Journal of Computational Physics 228 (2009) 641–660

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Splitting based finite volume schemes for ideal MHD equations

F.G. Fuchs, S. Mishra, N.H. Risebro *

Centre of Mathematics for Applications (CMA), University of Oslo, P.O. Box 1053, Blindern, N-0316 Oslo, Norway

ARTICLE INFO

Article history:
Received 25 April 2008
Received in revised form 13 August 2008
Accepted 19 September 2008
Available online 17 October 2008

Keywords:
Conservation laws
Induction equation
Divergence constraint
Upwinded source terms
MHD

ABSTRACT

We design finite volume schemes for the equations of ideal magnetohydrodynamics (MHD) and based on splitting these equations into a fluid part and a magnetic induction part. The fluid part leads to an extended Euler system with magnetic forces as source terms. This set of equations are approximated by suitable two- and three-wave HLL solvers. The magnetic part is modeled by the magnetic induction equations which are approximated using stable upwind schemes devised in a recent paper [F. Fuchs, K.H. Karlsen, S. Mishra, N.H. Risebro, Stable upwind schemes for the Magnetic Induction equation. Math. Model. Num. Anal., Available on conservation laws preprint server, submitted for publication, URL: <http://www.math.ntnu.no/conservation/2007/029.html>]. These two sets of schemes can be combined either component by component, or by using an operator splitting procedure to obtain a finite volume scheme for the MHD equations. The resulting schemes are simple to design and implement. These schemes are compared with existing HLL type and Roe type schemes for MHD equations in a series of numerical experiments. These tests reveal that the proposed schemes are robust and have a greater numerical resolution than HLL type solvers, particularly in several space dimensions. In fact, the numerical resolution is comparable to that of the Roe scheme on most test problems with the computational cost being at the level of a HLL type solver. Furthermore, the schemes are remarkably stable even at very fine mesh resolutions and handle the divergence constraint efficiently with low divergence errors.

1. Introduction

Many interesting problems in astrophysics, solar physics, electrical engineering and aerospace engineering are based on modeling the evolution of plasmas. Most models involve the equations of magneto-hydro dynamics (MHD) which read

$$\begin{aligned}
&\rho_t + \operatorname{div}(\rho\mathbf{u}) &&= 0,\\
&(\rho\mathbf{u})_t + \operatorname{div}\left(\rho\mathbf{u}\otimes\mathbf{u} + \left(p+\tfrac{1}{2}\mathbf{B}^2\right)\mathcal{I} - \mathbf{B}\otimes\mathbf{B}\right) &&= 0,\\
&E_t + \operatorname{div}\left(\left(E+p+\tfrac{1}{2}\mathbf{B}^2\right)\mathbf{u} - (\mathbf{u}\cdot\mathbf{B})\mathbf{B}\right) &&= 0,\\
&\mathbf{B}_t + \operatorname{div}(\mathbf{u}\otimes\mathbf{B} - \mathbf{B}\otimes\mathbf{u}) &&= 0,\\
&\operatorname{div}\mathbf{B} &&= 0,
\end{aligned} \tag{1.1}$$

where ρ denotes the density, $\mathbf{u} = \{u^1, u^2, u^3\}$ and $\mathbf{B} = \{B^1, B^2, B^3\}$ denote the velocity and the magnetic fields respectively, p the pressure and E the total energy of the plasma. The variables are related by the following ideal gas equation of state

* Corresponding author.
E-mail addresses: franzf@math.uio.no (F.G. Fuchs), siddharm@cma.uio.no (S. Mishra), nilshr@math.uio.no (N.H. Risebro).
URL: http://folk.uio.no/franzf/ (F.G. Fuchs).

0021-9991/$ - see front matter © 2008 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2008.09.027

$$E = \frac{p}{\gamma - 1} + \frac{1}{2}\rho|\mathbf{u}|^2 + \frac{1}{2}|\mathbf{B}|^2, \tag{1.2}$$

with γ being the gas constant. The above system is a system of conservation laws (with a constraint) in three dimensions on the following form

$$V_t + f(V)_x + g(V)_y + h(V)_z = 0,$$
$$\mathrm{div}\mathbf{B} = 0,$$

with $V = (\rho, \rho u^1, \rho u^2, \rho u^3, E, B^1, B^2, B^3)$ being the vector of conserved variables and f, g and h are the directional fluxes in the x, y and z directions respectively. The constraint that the divergence field should be solenoidal is a consequence of the fact that magnetic monopoles have not been observed in nature. A complete derivation of the MHD equations along with a description of the hypotheses on its validity is presented in [20]. We summarize the basic steps in the derivation below.

1.1. Derivation of the model

We non-dimensionalize all the quantities following [20] and choose units suitably so that the permeability of the medium is set to 1. Then the Eq. (1.1) are derived from the following set of physical laws,

Conservation of mass: The conservation of mass takes the following differential form (mass conservation for plasmas being the same as that of fluids),

$$\rho_t + \mathrm{div}(\rho\mathbf{u}) = 0. \tag{1.3}$$

Faraday's law: By using Maxwell's equations, the Stokes theorem and the fact that the electric field in a co-moving frame is zero, Faraday's law for the magnetic flux across a surface $\mathbf{S}$ bounded by a curve $\delta\mathbf{S}$ is given by

$$-\frac{d}{dt}\int_S \mathbf{B}\cdot\mathbf{dS} = \int_{\delta S} E\cdot dl$$

becomes

$$\mathbf{B}_t + \mathrm{div}(\mathbf{u}\otimes\mathbf{B} - \mathbf{B}\otimes\mathbf{u}) = -\mathbf{u}(\mathrm{div}\mathbf{B}). \tag{1.4}$$

The right hand side is proportional to div$\mathbf{B}$ and is often omitted due to the divergence constraint. Eq. (1.4) is referred to as the magnetic induction equation and can be considered as the special form of Maxwell equations governing the evolution of magnetic fields due to the action of a given velocity field $\mathbf{u}$.

Conservation of momentum: In differential form, the conservation of momentum for a plasma is given by,

$$(\rho\mathbf{u})_t + \mathrm{div}(\rho\mathbf{u}\otimes\mathbf{u} + p\mathcal{I}) = \mathbf{J}\times\mathbf{B}, \tag{1.5}$$

where $\mathbf{J}$ denotes the current density and $\mathcal{I}$ the 3×3 identity matrix. The above equation results from the fact that the momentum of the plasma changes due to the pressure and action of the Lorentz force $\mathbf{J}\times\mathbf{B}$ exerted by the magnetic field. Under the assumptions of ideal MHD, Ampere's law expresses the current density as

$$\mathbf{J} = \mathrm{curl}(\mathbf{B}). \tag{1.6}$$

Use of standard vector identities results in the following "semi-conservative" form,

$$(\rho\mathbf{u})_t + \mathrm{div}\left(\rho\mathbf{u}\otimes\mathbf{u} + \left(p + \frac{1}{2}\mathbf{B}^2\right)\mathcal{I} - \mathbf{B}\otimes\mathbf{B}\right) = -\mathbf{B}(\mathrm{div}\mathbf{B}). \tag{1.7}$$

Due to the divergence constraint, one usually neglects the right hand side of the above equation to get the momentum conservation in (1.1).

Conservation of energy: Defining the hydrodynamic energy of an ideal gas as

$$E^{\mathrm{hd}} = \frac{p}{\gamma - 1} + \frac{1}{2}\rho\mathbf{u}^2,$$

and using the conservation of this energy results in

$$E^{\mathrm{hd}}_t + \mathrm{div}((E^{\mathrm{hd}} + p)\mathbf{u}) = \mathbf{J}\cdot(\mathbf{B}\times\mathbf{u}). \tag{1.8}$$

The right hand side represents the change in energy due to the magnetic field. Using standard vector identities and Ampere's law, we obtain

$$\mathbf{J}\cdot(\mathbf{B}\times\mathbf{u}) = \left(\mathbf{B}\cdot\frac{\partial\mathbf{B}}{\partial t} - (\mathbf{u}\cdot\mathbf{B})(\mathrm{div}\mathbf{B}) - \mathrm{div}(\mathbf{B}\cdot\mathbf{B})\mathbf{u} - (\mathbf{u}\cdot\mathbf{B})\mathbf{B}\right).$$

Defining the total energy of the plasma as $E = E^{\mathrm{hd}} + \frac{1}{2}\mathbf{B}^2$, energy conservation takes the form

$$E_t + \mathrm{div}\left(\left(E + p + \frac{1}{2}\mathbf{B}^2\right)\mathbf{u} - (\mathbf{u}\cdot\mathbf{B})\mathbf{B}\right) = -(\mathbf{u}\cdot\mathbf{B})(\mathrm{div}\mathbf{B}). \tag{1.9}$$

Using the divergence constraint results in the energy equation in (1.1). Combining all the above we get

$$\begin{aligned}
&\rho_t + \mathrm{div}(\rho\mathbf{u}) && = 0,\\
&(\rho\mathbf{u})_t + \mathrm{div}(\rho\mathbf{u}\otimes\mathbf{u} + (p + \tfrac{1}{2}|\mathbf{B}|^2)I - \mathbf{B}\otimes\mathbf{B}) && = -c_1\mathbf{B}(\mathrm{div}\mathbf{B}),\\
&E_t + \mathrm{div}\left(\left(E + p + \tfrac{1}{2}|\mathbf{B}|^2\right)\mathbf{u} - (\mathbf{u}\cdot\mathbf{B})\mathbf{B}\right) && = -c_1(\mathbf{u}\cdot\mathbf{B})(\mathrm{div}\mathbf{B}),\\
&\mathbf{B}_t + \mathrm{div}(\mathbf{u}\otimes\mathbf{B} - \mathbf{B}\otimes\mathbf{u}) && = -c_2\mathbf{u}(\mathrm{div}\mathbf{B}),
\end{aligned} \tag{1.10}$$

with constants $c_1 = c_2 = 1$. Note that this form of the equations is symmetrizable by the results of [11]. An explicit inclusion of the divergence constraint means taking $c_1 = c_2 = 0$, and this is the standard form of the MHD Eq. (1.1).

We remark that MHD equations can be thought of as a combination of fluid dynamics coupled with magnetic fields. In fact, the "physical" form of these equations are a combination of the fluid Eqs.(1.3), (1.5) and (1.8) (with magnetic forces acting on them) and the magnetic induction Eq. (1.4). Collecting these equations, we can obtain the "physical" form

$$\begin{aligned}
&\rho_t + \mathrm{div}(\rho\mathbf{u}) && = 0,\\
&(\rho\mathbf{u})_t + \mathrm{div}(\rho\mathbf{u}\times\mathbf{u} + p\mathcal{I}) && = \mathbf{J}\times\mathbf{B},\\
&E^{\mathrm{hd}}_t + \mathrm{div}((E^{\mathrm{hd}} + p)\mathbf{u}) && = \mathbf{J}\cdot(\mathbf{B}\times\mathbf{u}),\\
&\mathbf{B}_t + \mathrm{div}(\mathbf{u}\otimes\mathbf{B} - \mathbf{B}\otimes\mathbf{u}) && = -\mathbf{u}(\mathrm{div}\mathbf{B}).
\end{aligned} \tag{1.11}$$

This system is non-conservative. We will use this "physical" splitting of the equation into the fluid part and a magnetic part in order to design efficient finite volume schemes for (1.1).

As remarked earlier, the MHD equations are a system of conservation laws in three dimensions. A calculation of eigenvalues of the Jacobians (see [22] for details) shows that the equations are hyperbolic but not strictly hyperbolic. In particular, the fast, slow and Alfven waves coincide at the triple point, and this can lead to considerable difficulties in the analysis of the model. Furthermore, a naive scaling of eigenvectors leads to singularities, and the eigenvectors must be properly scaled. Well-defined eigensystems for MHD equations have been proposed in [22 and 2]. Non-linearities in the equations result in formation of discontinuities such as shock waves and contact discontinuities. The complex structure of the equations leads to intermediate shocks and compound shocks which are difficult to analyze and simulate. A detailed description of the analytical and numerical difficulties concerning MHD equations can be found in [23].

In the absence of analytical results, the main approach in dealing with these equations has to been to devise efficient numerical schemes to approximate their solutions. For a long time finite volume methods have been the preferred means to solve conservation laws numerically. These methods are based on approximating the integral form of the conservation law inside each cell or control volume. Numerical fluxes at each cell interface are based on either exact or approximate Riemann solvers. Higher order accuracy in space is obtained by using non-oscillatory ENO/WENO type piecewise polynomial functions. Time integration is performed by using high order stability preserving Runge-Kutta methods. A detailed account of finite volume schemes for conservation laws can be found in [15].

