotti O., Fambri F., Dumbser M., 2015, *Mon. Not. R. Astron. Soc.*, 452, 3010
 Zhang X., Shu C.-W., 2010, *J. Comput. Phys.*, 229, 8918
 Zhao J., Tang H., 2017, *J. Comput. Phys.*, 343, 33
 Zhao X., Yang Y., Seyler C. E., 2014, *J. Comput. Phys.*, 278, 400
 Zhong X., Shu C.-W., 2013, *J. Comput. Phys.*, 232, 397
 Zhu W., Feng L.-l., Xia Y., Shu C.-W., Gu Q., Fang L.-Z., 2013, *Astrophys. J.*, 777, 48

Zhu C., Pakmor R., van Kerkwijk M. H., Chang P., 2015, *Astrophys. J. Lett.*, 806, L1

APPENDIX A: LOCALLY DIVERGENCE-FREE BASIS

A1 Divergence-free vector space

In this section, we take D to be the number of spatial dimensions, and work in spatial coordinates of the reference cell, $(\xi_1, \dots, \xi_D) \in [-1, 1]^D$, which are the indeterminates of our polynomials. Let \mathcal{P}_d be the vector space of D -variate polynomials of degree $\leq d$:

$$\mathcal{P}_d = \left\{ \sum_{k_1 + \dots + k_D \leq d} c_{k_1 \dots k_D} \xi_1^{k_1} \dots \xi_D^{k_D}, \quad c_{k_1 \dots k_D} \in \mathbb{R} \right\}. \quad (\text{A1})$$

This vector space has dimension:

$$n(d) := \dim \mathcal{P}_d = \sum_{k_1 + \dots + k_D \leq d} 1 \quad (\text{A2})$$

$$= \frac{1}{2}(d+1)(d+2) \quad \text{for } D = 2 \quad (\text{A3})$$

$$= \frac{1}{6}(d+1)(d+2)(d+3) \quad \text{for } D = 3. \quad (\text{A4})$$

Let \mathcal{V}_d be the space of polynomial vector fields of degree $\leq d$:

$$\mathcal{V}_d = \left\{ \begin{bmatrix} Q_1 \\ \vdots \\ Q_D \end{bmatrix}, \quad (Q_1, \dots, Q_D) \in \mathcal{P}_d^D \right\}, \quad (\text{A5})$$

and it follows that

$$\dim \mathcal{V}_d = D \dim \mathcal{P}_d = Dn(d). \quad (\text{A6})$$

We are interested in $\mathcal{V}_d^{\text{div}}$, the vector space of divergence-free polynomial vectors:

$$\mathcal{V}_d^{\text{div}} = \left\{ \mathbf{V} \in \mathcal{V}_d, \quad \sum_{\alpha=1}^D \frac{\partial V_\alpha}{\partial \xi_\alpha} = 0 \right\}. \quad (\text{A7})$$

We can obtain the dimension of $\mathcal{V}_d^{\text{div}}$ from the rank-nullity theorem. Consider the divergence as a linear application δ :

$$\delta : \mathcal{V}_d \rightarrow \mathcal{P}_{d-1} \quad (\text{A8})$$

$$\mathbf{V} \mapsto \nabla \cdot \mathbf{V}. \quad (\text{A9})$$

By definition of $\mathcal{V}_d^{\text{div}}$, $\mathcal{V}_d^{\text{div}} = \text{Ker } \delta$. Furthermore, for any $\mathbf{V} \in \mathcal{V}_d$, $\delta(\mathbf{V})$ is a scalar polynomial of degree $d-1$, so $\text{Im } \delta \subset \mathcal{P}_{d-1}$. But the reverse inclusion also holds, since for any scalar polynomial $Q \in \mathcal{P}_{d-1}$, we can construct $\mathbf{V} \in \mathcal{V}_d$ such that $\delta(\mathbf{V}) = Q$, for example:

$$\mathbf{V} = \begin{bmatrix} \int Q d\xi_1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \in \mathcal{V}_d, \quad (\text{A10})$$

from which we conclude that $\text{Im } \delta = \mathcal{P}_{d-1}$. Then, the rank-nullity theorem applied to δ states:

$$\dim \mathcal{V}_d = \dim \text{Im } \delta + \dim \text{Ker } \delta \quad (\text{A11})$$

$$= \dim \mathcal{P}_{d-1} + \dim \mathcal{V}_d^{\text{div}}, \quad (\text{A12})$$

from which we find:

$$N_{\text{div}} := \dim \mathcal{V}_d^{\text{div}} = Dn(d) - n(d-1). \quad (\text{A13})$$

d	N_{div}				
	0	1	2	3	4
D					
2	2	5	9	14	20
3	3	11	26	50	85

Table A1. Number N_{div} of divergence-free basis functions, corresponding to the number of degrees of freedom for the magnetic field, as a function of the number of spatial dimensions D and polynomial degree d .