The crucial ingredient in the design and performance of any finite volume method is a suitable choice of the numerical flux function at cell interfaces. Usually, numerical fluxes are built either from exact or approximate solutions to Riemann problems at each cell interface. Exact solutions (even to Riemann problems) for the MHD equations are very complicated (and largely unavailable), and are therefore seldom used in numerical methods. Hence, approximate Riemann solvers for the MHD equations are widely used. These solvers are either of the Roe (linearized) or the HLL (non-linear) types. Roe solvers based on either a simple average of the Jacobians across each interface or the Roe average, see [21], can be used as approximate Riemann solvers. A Roe-average for the ideal MHD equations was developed in [6]. A special form of the Roe solver based on entropy variables, proposed and tested in [2], will be used in some numerical experiments in this paper.

The main problem with Roe solvers is that they can result in negative pressures and densities. Another issue is the high computational cost of these solvers. As a consequence, an attractive alternative has been to use HLL solvers, see [15]. These non-linear solvers are based on approximating the wave structure of the full Riemann problem by a simplified wave structure. The one-dimensional form of the MHD equations result in seven waves for each Riemann problem. HLL solvers approximate the solution by fewer waves. The two-wave solvers based on wave speeds suggested in [8] are the simplest to implement. They are provably positive (in the sense that the pressure and density in the solution are positive) and entropy stable, but are too dissipative in most test problems. Three-wave HLL solvers based on heuristic considerations have been developed in [16 and 12]. The solver of [12] is also provably positive. A positivity preserving five wave solver for MHD was developed in [18]. Recently, three-, five- and seven-wave approximate Riemann solvers have been designed in [4]. These solvers are proved to be positive. A thorough comparison of different HLL solvers has been reported in [17,25].

There is no clear choice for an ideal HLL solver. Some of the solvers are too dissipative but guarantee positive pressures and densities. Some are complicated to design and implement and can be costly in terms of computational resources, particularly in several space dimensions. The computational results indicate that the three-wave solver of [16] appears to be the least dissipative among HLL-three-wave solvers, whereas the solvers of [12 and 4] are provably positivity preserving. However, none of the HLL-three-wave solvers have the resolution of the Roe solver. Given the above factors, there is considerable

scope for designing a simple HLL type two- or three-wave solver for the MHD equations which are less dissipative and hence, have more resolution than the available HLL two- or three-wave solvers and a resolution comparable to the Roe solver. Our main aim in this paper is to design such a HLL type solver. We would also like to point out that recent papers [17,25] have extensively compared different HLL solvers and concluded that the five wave solver designed in [18] is very robust and has a resolution comparable to the Roe solver.

A key issue in the design of numerical schemes for the MHD equations in several space dimensions is how to handle the divergence constraint. A standard numerical scheme is not going to satisfy a discrete version of the divergence constraint exactly, and even for smooth solutions will give divergence errors controlled by the truncation error. When the solution contains shocks, the discretization can lead to large divergence errors which may result in negative densities and pressures. There has been wide interest in designing numerical schemes which enforce, or control the divergence constraint. Popular methods include using projection onto divergence free fields, see [5], this involves solving an elliptic equation at each time step. Another popular choice is to stagger the grids or design updates which preserve some discrete divergence. An incomplete list of references dealing with this approach includes [1,7,9,23,24,27] and references therein.

Another simple method to deal with the divergence constraint was proposed by Powell in [19]. This involves using the symmetrizable version of MHD Eq. (1.10) derived earlier this section with $c_1 = c_2 = 1$. Taking divergence of the magnetic field in (1.10) results in

$$(\mathrm{div}\mathbf{B})_t + \mathrm{div}(\mathbf{u}(\mathrm{div}\mathbf{B})) = 0. \tag{1.12}$$

Hence, any non-zero divergence introduced by a numerical discretization should be swept away from the domain by the velocity field by this approach, provided that the boundaries are absorbing. A detailed comparison of different methods for divergence cleaning is reported in [29].

The HLL or Roe solvers mentioned earlier, are not designed to handle the divergence constraint. In fact, most of them are based on a locally one dimensional form of the MHD equations which presupposes that the normal component of the magnetic field across an interface is constant. Although this assumption is valid in one dimension, one has to use some ad hoc procedure to extend the HLL solvers to multiple space dimensions. It would be desirable to design a solver that also handles the divergence constraint and can be extended to several space dimensions in a natural way. Another aim of this paper is to design a solver that addresses the divergence constraint.

The approximate Riemann solvers that we design in this paper are based on the "physical form" (1.11) of the MHD equations. This form suggests a natural splitting of the equations into a fluid and a magnetic part. The fluid part is the Euler equations of hydrodynamics along with Lorentz forces exerted on the fluid due to the magnetic field (1.5). Hence, we can use approximate Riemann solvers of the HLL type for the Euler equations, these are well known in the literature, see [26] for details. We will use both a standard HLL two-wave solver as well as a HLLC three-wave solver for the Euler equation part of (1.11). It is not enough to treat the hydrodynamic part only. The right hand side of the fluid equations in (1.11) involves magnetic forcing terms and these terms have to discretized suitably. Since the forcing terms involve derivatives, we must upwind these derivatives. Furthermore, we need to enforce conservation of the variables. In order to do so, we will work with the first three equations of (1.1) (the conservative form) and treat the magnetic field as a coefficient in this extended Euler system and devise suitable HLL two-wave and three-wave solvers for this extended system.

The magnetic field is evolved by the magnetic induction Eq. (1.4). This is a linear system and has been studied extensively, see [3,27] and references therein. Despite being a linear system, (1.4) is not easy to deal with numerically. In a recent paper [10], we pointed out some of the difficulties involved in numerically approximating (1.4). We also developed a class of upwind schemes based on the form (1.4) without explicitly enforcing the divergence constraint and showed that the scheme converges and that the discrete divergence is bounded in L^2. In this paper we use the upwind schemes from [10] to approximate the magnetic part of (1.11).

We can also use the simple Lax-Friedrichs scheme to approximate (1.4). This scheme is very dissipative, but preserves a discrete version of the divergence constraint. Another class of schemes that can be used are the divergence preserving upwind schemes of [27]. When we implemented this scheme, we found that the resulting scheme for (1.1) was oscillatory.

Thus we emphasize that the main idea behind the schemes presented in this paper is to "split" the MHD equations into a fluid part and a magnetic part, and to use tailor made schemes for each part to devise a scheme for the full system. The choice of schemes for both the Euler part as well as the magnetic part, and the method to patch them together have to made judiciously in order to obtain a robust scheme for the MHD equations. The numerical resolution of these schemes is higher (in some cases, considerably higher) than that of HLL solvers and is comparable to the Roe solver. Furthermore, the splitting based schemes turn out to be remarkably stable at fine mesh resolutions. It is well known that computing at fine mesh resolutions results in lower numerical diffusions leading to instabilities. These instabilities can cause the pressure and density to be negative and the scheme crashes consequently. Most standard solvers for the MHD equations exhibit this behaviour. On the other hand, the splitting based schemes are quite stable and do not crash even for fine mesh resolutions. However, these schemes might still exhibit instabilities like the carbuncle phenomena which effect even very robust finite volume solvers for the Euler equations as well as MHD equations. Based on their stability as well as good resolution, simplicity of design and low computational cost, we believe that these schemes can be used as an alternative to both Roe solvers as well as HLL solvers in practical codes and do not require any extra divergence cleaning.

The rest of the paper is organized as follows: in Section 2, we present both the extended Euler solver as well as the upwind schemes for the induction equation and present the approximate Riemann solver for the MHD equations. These

schemes are compared with existing HLL and Roe solvers in a set of numerical experiments in both one and two space dimensions in Section 3 and we summarize the contents of this paper in Section 4.

2. Numerical schemes

In this section, we design finite volume schemes based on splitting the ideal MHD equations into its fluid part and magnetic part. Consider the ideal MHD Eq. (1.1) in the domain $\mathbf{D} = [X_L, X_R] \times [Y_L, Y_R] \times [Z_L, Z_R]$. For simplicity, we consider a uniform grid in space with mesh points given by $x_i = i\Delta x$, $y_j = j\Delta y$ and $z_k = k\Delta z$ where Δx, Δy and Δz are the mesh sizes in the x, y and z directions respectively. Let the Δt^n denote the time step at the n-th time level $t^n = \sum_{m<n} \Delta t^m$, be determined by a suitable CFL condition, and let the cell average at t^n be denoted by $U^n_{i,j,k}$.

With this notation, a general first order finite volume scheme reads

$$\begin{aligned} U^{n+1}_{i,j,k} &= V(U^n_{i-1,j,k}, U^n_{i,j-1,k}, U^n_{i,j,k-1}, U^n_{i,j,k}, U^n_{i+1,j,k}, U^n_{i,j+1,k}, U^n_{i,j,k+1}), \\ &= U^n_{i,j,k} - \tfrac{\Delta t^n}{\Delta x}(F(U^n_{i,j,k}, U^n_{i+1,j,k}) - F(U^n_{i-1,j,k}, U^n_{i,j,k})) - \tfrac{\Delta t^n}{\Delta y}(G(U^n_{i,j,k}, U^n_{i,j+1,k}) \\ &\quad - G(U^n_{i,j-1,k}, U^n_{i,j,k})) - \tfrac{\Delta t^n}{\Delta z}(H(U^n_{i,j,k}, U^n_{i,j,k+1}) - H(U^n_{i,j,k-1}, U^n_{i,j,k})), \end{aligned} \tag{2.1}$$

where F, G and H are numerical fluxes consistent with the directional fluxes f, g and h, respectively. The numerical fluxes will be determined by the splitting procedure outlined in the introduction.

2.1. Schemes for the extended Euler system

As mentioned before, we use the natural splitting of the MHD equations and divide the equations into a hydrodynamic and a magnetic part. We start with approximate Riemann solvers for the hydrodynamic part. This amounts to considering the mass, momentum and energy equations in (1.1), regarding the magnetic field as a known function that is constant in t. This results in the following extended Euler equations

$$\begin{aligned} &\rho_t + \operatorname{div}(\rho\mathbf{u}) && = 0, \\ &(\rho\mathbf{u})_t + \operatorname{div}\Big(\rho\mathbf{u}\otimes\mathbf{u} + \Big(p + \tfrac{1}{2}\mathbf{B}^2\Big)\mathcal{I} - \mathbf{B}\otimes\mathbf{B}\Big) && = 0, \\ &E_t + \operatorname{div}\Big(\Big(E + p + \tfrac{1}{2}\mathbf{B}^2\Big)\mathbf{u} - (\mathbf{u}\cdot\mathbf{B})\mathbf{B}\Big) && = 0, \end{aligned} \tag{2.2}$$

where the total energy E is given by the equation of state (1.2), and the magnetic field B is playing the role of a coefficient. This is a hyperbolic conservation law where the fluxes depend on the location through $\mathbf{B}$,

$$U^e_t + f^e(U^e, \mathbf{B})_x + g^e(U^e, \mathbf{B})_y + h^e(U^e, \mathbf{B})_z = 0, \tag{2.3}$$

with $U^e = \{\rho, \rho u^1, \rho u^2, \rho u^3, E\}$ being the conserved fluid variables and f^e, g^e, h^e are the directional fluxes in (2.2). Defining the vector of primitive variables $V^e = [\rho, u^1, u^2, u^3, p]$, we get the following quasilinear form the Eq. (2.2)

$$V^e_t + A^{1,e}(V^e, \mathbf{B})V^e_x + A^{2,e}(V^e, \mathbf{B})V^e_y + A^{3,e}(V^e, \mathbf{B})V^e_z = S^e(V, \mathbf{B}, D\mathbf{B}),$$

with $D\mathbf{B}$ denoting the matrix of partial derivatives of $\mathbf{B}$. Consider any unit vector $\mathbf{n} = (n^1, n^2, n^3)$ and take $u_{\mathbf{n}} := \mathbf{u}\cdot\mathbf{n}$, the directional Jacobian is given by

$$A^{1,e}n^1 + A^{2,e}n^2 + A^{3,e}n^3 = A^e\cdot\mathbf{n} = \begin{pmatrix} u_{\mathbf{n}} & \rho n^1 & \rho n^2 & \rho n^3 & 0 \\ 0 & u_{\mathbf{n}} & 0 & 0 & \frac{1}{\rho}n^1 \\ 0 & 0 & u_{\mathbf{n}} & 0 & \frac{1}{\rho}n^2 \\ 0 & 0 & 0 & u_{\mathbf{n}} & \frac{1}{\rho}n^3 \\ 0 & a_1 & a_2 & a_3 & u_{\mathbf{n}} \end{pmatrix},$$

where

$$\begin{aligned} a^1 &= \gamma n^1 p + (\gamma-1)(n^1(\mathbf{B}^2 - (B^1)^2) - B^1(n^2B^2 + n^3B^3)), \\ a^2 &= \gamma n^2 p + (\gamma-1)(n^2(\mathbf{B}^2 - (B^2)^2) - B^2(n^1B^1 + n^3B^3)), \\ a^3 &= \gamma n^3 p + (\gamma-1)(n^3(\mathbf{B}^2 - (B^3)^2) - B^3(n^1B^1 + n^2B^2)). \end{aligned}$$

The eigenvalues of the directional Jacobian are

$$\lambda^{1,5} = u_{\mathbf{n}} \mp \frac{1}{\sqrt{\rho}}\sqrt{\mathbf{a}\cdot\mathbf{n}}, \quad \text{and} \quad \lambda^{2,3,4} = u_{\mathbf{n}}.$$

The two extreme eigenvalues λ^1 and λ^5 correspond to acoustic waves with the sound speed being modified due to the presence of the magnetic field. The middle eigenvalues correspond to contact discontinuities and shear waves. If $\mathbf{B} = 0$, the system reduces to the Euler equations and the matrix is non strictly hyperbolic. For $\mathbf{B}\neq 0$, the above system is only weakly hyperbolic.

For notational simplicity, denote the extended sound speed in the x-direction as

$$a^{1,e} = \frac{1}{\sqrt{\rho}}\sqrt{\gamma p + (\gamma-1)((B^2)^2 + (B^3)^2)}.$$

Then the eigenvalues in the x-direction are given by

$$\lambda^{1,5} = u^1 \mp a^{1,e} \quad \text{and} \quad \lambda^{2,3,4} = u^1. \tag{2.4}$$

Our aim is to design approximate Riemann solvers of the HLL type for this extended system (2.2). We start with a simple two-wave solver.

2.1.1. HLL solver

We consider the extended Euler system (2.2) in the x-direction. The main feature of HLL solvers, see [15], is to approximate the true Riemann solution with a wave fan containing fewer waves, all of which are moving discontinuities. Thus for fixed time t, the approximate Riemann solution of a HLL solver is a piecewise constant function of x.

We start by defining the HLL solver which approximate the Riemann solution of (2.2) (in general containing 5 waves) with a wave fan containing only two waves. Let $U^e_{L,R}$, $\mathbf{B}_{L,R}$ denote the conserved fluid variables and magnetic field to the left and right of an interface. Then define the left and right fluxes by,

$$f^e_L = f^e\left(U^e_L, \frac{(\mathbf{B}_L + \mathbf{B}_R)}{2}\right), \quad f^e_R = f^e\left(U^e_R, \frac{(\mathbf{B}_L + \mathbf{B}_R)}{2}\right). \tag{2.5}$$

Note that we freeze the magnetic field locally in time and assume it to be constant across the Riemann fan. This essentially amounts to staggering the coefficient in (2.2). This strategy for dealing with coefficients in conservation laws was also used in [14]. Let s^e_L and s^e_R denote the left and right wave speeds and U^e_*, f^e_* the middle state and middle flux respectively. Conservation of U^e implies that

$$f^e_R - f^e_* = s^e_R(U^e_R - U^e_*), \quad f^e_* - f^e_L = s^e_L(U^e_* - U^e_L)$$

Solving the above equations results in

$$U^e_* = \frac{f^e_R - f^e_L - s^e_R U^e_R + s^e_L U^e_L}{s^e_L - s^e_R}, \quad f^e_* = \frac{s^e_R f^e_L - s^e_L f^e_R + s^e_L s^e_R (U^e_R - U^e_L)}{s^e_R - s^e_L}. \tag{2.6}$$

Once the wave speed are defined, the numerical flux across the interface can be written

$$F^{e,\mathrm{hll}} = F^e\left(U^e_L, U^e_R, \frac{(\mathbf{B}_L + \mathbf{B}_R)}{2}\right) = \begin{cases} f^e_L & \text{if} \quad s^e_L > 0 \\ f^e_* & \text{if} \quad s^e_L < 0 < s^e_R \\ f^e_R & \text{if} \quad s^e_R < 0 \end{cases}. \tag{2.7}$$

The wave speeds s^e_L and s^e_R are defined as in [8], let $A^{\hat{1},e}$ denote the arithmetic-average of the Jacobians $A^{1,e}_L$ and $A^{1,e}_R$, then the wave speeds are given by

$$\begin{aligned} s^e_L &= \min\{u^1_L - a^{1,e}_L, \hat{u^1} - a^{\hat{1},e}\} \\ s^e_R &= \max\{u^1_R + a^{1,e}_R, \hat{u^1} + a^{\hat{1},e}\} \end{aligned} \tag{2.8}$$

where $\hat{u^1}$ and $a^{\hat{1},e}$ are the normal velocity and the extended sound speed of the averaged Jacobian respectively. This solver is design to approximate only the outermost waves of the Riemann solution and will not approximate contact discontinuities or shear waves very well. The choice of the wave speeds by comparing with an averaged Jacobian is an attempt to replicate the strategy of [8] of using Roe-averages to increase resolution at isolated shocks. We were unable to obtain a Roe-matrix for the extended Euler system. Furthermore, the above choice of wave speeds does not imply that the scheme will resolve isolated fast magnetosonic shocks exactly due to the splitting but will hopefully approximate such discontinuities with good accuracy.

2.1.2. HLLC solver

HLL-three-wave solvers model the action of the two outer waves (as the HLL solver) as well as the contact/shear wave. Hence we must define three speeds $s^e_L \leqslant s^e_M \leqslant s^e_R$. The approximate three-wave solver and the corresponding numerical flux have the form

$$
\begin{aligned}
U^{e,\mathrm{hllc}} &= \begin{cases} U_L^e & \text{if} \quad s_L^e > 0, \\ U_L^{e,*} & \text{if} \quad s_L^e < 0 < s_M^e, \\ U_R^{e,*} & \text{if} \quad s_M^e < 0 < s_R^e, \\ U_R^e & \text{if} \quad s_R^e < 0, \end{cases} \\
F^{e,\mathrm{hllc}} &= F^{e,\mathrm{hllc}}\left(U_L^e, U_R^e, \tfrac{(\mathbf{B}_L+\mathbf{B}_R)}{2}\right) = \begin{cases} f_L^e & \text{if} \quad s_L^e > 0, \\ f_L^{e,*} & \text{if} \quad s_L^e < 0 < s_M^e, \\ f_R^{e,*} & \text{if} \quad s_M^e < 0 < s_R^e, \\ f_R^e & \text{if} \quad s_R^e < 0, \end{cases}
\end{aligned} \tag{2.9}
$$

Local conservation implies that

$$
\begin{aligned}
&s_L^e U_L^{e,*} - f_L^{e,*} = s_L^e U_L^e - f_L^e, \quad s_M^e U_R^{e,*} - f_R^{e,*} = s_M^e U_L^{e,*} - f_L^{e,*} \\
&\text{and} \quad s_R^e U_R^e - f_R^e = s_R^e U_R^{e,*} - f_R^{e,*}.
\end{aligned} \tag{2.10}
$$

Solving the above equations, we obtain the following expressions for $f_L^{e,*}$ and $f_R^{e,*}$

$$
f_L^{e,*} = f_*^e - \frac{s_L^e(s_R^e - s_M^e)}{s_R^e - s_L^e}\Delta U^{e,*}, \quad f_R^{e,*} = f_*^e + \frac{s_R^e(s_M^e - s_L^e)}{s_R^e - s_L^e}\Delta U^{e,*}, \tag{2.11}
$$

where $\Delta U^{e,*} = U_R^{e,*} - U_L^{e,*}$ and f_*^e is the middle flux of the HLL solver (2.6).

We follow [16] and determine the two middle states by the following simple observation

$$
\Delta U^{e,*} = \alpha(U_R^e - U_L^e),
$$

where α is a constant to be determined, thus the jump across the middle wave is proportional to the difference between the left and right states. The case of $\alpha = 0$ corresponds gives the *HLL* solver. When $\alpha = 1$, the solver gives an isolated single discontinuity. We choose s_L^e and s_R^e as the Einfeldt speeds given in (2.8). Since the middle wave is supposed to model the contact discontinuity, s_M^e is chosen as the corresponding velocity of the averaged Jacobian $\widehat{A^{1,e}}$ i.e., $s_M^e = \hat{u}^1$. Let $c_* = |a_L^{1,e} - s_M^e|$ and

$$
s = \frac{\|\Delta f^e - s_M^e \Delta U^e\|}{\|\Delta U^e\|},
$$

where $\Delta U^e = U_R^e - U_L^e$, $\Delta f^e = f_R^e - f_L^e$ and $\|C\| = \sum_k |C^k|$. The factor α is then defined by

$$
\alpha = \max\left\{0, 1 - \frac{s}{c_*}\right\}
$$

This choice of α is motivated by the following argument. Consider the quantity

$$
\|\Delta f^e - s_M^e \Delta U^e\|.
$$

At an isolated contact discontinuity, the above is equal to zero. Thus $\alpha = 1$ should represent this situation adequately. Similarly, if an isolated shock is present, then α should be equal to zero and

$$
\|\Delta f^e - s_M^e \Delta U^e\| = c_* \|\Delta U^e\|.
$$

This implies that the above linear interpolation gives the correct values of α in this regime.

It is straightforward to extend the HLL and the HLLC solvers to three dimensions. Let $F^{e,\mathrm{hllc}}$, $G^{e,\mathrm{hllc}}$ and $H^{e,\mathrm{hllc}}$ be HLLC fluxes consistent with f^e, g^e and h^e and defined by (2.11) with the obvious modifications in the y and z directions. Let the discrete time derivative for any function b be denoted

$$
D_t^+ b^n = \frac{b^{n+1} - b^n}{\Delta t},
$$

and the average of two states across an interface as

$$
\bar{b}_{l+1/2} = \frac{b_l + b_{l+1}}{2}.
$$

Similarly we define the discrete spatial derivatives as

$$
\mathcal{D}_x^{\pm} b_{i,j,k} = \pm \frac{b_{i\pm1,j,k} - b_{i,j,k}}{\Delta x},
$$

with analogous definitions of $D_y^{\pm}$ and $D_z^{\pm}$. Then the HLLC scheme in three dimensions reads

$$
D_t^+ U_{i,j,k}^{e,n} = -D_x^- F^{e,\mathrm{hllc}}(U_{i,j,k}^{e,n}, U_{i+1,j,k}^{e,n}, \overline{\mathbf{B}}_{i+1/2,j,k}) - D_y^- G^{e,\mathrm{hllc}}(U_{i,j,k}^{e,n}, U_{i,j+1,k}^{e,n}, \overline{\mathbf{B}}_{i,j+1/2,k}) - D_z^- H^{e,\mathrm{hllc}}(U_{i,j,k}^{e,n}, U_{i,j,k+1}^{e,n}, \overline{\mathbf{B}}_{i,j,k+1/2}) \tag{2.12}
$$

The HLL solver is extended to three space dimensions analogously.

As a shorthand notation we write

$$U^{e,n+1}_{i,j,k} = V^{e,\mathrm{hllc}}(U^{e,n}_{\dots}, \mathbf{B}^n_{\dots}, \Delta t^n),$$

to indicate a single application of the HLLC scheme with a timestep Δt^n.

Remark. The usual approach in designing a HLLC solver for Euler equations (see [26]) uses the fact that the pressure and normal velocities are constant across the middle wave and that the tangential velocities only jump across the middle wave. Plugging these assumptions into (2.10) results in a slightly different version of the HLLC solver for the extended Euler system (2.2), with formulas similar to those obtained in [12] for the MHD equations. We have found that the numerical results with this solver are slightly more dissipative than those obtained with the solver described here.

2.2. Schemes for the induction equation

The next step in this splitting procedure is to devise an efficient scheme for the induction Eq. (1.4). Rewriting the induction equation using the divergence constraint we obtain,

$$\partial_t \mathbf{B} + \mathrm{curl}(\mathbf{B} \times \mathbf{u}) = 0, \tag{2.13}$$

and this implies

$$\partial_t(\mathrm{div}\mathbf{B}) = 0. \tag{2.14}$$

Thus if the divergence is initially zero, it remains so and satisfies the divergence constraint in (1.1). The above equation is a linear hyperbolic system but not strictly hyperbolic. In general (see [10]), it is not possible to symmetrize (2.13) without explicitly using the divergence constraint. One way to symmetric these equations is to write them in the Godunov-Powell form (1.4) mentioned in the introduction. Following [10] and using standard vector identities, we can obtain the following symmetrized form of the system (2.13),

$$\partial_t \mathbf{B} + u^1 \partial_x(\mathbf{B}) + u^2 \partial_y(\mathbf{B}) + u^3 \partial_z(\mathbf{B}) = (\mathbf{B} \cdot \nabla)\mathbf{u} - \mathbf{B}(\mathrm{div}\mathbf{u}). \tag{2.15}$$

For a given smooth velocity field $\mathbf{u}$, it is easy to prove that weak solutions to the above equation exist, are unique and satisfy an energy estimate, see [10] for the relevant references. It seems essential to rewrite (2.13) in the symmetric form (2.15) in order to obtain well-possedness. Furthermore, all the above forms are equivalent if the initial divergence is zero and the solutions are sufficiently smooth.