A2 Generation of basis functions

Table A1 shows the number of divergence-free basis functions for $D \in \{2, 3\}$ and some values of the degree d . The number of basis functions grows rapidly with d , and determining them manually quickly becomes error prone, if not intractable. In addition, we not only need the functions ϕ_l but also their D spatial derivatives and the diagonal of the mass matrix. We therefore resort to symbolic computation using the Python package SymPy (Meurer et al. 2017) to compute all the required symbolic expressions and generate corresponding C code.

We adopt a simple, brute-force approach to basis function generation, which we now illustrate in 3D for clarity. We start by choosing a sequence $\{Q_i\}$ of $n(d+1)$ polynomials which form a basis of \mathcal{P}_{d+1} . At this stage, we can choose the Legendre polynomials, or simply the canonical basis of monomials. We then generate a pool of vectors of \mathcal{V}_{d+1} of the form:

$$\left\{ \begin{bmatrix} Q \\ 0 \\ 0 \end{bmatrix}, Q \in \{Q_i\} \right\} \cup \left\{ \begin{bmatrix} 0 \\ Q \\ 0 \end{bmatrix}, Q \in \{Q_i\} \right\} \cup \left\{ \begin{bmatrix} 0 \\ 0 \\ Q \end{bmatrix}, Q \in \{Q_i\} \right\}. \quad (\text{A14})$$

For each vector \mathbf{V} of this pool, $\nabla \times \mathbf{V}$ is a vector of $\mathcal{V}_d^{\text{div}}$ since the curl is divergence-free, and at least one degree less than \mathbf{V} . We discard the \mathbf{V} whose curl vanishes, and sort the remaining $\nabla \times \mathbf{V}$ by increasing total degree, and optionally some other arbitrary choices such as ordering of degree on vector components.

Finally, we start from the first (lowest-degree) vector polynomial in the resulting sequence, and perform a Gram-Schmidt orthogonalization of the sequence vectors with respect to the dot product:

$$\langle \mathbf{V}, \mathbf{W} \rangle = \int_{[-1,1]^3} (V_1 W_1 + V_2 W_2 + V_3 W_3) \, d\xi, \quad (\text{A15})$$

which is exactly the dot product defined at (10) restricted to the components of the magnetic field. Once we have collected N_{div} mutually orthogonalized non-null vectors, the process is complete and we have obtained an orthogonal basis of $\mathcal{V}_d^{\text{div}}$, sorted by increasing total degree.

APPENDIX B: PROJECTION EFFECTS WITH LDF BASES

In this section, we illustrate that care should be taken when L^2 -projecting non-divergence-free functions onto LDF basis vectors. Such a situation could arise for example with our Powell scheme, when projecting limited solutions back onto DG weights, as discussed in Section 4.2.4.

The main issue with these projections is that lower-degree contributions can contaminate higher-degree terms. Table B1 illustrates this effect with a simple first-degree 2D vector field $\mathbf{u} = (ax, by)$ by showing the projections $\int_K \mathbf{u} \cdot \phi_l$ for each 2D LDF basis function

Degree	LDF basis index l	Projection
0	0	0
	1	0
1	2	0
	3	0
	4	$\frac{4}{3}(a-b)$
2	5	0
	6	0
	7	0
	8	0
3	9	0
	10	0
	11	$\frac{152}{249}(a+b)$
	12	$-\frac{2}{3}(a+b)$
	13	0
4	14	0
	15	0
	16	0
	17	0
	18	0
	19	0
	5	20
21		0
22		$\frac{1268}{28629}(a+b)$
23		$-\frac{2}{45}(a+b)$
24		$-\frac{220}{121263}(a+b)$
25		0
26		0

Table B1. Contamination of higher-order coefficients resulting from the L^2 projection of $\mathbf{u} = (ax, by)$ onto a locally divergence-free basis.

ϕ_l . Note that $\nabla \cdot \mathbf{u} = a + b$, and even though \mathbf{u} has only degree 1, the projection can impact arbitrarily high degrees of the LDF basis whenever $a + b \neq 0$. This effect is due to the LDF basis not spanning the whole space of polynomial vector fields, and can be seen as a special form of polynomial aliasing with LDF bases.

APPENDIX C: DIVERGENCE

Different prescriptions to estimate the global divergence of the magnetic field have been proposed in the context of MHD or DG schemes (see e.g. Cockburn et al. 2004; Winters & Gassner 2016). In this work, we choose to define the divergence in a cell K of volume V as the cell-averaged quantity:

$$(\nabla \cdot \mathbf{B})_K := \frac{1}{V} \int_{\partial K} \mathbf{B}_F \cdot \mathbf{n}, \quad (\text{C1})$$

where the integral runs over all faces, and \mathbf{B}_F is the value of the magnetic field at the face, whose normal component B_F^n is chosen as in 4.1.3.2 with the HLLD solver (or returned by the Riemann solver, if using an 8-wave solver which provides an interface state).

This definition arises from averaging $\nabla \cdot \mathbf{B}$ over the cell volume, applying the divergence theorem, and setting $\mathbf{B} := \mathbf{B}_F$ at the face. It adopts the finite volume formulation of DG, while being sensitive to the choice of normal component of \mathbf{B} which is seen by the Riemann solver.

This paper has been typeset from a $\text{\TeX}/\text{\LaTeX}$ file prepared by the author.