Our aim in this section is to present efficient numerical schemes for the induction equation. The schemes should be stable in a suitable sense and not introduce too much divergence errors. Furthermore, it is obviously important that they should work effectively with the HLL Euler solvers of the last section to yield a robust scheme for the MHD equations. To simplify the presentation, we consider the induction equation in two dimensions,

$$\begin{aligned} &\partial_t B^1 + \partial_y(u^1 B^2 - u^2 B^1) = 0, \\ &\partial_t B^2 - \partial_x(u^1 B^2 - u^2 B^1) = 0. \end{aligned} \tag{2.16}$$

Consider a given velocity field $\mathbf{u}$ and a uniform discretization in both the x and y directions with mesh sizes Δx and Δy. We start with the following notation, for a continuous function $a(x, y)$, we set

$$a_{ij} = a(x_i, y_j), \quad i \text{ and } j \in \mathbb{Z}/2,$$

and define and the central differences

$$D^0_{x,y} = \frac{1}{2}(D^+_{x,y} + D^-_{x,y}).$$

Furthermore, set $[a]^+ = \max\{a, 0\}$ and $[a]^- = \min\{a, 0\}$.

2.2.1. The Lax-Friedrichs scheme

The simplest scheme for the hyperbolic system (2.16) is the Lax-Friedrichs scheme. Set $\phi^n_{ij} = B^{1,n}_{ij} u^{2,n}_{ij} - B^{2,n}_{ij} u^{1,n}_{ij}$ and define this scheme as

$$\begin{aligned} D^+_t B^{1,n}_{ij} &= \tfrac{1}{\Delta t}\left[\left(\tfrac{\Delta x}{2}\right)^2 D^+_x D^-_x(B^{1,n}_{ij}) + \left(\tfrac{\Delta y}{2}\right)^2 D^+_y D^-_y(B^{1,n}_{ij})\right] + D^0_y(\phi^n_{ij}), \\ D^+_t B^{2,n}_{ij} &= \tfrac{1}{\Delta t}\left[\left(\tfrac{\Delta x}{2}\right)^2 D^+_x D^-_x(B^{2,n}_{ij}) + \left(\tfrac{\Delta y}{2}\right)^2 D^+_y D^-_y\left(B^{2,n}_{ij}\right)\right] - D^0_x(\phi^n_{ij}). \end{aligned} \tag{2.17}$$

Note that this scheme is based on using a second central difference for the flux coupled with a artificial diffusion term. In general, this scheme is too dissipative for practical applications, but if the central discrete divergence is constant, the scheme preserves this central divergence. To be concrete, let the central discrete divergence be defined as

$$d^n_{ij} = \mathrm{div}^0(\mathbf{B}^n_{ij}) = D^0_x(B^{1,n}_{ij}) + D^0_y(B^{2,n}_{ij}). \tag{2.18}$$

Under the Lax-Friedrichs scheme, the evolution of $d^n_{i,j}$ reads

$$d^{n+1}_{i,j} = d^n_{i,j} + \left(\frac{\Delta x}{2}\right)^2 D^+_x D^-_x(d^n_{i,j}) + \left(\frac{\Delta y}{2}\right)^2 D^+_y D^-_y(d^n_{i,j}).$$

Thus if $d^n_{i,j} = 0$, the above identity implies that $d^{n+1}_{i,j} \equiv 0$ and the scheme preserves divergence free fields.
As shorthand notation for a single application of the Lax-Friedrichs scheme we write $\mathbf{B}^{n+1}_{ij} = W^{m,\mathrm{LxF}}(\mathbf{u}^n_{\dots}, \mathbf{B}^n_{\dots}, \Delta t^n)$.

2.2.2. Divergence preserving upwind schemes

As mentioned in the introduction, there exists an extensive literature devoted to devising suitable upwind schemes which preserve some form of discrete divergence in (2.16). We present the divergence preserving scheme of Torrilhon and Fey [27] as an example. This scheme is given by

$$\begin{aligned} D^+_t B^{1,n}_{i,j} &= \tfrac{1}{2}(D^-_y(\varphi^1_{i,j} + \varphi^1_{i-1,j} + \varphi^2_{i,j} + \varphi^2_{i+1,j}) + D^+_y(\varphi^3_{i,j} + \varphi^3_{i+1,j} + \varphi^4_{i,j} + \varphi^4_{i-1,j})), \\ D^+_t B^{2,n}_{i,j} &= -\tfrac{1}{2}(D^-_x(\varphi^1_{i,j} + \varphi^1_{i,j-1} + \varphi^4_{i,j} + \varphi^4_{i,j+1}) + D^+_x\left(\varphi^2_{i,j} + \varphi^2_{i,j-1} + \varphi^3_{i,j} + \varphi^3_{i,j+1}\right)), \end{aligned} \tag{2.19}$$

where

$$\varphi^k_{i,j} = \omega^k_{i,j}(u^1_{i,j}B^{2,n}_{i,j} - u^2_{i,j}B^{1,n}_{i,j}),$$

and

$$\omega^k_{i,j} = \frac{[\mathbf{n}^k \cdot \mathbf{u}_{ij}]^+}{\sum_{k=1}^{4}[\mathbf{n}^k \cdot \mathbf{u}_{ij}]^+},$$

$\mathbf{n}^1 = (1,1),\ \mathbf{n}^2 = (-1,1),\ \mathbf{n}^3 = (-1,-1)$ and $\mathbf{n}^4 = (1,-1)$.

This scheme preserves the following discrete divergence,

$$\mathrm{div}^*(\mathbf{B}_{i,j}) = \frac{1}{4}(D^0_x(B^1_{i,j+1} + 2B^1_{i,j} + B^1_{i,j-1}) + D^0_y(B^2_{i+1,j} + 2B^2_{i,j} + B^1_{i-1,j})).$$

If $\mathbf{B}$ is smooth, div^* differs from div^0 by $\mathcal{O}(\Delta x^2 + \Delta y^2)$, with a constant depending on the second derivatives of $\mathbf{B}$. More details on this scheme can be found in [27,28]. We write this scheme as $\mathbf{B}^{n+1}_{ij} = W^{m,\mathrm{TF}}(\mathbf{u}^n_{\dots}, \mathbf{B}^n_{\dots}, \Delta t^n)$.

2.2.3. A stable upwind scheme

It is desirable to have a scheme which is stable in the L^2 norm, and which does not create spurious oscillations. The natural starting point to design a scheme with these properties is the Godunov-Powell form (1.4), and then use the symmetrized form (2.15). This strategy was adopted in [10 and 3]. Introduce the following notation,

$$D_x\{u,B\}_{i,j} = D^-_x([u_{i+1/2,j}]^+ B_{i,j}) + D^+_x([u_{i-1/2,j}]^- B_{i,j}) = [u_{j+1/2,j}]^- D^+_x B_{i,j} + [u_{i-1/2,j}]^+ D^-_x B_{i,j} + B_{ij} D^-_x u_{i+1/2,j}. \tag{2.20}$$

An analogous expression defines $D_y\{u,B\}$. If u and B are smooth functions, then

$$\partial_x(uB)(x_i,y_j) = D_x\{u,B\}_{i,j} + \mathcal{O}(\Delta x).$$

Then the stable upwind from [10] takes the form

$$\begin{aligned} D^+_t B^{1,n}_{i,j} &= (D_y\{u^{1,n}, B^{2,n}\}_{i,j} - D_y\{u^{2,n}, B^{1,n}\}_{i,j}) - [u^{1,n}_{i-1/2,j}]^+ D^-_x(B^{1,n}_{i,j}) - [u^{1,n}_{i+1/2,j}]^- D^+_x(B^{1,n}_{i,j}) \\ &\quad -[u^{1,n}_{i,j-1/2}]^+ D^-_y(B^{2,n}_{i,j}) - [u^{1,n}_{i,j+1/2}]^- D^+_y(B^{2,n}_{i,j}), \\ D^+_t B^{2,n}_{i,j} &= -(D_x\{u^{1,n}, B^{2,n}\}_{i,j} - D_x\{u^{2,n}, B^{1,n}\}_{i,j}) - [u^{2,n}_{i-1/2,j}]^+ D^-_x(B^{1,n}_{i,j}) - [u^{2,n}_{i+1/2,j}]^- D^+_x(B^{1,n}_{i,j}) \\ &\quad -[u^{2,n}_{i,j-1/2}]^+ D^-_y(B^{2,n}_{i,j}) - [u^{2,n}_{i,j+1/2}]^- D^+_y(B^{2,n}_{i,j}). \end{aligned} \tag{2.21}$$

Hence, the above scheme is based on upwinding the Godunov-Powell source term in (1.4). This scheme can also be written down in terms of the non-conservative symmetric form (2.15). Under the assumptions of sufficiently smooth velocity fields, it is shown to be stable in L^2 and hence converges to a weak solution of the magnetic induction equation. Furthermore, in the simple case of constant velocity fields, the above scheme is *TVD*. Numerical experiments indicating robustness of this scheme were presented in [10]. As mentioned before, the introduction of the Godunov-Powell source term results in divergence errors being transported by the velocity field, and one would like control these errors, but unfortunately we have not been able to prove that the divergence remains bounded in L^2. Nevertheless, numerical experiments indicate that the divergence generated by the scheme remains small.

Note also that this scheme requires evaluation of the velocities at the cell interfaces. We do this by averaging, i.e.,

$$u^n_{i+1/2,j} = \frac{1}{2}(u^n_{i+1,j} + u^n_{i,j}).$$

We write this scheme as $\mathbf{B}_{ij}^{n+1} = W^{m,\mathrm{SUS}}(\mathbf{u}_{\dots}^{n}, \mathbf{B}_{\dots}^{n}, \Delta t^{n})$.

2.2.4. A stable "non-conservative" upwind scheme

Since the SUS scheme of the last section did not lead to a rigorous divergence bound, the authors designed a slightly modified version of this scheme in [10] for which it was possible to establish L^2 bounds on the discrete divergence.

This upwind scheme is based on the non-conservative symmetric form (2.15) and hence we refer to it as the stable non-conservative upwind scheme (SUS-N). Define the auxiliary function σ_δ as an even smooth function such that

$$\sigma_\delta(a) = \begin{cases} \frac{\delta}{2}, & \text{if } |a| \leqslant \delta/2, \\ 0, & \text{if } |a| \geqslant \delta. \end{cases}$$

Furthermore, we demand that σ is non-increasing in the interval $[0, \delta]$ and that $|\sigma'_\delta(a)| < 2$ for all a. Then in two dimensions, the scheme reads

$$\begin{aligned} D_t^+ B_{ij}^{1,n} = & -[u_{ij}^{1,n}]^- D_x^+(B_{ij}^{1,n}) - [u_{ij}^{1,n}]^+ D_x^-(B_{ij}^{1,n}) - [u_{ij}^{2,n}]^- D_y^+(B_{ij}^{1,n}) - [u_{ij}^{2,n}]^+ D_y^-(B_{ij}^{1,n}) - D_y^0(u_{ij}^{2,n}) B_{ij}^{1,n} + D_y^0(u_{ij}^{1,n}) B_{ij}^{2,n} \\ & + \sigma_\delta(u_{ij}^{1,n}) \Delta x D_x^+ D_x^-(B_{ij}^{1,n}) + \sigma_\delta(u_{ij}^{2,n}) \Delta y D_y^+ D_y^-(B_{ij}^{1,n}), \\ D_t^+ B_{ij}^{2,n} = & -[u_{ij}^{1,n}]^- D_x^+\left(B_{ij}^{2,n}\right) - [u_{ij}^{1,n}]^+ D_x^-(B_{ij}^{2,n}) - [u_{ij}^{2,n}]^- D_y^+(B_{ij}^{2,n}) - [u_{ij}^{2,n}]^+ D_y^-(B_{ij}^{2,n}) + D_x^0(u_{ij}^{2,n}) B_{ij}^{1,n} - D_x^0(u_{ij}^{1,n}) B_{ij}^{2,n} \\ & + \sigma_\delta(u_{ij}^{1,n}) \Delta x D_x^+ D_x^-(B_{ij}^{2,n}) + \sigma_\delta(u_{ij}^{2,n}) \Delta y D_y^+ D_y^-(B_{ij}^{2,n}), \end{aligned} \tag{2.22}$$

Note that we add a small amount of explicit numerical diffusion at the points $\mathbf{u} = 0$ in addition to upwinding the derivatives. In [10], we were able to show that the approximate solutions generated by the above scheme (2.22) are bounded in L^2 and the standard central discrete divergence (2.18) is also bounded in L^2 under the assumptions that velocity field is sufficiently smooth. Furthermore, the approximate solutions are *TVD* when the velocity field is constant. We write an application of a single step with this scheme as $\mathbf{B}_{ij}^{n+1} = W^{m,\mathrm{SUS\text{-}N}}(\mathbf{u}_{\dots}^{n}, \mathbf{B}_{\dots}^{n}, \Delta t^{n})$.

Remark. It is straightforward extend all the above schemes to three space dimensions. For the sake of clarity we give the full three dimensional form of the SUS scheme. The extension of the other schemes to three dimensions is analogous.

$$\begin{aligned} D_t^+ B_{i,j,k}^{1,n} = & -D_y\{u_{i,j,k}^{2,n}, B_{i,j,k}^{1,n}\} + D_y\{u_{i,j,k}^{1,n}, B_{i,j,k}^{2,n}\} - D_z\{u_{i,j,k}^{3,n}, B_{i,j,k}^{1,n}\} + D_z\{u_{i,j,k}^{1,n}, B_{i,j,k}^{3,n}\} - ([u_{i-\frac{1}{2},j,k}^{1,n}]^+ D_x^-(B_{i,j,k}^{1,n}) + [u_{i+\frac{1}{2},j,k}^{1,n}]^- D_x^+(B_{i,j,k}^{1,n})) \\ & - ([u_{i,j-\frac{1}{2},k}^{1,n}]^+ D_y^-(B_{i,j,k}^{2,n}) + [u_{i,j+\frac{1}{2},k}^{1,n}]^- D_y^+(B_{i,j,k}^{2,n})) - ([u_{i,j,k-\frac{1}{2}}^{1,n}]^+ D_z^-(B_{i,j,k}^{3,n}) + [u_{i,j,k+\frac{1}{2}}^{1,n}]^- D_z^+(B_{i,j,k}^{3,n})), \\ D_t^+ B_{i,j,k}^{2,n} = & -D_x\{u_{i,j,k}^{1,n}, B_{i,j,k}^{2,n}\} + D_x\{u_{i,j,k}^{2,n}, B_{i,j,k}^{1,n}\} - D_z\{u_{i,j,k}^{3,n}, B_{i,j,k}^{2,n}\} + D_z\{u_{i,j,k}^{2,n}, B_{i,j,k}^{3,n}\} - ([u_{i-\frac{1}{2},j,k}^{2,n}]^+ D_x^-(B_{i,j,k}^{1,n}) + [u_{i+\frac{1}{2},j,k}^{2,n}]^- D_x^+(B_{i,j,k}^{1,n})) \\ & - ([u_{i,j-\frac{1}{2},k}^{2,n}]^+ D_y^-(B_{i,j,k}^{2,n}) + [u_{i,j+\frac{1}{2},k}^{2,n}]^- D_y^+(B_{i,j,k}^{2,n})) - ([u_{i,j,k-\frac{1}{2}}^{2,n}]^+ D_z^-(B_{i,j,k}^{3,n}) + [u_{i,j,k+\frac{1}{2}}^{2,n}]^- D_z^+(B_{i,j,k}^{3,n})), \\ D_t^+ B_{i,j,k}^{3,n} = & -D_x\{u_{i,j,k}^{1,n}, B_{i,j,k}^{3,n}\} + D_x\{u_{i,j,k}^{3,n}, B_{i,j,k}^{1,n}\} - D_y\{u_{i,j,k}^{2,n}, B_{i,j,k}^{3,n}\} + D_y\{u_{i,j,k}^{3,n}, B_{i,j,k}^{2,n}\} \\ & - ([u_{i-\frac{1}{2},j,k}^{3,n}]^+ D_x^-(B_{i,j,k}^{1,n}) + [u_{i+\frac{1}{2},j,k}^{3,n}]^- D_x^+(B_{i,j,k}^{1,n})) - ([u_{i,j-\frac{1}{2},k}^{3,n}]^+ D_y^-(B_{i,j,k}^{2,n}) + [u_{i,j+\frac{1}{2},k}^{3,n}]^- D_y^+(B_{i,j,k}^{2,n})) - ([u_{i,j,k-\frac{1}{2}}^{3,n}]^+ D_z^-(B_{i,j,k}^{3,n}) \\ & + [u_{i,j,k+\frac{1}{2}}^{3,n}]^- D_z^+(B_{i,j,k}^{3,n})). \end{aligned}$$

2.3. Combining the schemes

In order to obtain a scheme for the full MHD equations, we can now piece together the schemes for the hydrodynamic part with the schemes for the induction equation.

This can be done either simultaneously, resulting in update formula

$$\begin{aligned} U_{i,j,k}^{e,n+1} &= V^{e,\mathrm{type}_1}\left(U_{\dots}^{e,n}, \mathbf{B}_{\dots}^{n}, \Delta t^{n}\right), \\ \mathbf{B}_{i,j,k}^{n+1} &= W^{m,\mathrm{type}_2}(\mathbf{u}_{\dots}^{n}, \mathbf{B}_{\dots}^{n}, \Delta t^{n}), \end{aligned} \tag{2.23}$$

or sequentially, in which case we have the update formula

$$\begin{aligned} U_{i,j,k}^{e,n+1} &= V^{e,\mathrm{type}_1}\left(U_{\dots}^{e,n}, \mathbf{B}_{\dots}^{n}, \Delta t^{n}\right), \\ \mathbf{B}_{i,j,k}^{n+1} &= W^{m,\mathrm{type}_2}(\mathbf{u}_{\dots}^{n+1}, \mathbf{B}_{\dots}^{n}, \Delta t^{n}). \end{aligned} \tag{2.24}$$

Here "type$_1$" is either "HLL" or "HLLC", and "type$_2$" is one of "LxF", "TF", "SUS" or "SUS-N". Therefore all possible combinations give 16 possible update formulas. Furthermore, we can also reverse the order in the sequential update (2.24) by evolving the magnetic part first, followed by the evolution of the fluid part. We found no difference in the numerical results by reversing orders in (2.24).

Remark. Note that both the above schemes are consistent with the form (1.10) of the MHD equations with constants $c_1 = 0$ and $c_2 = 1$. This form is a special form of the equations which conserves mass, momentum and energy and has the Godunov-Powell source term only in the induction equations. This form is also formally equivalent to (1.1) for divergence free data.

Remark. The splitting schemes based on (2.24) are formally first order in both space and time. It is easy to increase the spatial accuracy by a standard ENO/WENO type non-oscillatory piecewise polynomial reconstruction. Second order formal accuracy in time can be accomplished by a Strang splitting, and further increased by using standard strong stability preserving Runge-Kutta methods.

3. Numerical examples

We are going to test the splitting schemes of the last section on a variety of numerical experiments in both one and two space dimensions. These experiments have been used in the literature on numerical schemes for MHD equations as benchmark test cases to compare different methods. The key issues are both accuracy and stability. Since the schemes of this paper are first order methods, we will compare them with three standard first order finite volume schemes for MHD equations: (1) a HLL two-wave solver (see [8]) for the full MHD equations which we will refer to as the HLL2 solver, (2) a HLL-three-wave solver for the MHD equations designed in [16], referred to HLL3 solver and (3) a Roe solver. The Roe solver that we use is based on adding symmetric diffusion in the entropy variables, in the spirit of [13] for Navier-Stokes equations and [2] for Euler equations. This Roe solver has better accuracy as well as stability compared to the standard Roe solvers used in the literature. More details about this Roe solver will be reported in a forthcoming paper. In terms of computational cost, the HLL2 solver is as costly as the HLL/SUS solver, the HLL3 solver has the same cost as the HLLC/SUS solver and the Roe solver is the most expensive. So, we will see whether the HLL/SUS and HLLC/SUS solvers do better than the HLL2 and HLL3 solvers respectively and how they compare with the Roe solver.

Remark. With a small choice of δ, e.g., $\delta = \min\{\Delta x, \Delta y\}/2$, we observed no difference between the SUS and the SUS-N solvers. Therefore we choose to report results using the SUS scheme only.

We concentrate on one- and two-dimensional examples, therefore we let the computational domain $[X_L, X_R] \times [Y_L, Y_R]$ be discretized uniformly in each direction leading to mesh sizes of Δx and Δy respectively. The time step is determined by the following CFL condition

$$\frac{\max_{ij}\{|u_{ij}^{1,n}| + a_{ij}^{1,e,n}, \quad |u_{ij}^{2,n}| + a_{ij}^{2,e,n}\}\Delta t^n}{\min\{\Delta x, \Delta y\}} \leqslant 1,$$

where $a_{ij}^{l,e,n}$ is the extended sound speed in the l-th direction in the cell I_{ij}. We run the numerical experiments at the same CFL-number of 0.45.

We use the following notation for errors, let W^M be a component of the numerical solution given by a scheme using a $M \times M$ grid, and W^R the same component of a reference solution. We define the relative error as

$$\Re E_M^p(W) = 100 \times \frac{\|W^M - W^R\|_p}{\|W^R\|_p},$$

where $\|\cdot\|_p$ is the discrete L^p norm. In this way we refer to e.g., $\Re E_{100}^1(\rho)$, denoting the relative difference in L^1 between the density calculated on a 100×100 grid and a reference density.

We have used $\gamma = 5/3$ and the simultaneous update formula (2.23) in all examples. We found that using the sequential update (2.24) gave very similar results.

3.1. Shock tube

We start with a one-dimensional shock tube test case. The initial conditions are given by

$$(\rho, \rho u_1, \rho u_2, \rho u_3, B^1, B^2, B^3, p) = \begin{cases} (1.0, +1.0, 0.0, 0.0, 0.7, 1.0, 0.0, 1.0), & \text{if } x < 0.5 \\ (0.3, -0.3, 0.0, 0.0, 0.7, 0.2, 0.0, 0.2), & \text{otherwise.} \end{cases} \tag{3.1}$$

The computational domain is $(x, t) \in [0, 1.5] \times [0, 0.5]$ with Neumann boundary conditions. All schemes use the CFL-number 0.45. The Riemann problem has a complicated solution containing shocks, rarefactions and contact discontinuities. We test different schemes on this problem and report the results in Fig. 1 and Table 1. We compute a reference solution with the standard HLL2 solver using 8000 mesh points. The results in Fig. 1 compare the HLLC/LxF, HLL/SUS and HLLC/SUS schemes with 400 mesh points on this problem. As shown in this figure, all three schemes resolve the complicated solution fairly well with the HLLC/SUS scheme being the least dissipative and the HLLC/LxF being the most dissipative. Note that HLLC/LxF is much less dissipative than the standard LxF scheme for a system, since the "Euler part" is approximated with a HLL solver. Table 1 gives a more quantitative comparison of the schemes. In this table, a "–" means that the scheme crashed due to negative density or pressure. We show relative errors in L^1 on different meshes for both the density and a component of the magnetic field and compare the HLL, HLL3, HLL/SUS, HLLC/SUS, HLLC/LxF, HLLC/TF and Roe schemes.

From the table, all the splitting based solvers yielded reasonably small errors for this problem. Surprisingly, the errors due to the HLLC/LxF scheme are not very high and this implies that the split scheme is much less dissipative than a standard

Lax-Friedrichs scheme would have been. In fact, this scheme has lower errors than the standard HLL scheme. As expected the HLLC/SUS scheme had lower errors than the HLL/SUS scheme and these errors were comparable to the HLLC scheme. The HLLC/TF had the best resolution among the splitting solvers in this case. Unfortunately, the approximate magnetic field was oscillatory. This is to be expected as in [10], we demonstrated via a numerical experiment that the TF scheme leads to oscillations around discontinuities and this might be the reason for the instabilities reported here.

Comparing the splitting schemes with the standard schemes, we see from the above table that the errors due to the HLL/SUS scheme and the HLLC/SUS scheme are comparable to those generated by the HLL scheme and HLL3 scheme respectively. For the density, the HLLC/SUS scheme has slightly greater resolution than the HLL3 scheme. Furthermore, none of the HLL

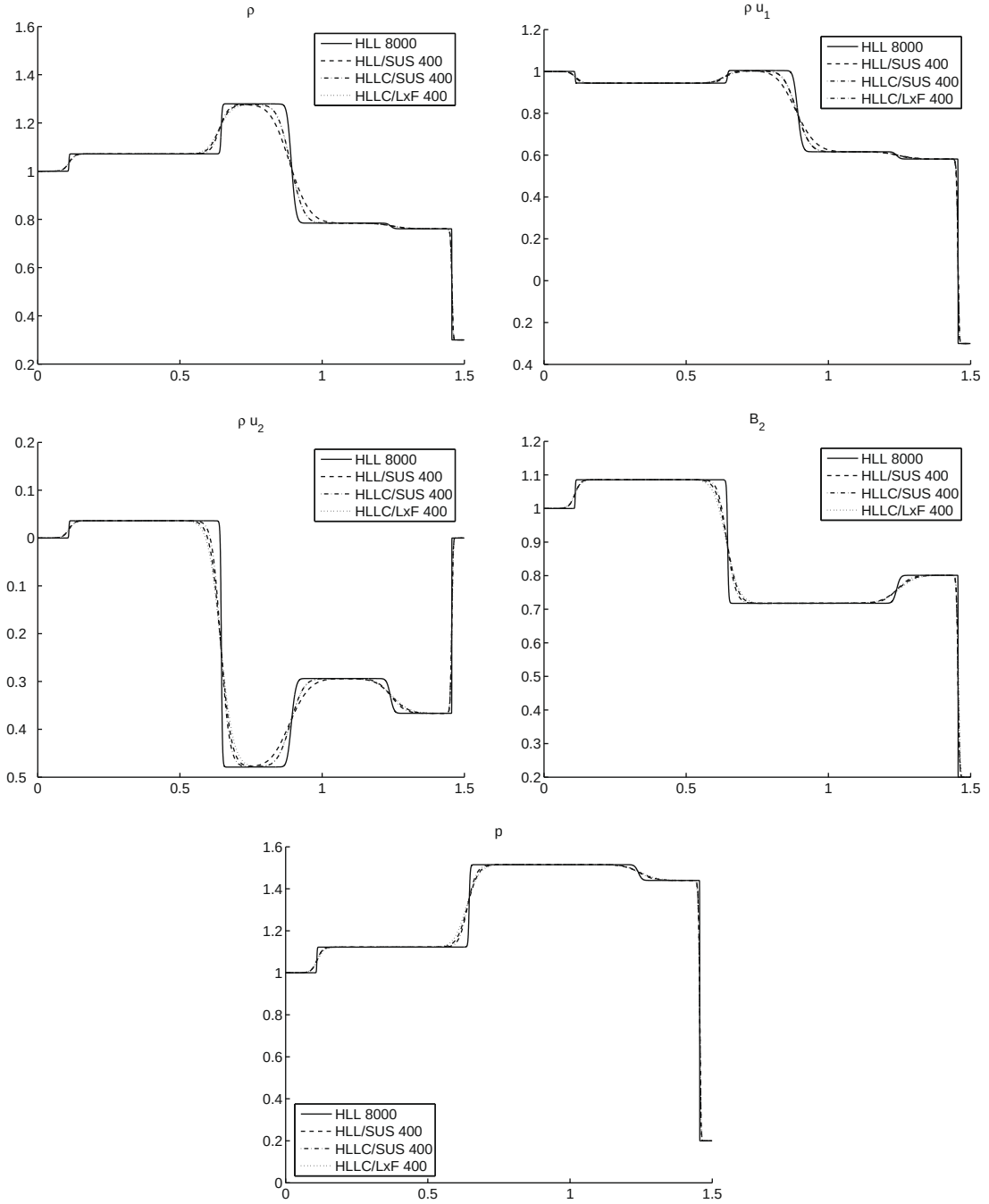

Fig. 1. Results for the Brio-Wu shock tube at time $t = 0.5$ with HLL/SUS, HLLC/SUS and HLLC/LxF with 400 mesh points compared with the reference solution.

Table 1
Relative errors, $\Re E_M^1(\rho)$ (top) and $\Re E_M^1(B^2)$ (bottom), at time $t = 0.5$ for the Brio–Wu shock tube for various mesh sizes M.

M	HLL	HLL3	HLL/SUS	HLLC/SUS	HLLC/LxF	HLLC/TF	Roe
100	4.44	3.35	4.49	3.26	3.30	3.22	2.42
200	2.87	1.95	2.65	1.90	2.08	1.86	1.23
400	1.88	1.22	1.73	1.18	1.26	1.15	0.73
800	1.13	0.64	1.02	0.62	0.73	0.61	0.30
1600	0.62	0.28	0.55	0.27	0.37	0.26	0.10
100	3.32	2.75	3.06	2.81	3.56	2.09	1.89
200	2.04	1.56	1.77	1.55	2.14	1.25	0.85
400	1.29	0.97	1.11	0.95	1.35	0.73	0.50
800	0.75	0.51	0.60	0.48	0.74	0.40	0.18
1600	0.38	0.23	0.29	0.21	0.34	0.16	0.09

two- and three-wave solvers presented above have the resolution comparable to the Roe solver particularly in the density variable. However, recent papers [17,25] have reported that the HLL five wave solver has comparable accuracy as the Roe solver.

This experiment indicates that the splitting scheme are comparable to the standard HLL two- and three-wave schemes in one space dimension. We have performed tests with other shock tubes and obtain similar qualitative and quantitative results.

One of the key robustness criteria in one-dimensional MHD is whether the numerical schemes yield approximations with positive pressure and density. It is well known that the Roe solver is not positivity preserving and the references cited in the introduction prove that the HLL2 solver and other HLL-three and five solvers are proved to be positivity preserving. We are unable to provide a proof that our splitting based solvers are positivity preserving in one space dimension. However, numerical tests indicated that these solvers do preserve positive pressure and density in practice. In order to test this assertion, we present the following numerical experiment.

3.2. Super-fast expansion

This test problem has been used in [18] and other references therein as a test of positivity preservation for one-dimensional MHD solvers. The set up is a shock tube with initial data,

$$(\rho, \rho u_1, \rho u_2, \rho u_3, B^1, B^2, B^3, p) = \begin{cases} (1.0, -3.1, 0, 0, 0, 0.5, 0, 0.45), & \text{if } x < 0.5 \\ (1.0, 3.1, 0, 0, 0, 0.5, 0, 0.45), & \text{otherwise.} \end{cases} \tag{3.2}$$

The data has been selected in such a manner that the exact solution is an expansion with fast wave mach number equal to 3.1. The solution contains very low densities and pressures and is designed to test positivity of solvers. As expected, the HLL2 and HLL3 solvers retained positivity and the Roe solver crashed due to negative pressure. Similarly, the HLLC/LxF solver retained positivity. We present the results for the HLL/SUS and HLLC/SUS solvers in Fig. 2, where we plot the logarithm of the pressure computed by the HLL/SUS and the HLLC/SUS solvers for 400 and 1600 mesh points. The figure clearly demonstrates that both solvers retain positivity. The HLLC/TF scheme crashed due to oscillations.

This experiment does not prove that our HLL/SUS and HLLC/SUS schemes are positivity preserving. However, based on this and many other one dimensional numerical tests, we can conclude that these solvers are robust for one-dimensional problems. Encouraged by these results, we move on to two-dimensional test problems.

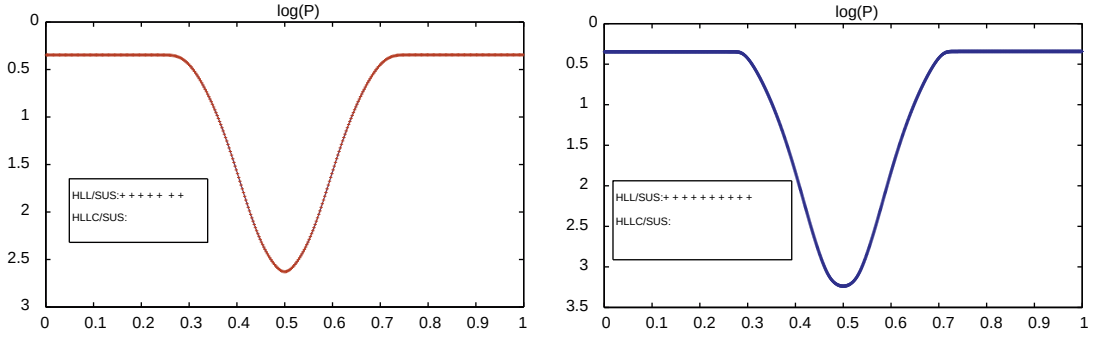

Fig. 2. Numerical results for the superfast expansion problem. We show $log_{10}(P)$ at time $t = 0.2$ with both HLL/SUS and HLLC/SUS solvers. Left: 400 mesh points and right: 1600 mesh points.

3.3. Orszag-Tang vortex

The so-called Orzag-Tang vortex is a well known benchmark test for two dimensional schemes for the MHD equations (see [29]). For this problem, the initial data are given by

$$(\rho, \rho u_1, \rho u_2, \rho u_3, B^1, B^2, B^3, p) = (\gamma^2, -\rho \sin(y), \rho \sin(x), 0.0, -\sin(y), \sin(2x), 0.0, \gamma). \tag{3.3}$$

The computational domain is $(\mathbf{x}, t) \in [0, 2\pi]^2 \times [0, \pi]$ with periodic boundary conditions. We present numerical results with different schemes in Fig. 3 and Table 2.

Even though the initial data are smooth, the solution develops discontinuities in form of shocks along the diagonals, together with a vortex in the center of the domain. The solution has a rich structure consisting of shocks, vortices and other interesting smooth regions. The issues with any numerical scheme are resolution of the shocks as well as the central vortex. Another issue is that of control of divergence in some discrete norm. In fact, it is widely believed that lack of divergence control can lead to negative pressures and hence crashes in this test case. So ensuring stability of the solver, particularly at fine mesh resolutions is a challenge. There is no accepted reference solution in this case and many papers have used the maximum pressure as a measure of accuracy and of a scheme. We plot the pressure on a 100×100 mesh in Fig. 3 and compare HLL, HLL/SUS, HLL3 and HLLC/SUS schemes for qualitative behavior. As shown in Fig. 3, the differences in resolving the solution are much more pronounced in this case. In particular, the HLL scheme is very dissipative and the central vortex is not well resolved. Even the HLL3 scheme is quite dissipative and the shocks along the diagonal are smeared. On the other hand, the schemes based on splitting resolve the shocks very well. The central vortex is resolved by both the HLL/SUS and the HLLC/SUS scheme, and the shocks are much sharper than the features computed by the HLL3 solver for the full MHD equations. A thorough quantitative comparison is provided by considering the maximum pressures in Table 2 for different mesh resolutions. From Table 2, we find that at relatively coarse mesh resolutions (upto 400×400 mesh points), the splitting solv-

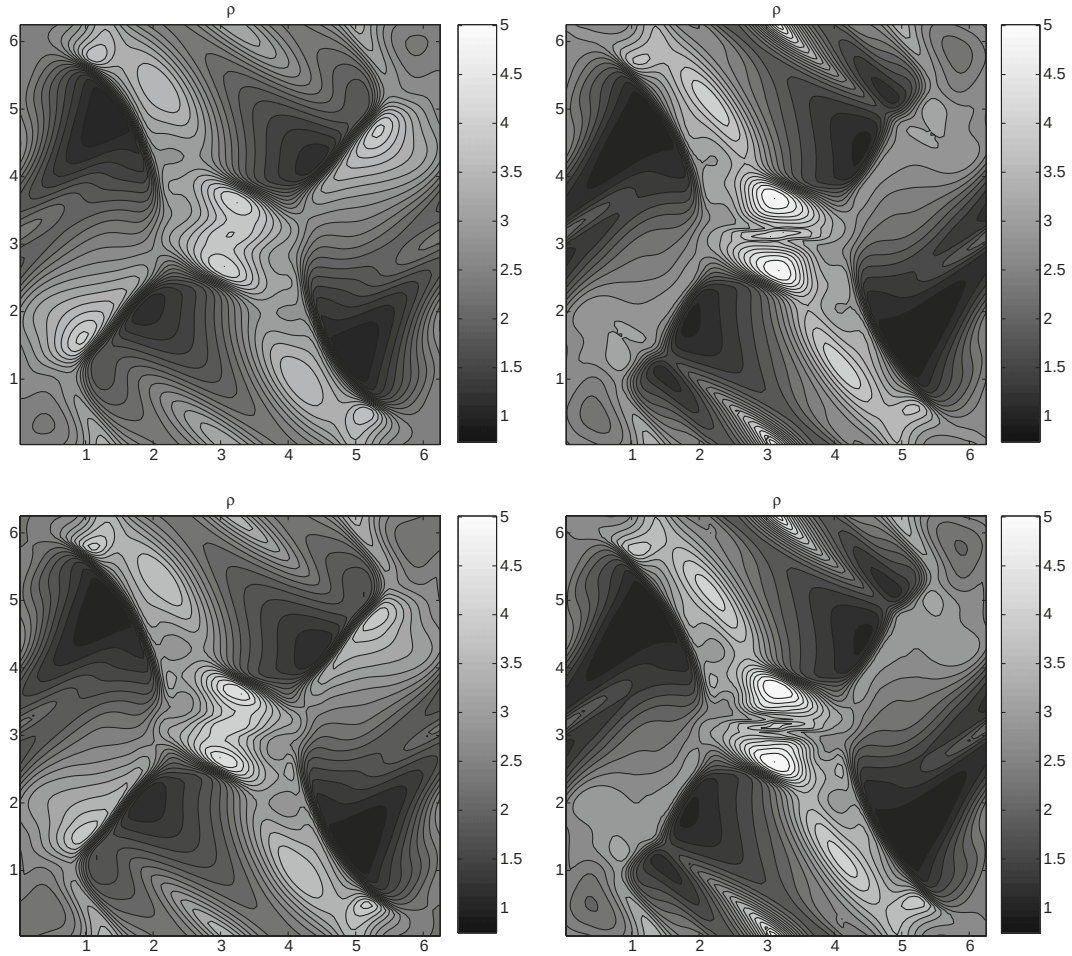

Fig. 3. This figure shows the pressure computed with (from top left to bottom right): HLL2, HLL/SUS, HLL3 and HLLC/SUS on a 100×100 mesh at time $t = \pi$.

Table 2
Maximum pressure (top) and the L^1 norm of the divergence defined by (2.18) (bottom) at time $t = \pi$ for the Orszag-Tang vortex, for different schemes using a $M \times M$ grid.

M	HLL	HLL(Pr)	HLL3	HLL3(Pr)	HLL/SUS	HLLC/SUS	HLLC/LxF	Roe
100	4.00	4.01	4.41	4.42	4.94	5.04	4.61	5.27
200	4.74	4.76	4.94	4.95	5.39	5.41	4.71	5.39
400	5.11	–	5.21	–	5.79	5.81	5.00	5.88
800	–	–	–	–	6.05	6.07	5.26	–
1600	–	–	–	–	6.21	6.22	5.52	–
100	1.92	0.00	2.81	0.00	4.17	4.28	0.00	7.77
200	1.77	0.00	2.93	0.00	3.23	3.32	0.00	6.94
400	1.47	–	2.60	–	2.46	2.50	0.00	5.64
800	–	–	–	–	1.85	1.87	0.00	–
1600	–	–	–	–	1.38	1.39	0.00	–

ers have much higher resolutions (measured in terms of maximum pressure) than the standard HLL and HLL3 solvers. In fact, even the HLL/SUS leads to sharper resolution of the solution than the more expensive HLL3 solver. Similarly, both the splitting solvers compare very well with the Roe solver. In fact, the HLLC/SUS leads to almost the same maximum pressure as the Roe solver. It should added that the computational cost of the Roe solver is considerably higher than the HLLC/SUS solver.

One would expect that this high accuracy of the splitting based solvers should come at a price of reduced stability. On the contrary, Table 2 shows that the splitting based solvers of this paper do not crash even for 1600×1600 mesh points and show increased resolution on these very fine meshes. On the other hand, the standard HLL, HLL3 and Roe solvers crashed on 800×800 and finer meshes due to instabilities. Thus on this test problem, the splitting based HLL solvers are at least as accurate as the Roe solver and more accurate than the HLL solvers and are far more stable.

One possible explanation of this behavior is divergence errors. We compute the standard discrete divergence (2.18) and show the results in Table 2. The initial data has zero divergence, but the solution has several shocks, and numerical solutions may have non-zero divergence. The standard HLL, HLL3 and Roe does not preserve the divergence constraint. The splitting based solvers have some divergence cleaning built into them since they are based on adding the Godunov-Powell source term. Yet, we observe that on coarse meshes, the HLLC/SUS scheme has larger divergence errors than the HLL3 scheme. This could be due to the periodic boundary conditions which re-introduce any non-zero divergence swept away from the domain. Observe from Table 2, that the divergence errors due to the HLL/SUS and HLLC/SUS reduce with increasing mesh resolution. In fact, the divergence error due to the HLLC/SUS is lower than the errors due to both HLL3 scheme and the Roe scheme on the 400×400 mesh. This behaviour of the divergence error might be a reason for the robustness of the splitting based solvers.

Even the HLLC/LxF scheme is fairly robust and we did not observe any crashes. The numerical resolution of the HLLC/LxF scheme is inferior to that of the HLL/SUS and the HLLC/SUS schemes in this case, even though HLLC/LxF preserve divergence. The HLLC/TF scheme crashed even on the coarse 100×100 mesh in this test case.

In order to investigate the connection between divergence errors, stability and overall accuracy, we have tested versions of the HLL, HLL3 and Roe solvers with additional divergence "cleaning". As mentioned in the introduction, a wide variety of divergence cleaning procedures have been proposed, and their relative merits/demerits have been discussed at length, see [29] and references therein. On this particular problem, we decided to couple the HLL and HLL3 solvers with divergence cleaning by the projection method designed in [5] and described in [29]. The projection method is based on the Hodge decomposition of $\mathbf{B}$ and involves projecting $\mathbf{B}$ to a divergence free field at the end of each time step by solving a Poisson equation. We chose the central form of discrete divergence (2.18) and solve the resulting discrete Poisson's equation by a preconditioned conjugate gradient method suggested in [29]. The method is considerably (by several orders of magnitude) more expensive than the basic finite volume schemes as the Poisson solver is used at every time step. The resulting schemes were called HLL(Pr) and HLL3(Pr) indicating that the HLL and HLL3 solvers were coupled with projection based divergence cleaning. The quantitative results with the HLL(Pr) and HLL3(Pr) schemes are shown in Table 2. As shown in this table, for coarse meshes up to 200×200 mesh – the HLL(Pr) and HLL3(Pr) keep divergence zero. But, the quality of numerical solution is very similar to that of the HLL and HLL3 schemes. In fact, the accuracy (measured by the maximum pressure) is practically unchanged when the HLL and HLL3 solvers were equipped with divergence cleaning. This indicates that controlling divergence or preserving the divergence constraint does not increase the accuracy of the scheme. More surprisingly, the HLL(Pr) and HLL3(Pr) schemes crashed even on a 400×400 mesh at times close to $t = \pi/3$ i.e., at one-third of the final time. The results obtained just before the crash showed oscillations near the shocks, and these oscillations may have caused negative pressure. One possible explanation of the oscillations is that the Poisson solver is second order accurate and can introduce slight oscillations. Without divergence cleaning, the HLL and HLL3 solvers did not crash at this mesh resolution (although they crashed on finer meshes).

This test indicates that divergence errors are not necessarily the only cause of instabilities on multi-dimensional MHD simulations, and that divergence cleaning itself can cause instability.

Hence, from this test problem, it appears that the splitting based HLL/SUS and HLLC/SUS solvers are both more accurate (and as accurate as the Roe solver) as well as stable compared with the standard schemes.

3.4. Rotor problem

This two-dimensional test problem is another standard benchmark for numerical solutions of the MHD equations and was first reported in [29].

The computational domain is $(\mathbf{x}, t) \in [0,1]^2 \times [0, 0.295]$ with artificial Neumann type boundary conditions. The initial data are given by

$$\rho = \begin{cases} 10.0 & \text{if } r < 0.1, \\ 1 + 9f(r) & \text{if } 0.1 \leqslant r < 0.115, \\ 1.0 & \text{otherwise}, \end{cases}$$

with $r(\mathbf{x}) = |\mathbf{x} - (0.5, 0.5)|$ and

$$f(r) = \frac{23 - 200r}{3}.$$

The other variables are initially

$$(\rho u^1, \rho u^2) = \begin{cases} (-(10y-5)\rho, (10x-5)\rho) & \text{if } r < 0.1, \\ (-(10y-5)f(r)\rho, (10x-5)f(r)\rho) & \text{if } 0.1 \leqslant r < 0.115, \\ (0.0, 0.0) & \text{otherwise}, \end{cases}$$

$$(\rho u^3, B^1, B^2, B^3, p) = (0.0, 2.5/\sqrt{\pi}, 0.0, 0.0, 0.5).$$

The initial velocity and magnetic fields are such that the variables are rotated in the domain. The pressure drops to very low values in the center, and this test case is set up in order to determine how a scheme handles low pressures. We present numerical results in Table 3 and in Fig. 4. In Fig. 4, we plot the pressure at the final time on a 200×200 mesh and compare the HLL, HLL3, HLL/SUS and HLLC/SUS schemes. At this resolution, we see that the differences in the all the solvers is not as pronounced as the Orszag-Tang vortex even though the HLLC/SUS has the best resolution among the four solvers. We compute the solutions on a series of meshes and report results in Table 3. In the absence of any accepted measure of errors, we report divergence errors in the L^1 norm. From the table, we find that the HLL/SUS and HLLC/SUS solvers are stable even at the fine resolution of 1600×1600 mesh points. Despite the very low pressures, these solvers did not crash for any of the mesh resolutions that we ran in this case. On the other hand, the HLL, HLL3 and Roe solvers crashed at the resolution of 800×800 mesh points due to negative pressures. The divergence errors generated with both the HLL/SUS and HLLC/SUS schemes were quite low on this test problem. The HLLC/LxF scheme was stable but dissipative while the HLLC/TF crashed at all resolutions that we tested. To further investigate the effect of divergence errors on stability, we decided to test a different divergence cleaning technique (than the projection approach used for the Orszag-Tang vortex) for the HLL and HLL3 solvers. As this problem has non-reflecting boundary conditions, we can employ divergence cleaning by the Godunov-Powell source term technique [19]. We implement the discretization of the Godunov-Powell source term described in [20] where the authors used a central discretization of the divergence terms on the right hand sides of (1.10) and evaluated the rest of the right hand side at the given cell. The resulting HLL and HLL3 schemes together with this discretization of the source term were denoted by HLL(Ps) and HLL3(Ps), and the divergence errors are shown in Table 3. As shown in this table, the Powell source term did not affect the divergence errors much. Furthermore, both HLL(Ps) and HLL3(Ps) crashed on a 400×400 mesh. This crash was probably due to slight oscillations introduced by the source term. In [10], we showed that central discretizations of Powell source term can lead to oscillations even for the induction Eq. (1.4). This also happens for the MHD equations as shown by this test example.

3.5. Two-dimensional Riemann problem

This two-dimensional Riemann problem was proposed in [28] as test case to compare stability, resolution as well as divergence control. The computational domain is $(\mathbf{x}, t) \in [0, 0.8]^2 \times [0, 0.1]$ with Neumann boundary conditions. The initial conditions are given by

Table 3
The discrete divergence defined by (2.18) at $t = 0.295$ for the Rotor problem using a $M \times M$ grid.

M	HLL	HLL2(Ps)	HLL3	HLL3(Ps)	HLL/SUS	HLLC/SUS	Roe
100	6.9e−2	6.4e−2	1.1e−1	9.7e−2	7.1e−2	7.8e−2	1.2e−1
200	6.3e−2	6.2e−2	9.3e−2	9.4e−2	5.5e−2	6.1e−2	1.0e−1
400	5.2e−2	–	7.3e−2	–	4.3e−2	4.6e−2	9.9e−2
800	–	–	–	–	3.2e−2	3.4e−2	–
1600	–	–	–	–	2.3e−2	2.5e−2	–

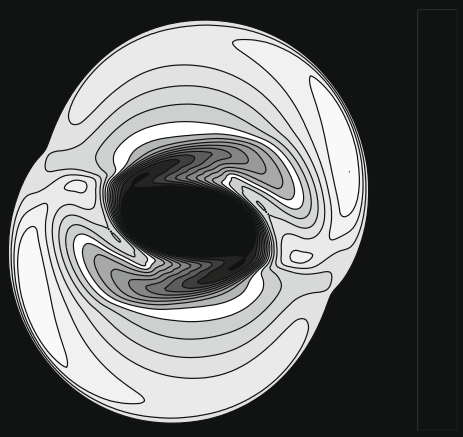

Fig. 4. Pressure at $t = 0.295$ for the Rotor problem on a 200×200 mesh. Top left: HLL, top right: HLL/SUS, bottom left: HLL3, bottom right: HLLC/SUS.

Table 4
Two-dimensional Riemann problem; the L^1 norm of the discrete divergence defined by (2.48), using a $M \times M$ mesh.

M	HLL	HLL3	HLL/SUS	HLLC/SUS	HLLC/TF	Roe
125	2.9e−1	2.9e−1	2.9e−1	2.9e−1	5.8e−3	3.1e−1
250	2.9e−1	3.0e−1	2.9e−1	2.9e−1	4.9e−3	3.1e−1
500	3.0e−1	3.0e−1	2.9e−1	2.9e−1	3.5e−1	3.1e−1
1000	3.0e−1	3.0e−1	2.9e−1	2.9e−1	–	3.1e−1
2000	3.0e−1	3.0e−1	3.0e−1	3.0e−1	–	3.1e−1

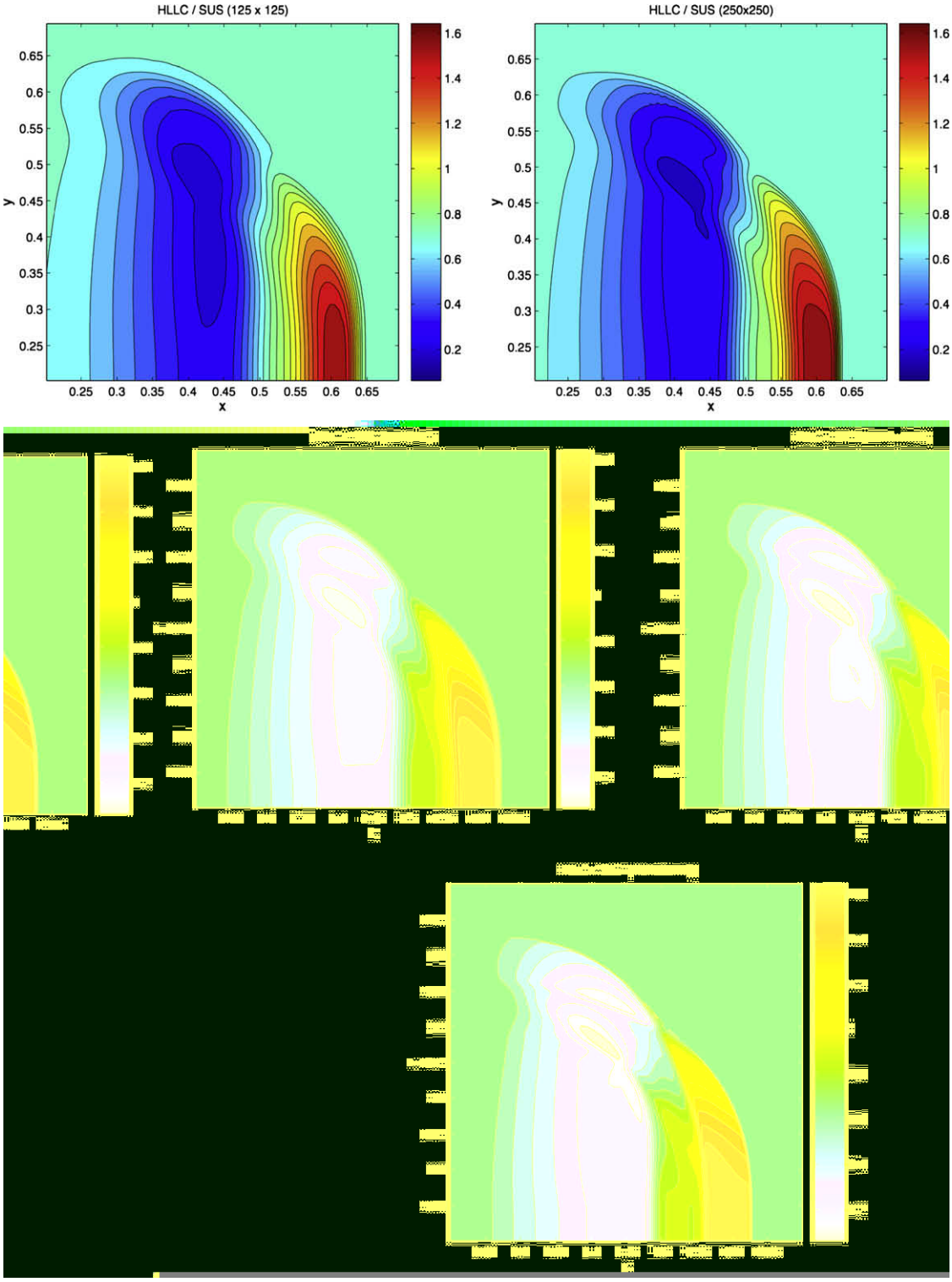

Fig. 5. Convergence under mesh refinement for the HLLC/SUS scheme with initial data given by (3.4).

4. Conclusion

We have presented finite volume schemes for the ideal MHD equations. Our schemes are based on splitting the ideal MHD equations into two parts: a fluid part consisting of an extended Euler system (2.2) with the magnetic forces as source terms, and a magnetic part consisting of the magnetic induction Eq. (1.4) with fluid velocity driving the evolution of the magnetic field. The extended Euler equations are solved by designing HLL two-wave and HLL-three-wave solvers with staggered magnetic fields. This form of the equations leads to an implicit upwinding of the magnetic force terms while retaining conservation of the fluid variables. The magnetic induction equation is solved by stable upwind schemes designed in a recent paper [10]. The schemes are based on adding a suitable Godunov-Powell source term and upwinding this source term. The two types of schemes are combined either simultaneously or using an operator splitting procedure.

We have tested the schemes in a series of benchmark numerical experiments and compared them with standard HLL and Roe type approximate Riemann solvers. In the one-dimensional test cases, the HLL/SUS and the HLLC/SUS schemes are comparable in their accuracy to the standard HLL and HLL-three-wave solvers. The HLLC/LxF scheme was also reasonably accurate. HLLC/TF scheme was even more accurate than the HLLC/SUS scheme but the resulting approximations contained small oscillations. Although we were unable to prove that HLL/SUS and HLLC/SUS are positivity preserving in one space dimension, we present numerical evidence showing that these schemes are positivity preserving in practice.

The differences between the schemes were more pronounced in the two-dimensional test cases. We observed that the splitting based HLL/SUS and HLLC/SUS schemes were more accurate than the standard HLL and HLL-three solvers on coarse meshes. In fact, the HLLC/SUS scheme was as accurate as the Roe scheme on some experiments. On fine meshes, the standard schemes crashed due to negative pressures, whereas the HLL/SUS, HLLC/SUS and HLLC/LxF scheme were stable and led to a good approximation of complex flow features. Furthermore, we also added extra divergence cleaning to the standard HLL and HLL-three solvers in form of the projection method and the discretized Godunov-Powell source term. These divergence cleaning technique did not increase the stability of the standard HLL solvers. This should be contrasted with the robustness observed for the splitting, particularly on very fine meshes.

Given the fact that the HLL/SUS and HLLC/SUS schemes are simpler to design, easier to implement, as accurate and more stable than the standard HLL-three and Roee solvers, these schemes can replace existing HLL and Roe solvers in practical computations.

References

[1] D.S. Balsara, D. Spicer, A staggered mesh algorithm using high order Godunov fluxes to ensure solenoidal magnetic fields in magnetohydrodynamic simulations, J. Comp. Phys. 149 (2) (1999) 270–292.
[2] T.J. Barth, Numerical methods for gas dynamics systems, in: D. Kröner, M. Ohlberger, C. Rohde (Eds.), An Introduction to Recent Developments in Theory and Numerics for Conservation Laws, Springer, 1999.
[3] N. Besse, D. Kröner, Convergence of the locally divergence free discontinuous Galerkin methods for induction equations for the 2D-MHD system, M2AN Math. Model. Num. Anal. 39 (6) (2005) 1177–1202.
[4] F. Bouchut, C. Klingenberg, K. Waagan, A multi-wave approximate Riemann solver for ideal MHD based on relaxation I- theoretical framework, Numer. Math. 108 (1) (2007) 7–42.
[5] J.U. Brackbill, D.C. Barnes, The effect of nonzero div$B$ on the numerical solution of the magnetohydrodynamic equations, J. Comp. Phys. 35 (1980) 426–430.
[6] P. Cargo, G. Gallice, Roe matrices for ideal MHD and systematic construction of Roe matrices for systems of conservation laws, J. Comp. Phys. 136 (2) (1997) 446–466.
[7] W. Dai, P.R. Woodward, A simple finite difference scheme for multi-dimensional magnetohydrodynamic equations, J. Comp. Phys. 142 (2) (1998) 331–369.
[8] B. Einfeldt, On the Godunov type methods for gas dynamics, SIAM.J. Num. Anal. 25 (2) (1988) 294–318.
[9] C. Evans, J.F. Hawley, Simulation of magnetohydrodynamic flow: a constrained transport method, Astrophys. J. 332 (1998) 659.
[10] F. Fuchs, K.H. Karlsen, S. Mishra, N.H. Risebro, Stable upwind schemes for the Magnetic Induction equation. Math. Model. Num. Anal,. Available on conservation laws preprint server, submitted for publication, URL: <http://www.math.ntnu.no/conservation/2007/029.html>.
[11] S.K. Godunov, The symmetric form of magnetohydrodynamics equation, Num. Meth. Mech. Cont. Media 1 (1972) 26–34.
[12] K.F. Gurski, An HLLC-type approximate Riemann solver for ideal Magneto-hydro dynamics, SIAM. J. Sci. Comp. 25 (6) (2004) 2165–2187.
[13] T.J. R Hughes, L.P. Franca, M. Mallet, A new finite element formulation for CFD I: symmetric forms of the compressible Euler and Navier-Stokes equations and the second law of thermodynamics, Comp. Meth. Appl. Mech. Engg. 54 (1986) 223–234.
[14] K.H. Karlsen, S. Mishra, N.H. Risebro, Semi-Goudunov schemes for systems of conservation laws, J. Eng. Math. 60 (2007) 337–349.
[15] R.J. LeVeque, Finite Volume Methods for Hyperbolic Problems, Cambridge University Press, Cambridge, 2002.
[16] T.J. Linde, A three adaptive multi fluid MHD model for the heliosphere, Ph.D Thesis, University of Michigan, Ann-Arbor, 1998.
[17] A. Mignone et al, Pluto: a numerical code for computational astrophysics, Astrophys. J. Suppl. 170 (2007) 228–242.
[18] T. Miyoshi, K. Kusano, A multi-state HLL approximate Riemann solver for ideal magneto hydro dynamics, J. Comp. Phys. 208 (1) (2005) 315–344.
[19] K.G. Powell, An approximate Riemann solver for magneto-hydro dynamics (that works in more than one space dimension), Technical Report, vol. 94 – 24, ICASE, Langley, VA, 1994.
[20] K.G. Powell, P.L. Roe. T.J. Linde, T.I. Gombosi, D.L. De zeeuw, A solution adaptive upwind scheme for ideal MHD, J. Comp. Phys. 154 (2) (1999) 284–309.
[21] P.L. Roe, Approximate Riemann solvers, parameter vectors and difference schemes, J. Comput. Phys. 43 (1981) 357–372.
[22] P.L. Roe, D.S. Balsara, Notes on the eigensystem of magnetohydrodynamics, SIAM. J. Appl. Math. 56 (1) (1996) 57–67.
[23] J. Rossmanith, A wave propagation method with constrained transport for shallow water and ideal magnetohydrodynamics, Ph.D Thesis, University of Washington, Seattle, 2002.

[24] D.S. Ryu, F. Miniati, T.W. Jones, A. Frank, A divergence free upwind code for multidimensional magnetohydrodynamic flows, Astrophys. J. 509 (1) (1998) 244–255.
[25] J.M. Stone, T.A. Gardiner, P. Teuben, J.F. Hawley, J.B. Simon, Athena: a new code for astrophysical MHD, Astrophys. J. Suppl. 178 (2008) 137–177.
[26] E.F. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics, Springer Verlag, Berlin, 1997.
[27] M. Torrilhon, M. Fey, Constraint-preserving upwind methods for multidimensional advection equations, SIAM.J. Num. Anal. 42 (4) (2004) 1694–1728.
[28] M. Torrilhon, Locally divergence preserving upwind finite volume schemes for magnetohyrodynamic equations, SIAM. J. Sci. Comp. 26 (4) (2005) 1166–1191.
[29] G. Toth, The div$B = 0$ constraint in shock capturing magnetohydrodynamics codes, J. Comp. Phys. 161 (2000) 605–652.