New Astronomy 14 (2009) 139–148

Contents lists available at ScienceDirect

New Astronomy

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

A simple unsplit Godunov method for multidimensional MHD

James M. Stone *, Thomas Gardiner [1]

Department of Astrophysical Sciences, Princeton University, Princeton, NJ 08544, United States

ARTICLE INFO

Article history:
Received 3 April 2008
Received in revised form 20 June 2008
Accepted 24 June 2008
Available online 3 July 2008
Communicated by G.F. Gilmore

PACS:
95.30.Qd
95.75.Pq

Keywords:
Hydrodynamics
MHD
Methods: numerical

ABSTRACT

We describe a numerical algorithm based on Godunov methods for integrating the equations of compressible magnetohydrodynamics (MHD) in multidimensions. It combines a simple, dimensionally-unsplit integration method with the constrained transport (CT) discretization of the induction equation to enforce the divergence-free constraint. We present the results of a series of fully three-dimensional tests which indicate the method is second-order accurate for smooth solutions in all MHD wave families, and captures shocks, contact and rotational discontinuities well. However, it is also more diffusive than other more complex unsplit integrators combined with CT. Thus, the primary advantage of the method is its simplicity. It does not require a characteristic tracing step to construct interface values for the Riemann solver, it is straightforward to extend with additional physics, and it is suitable for use with nested and adaptive meshes. The method is implemented as one of two dimensionally unsplit MHD integrators in the Athena code, which is freely available for download from the web.

1. Introduction

Numerical methods for compressible magnetohydrodynamics (MHD) have become indispensable for the study of a wide variety of problems in astrophysics, such as turbulence and angular momentum transport in accretion disks (Balbus, 2003), the production and collimation of jets and winds (de Gouveia Dal Pino, 2005), and the properties of supersonic turbulence in the ISM (McKee and Ostriker, 2007). Although a wide variety of numerical algorithms for integrating the equations of MHD are possible, there has been a substantial amount of effort in recent years to extend Godunov methods to MHD (Dai and Woodward, 1998; Ryu et al., 1998; Falle et al., 1998; Powell et al., 1999; Balsara and Spicer, 1999; Tóth, 2000,; Dedner et al., 2002; Pen et al., 2003; Londrillo and Del Zanna, 2004; Ziegler, 2004; Crockett et al., 2005; Fromang et al., 2006; Mignone et al., 2007; Cunningham et al., 2007), primarily because such methods are very good for shock capturing, and flows involving contact and/or rotational discontinuities. In addition, single-step Eulerian methods that solve the equations of motion in the conservative form are better suited for use with static or adaptive mesh refinement (SMR or AMR, respectively).

In a recent series of papers (Gardiner and Stone, 2005, 2008, hereafter GS05 and GS08, respectively), we have described the extension of the directionally unsplit corner transport upwind (CTU) integrator of Colella (1990) to MHD using the constrained transport (CT) method of Evans and Hawley (1988) to enforce the divergence-free constraint on the magnetic field. We refer to this combination as the CTU + CT algorithm. Several novel ingredients of the MHD CTU + CT algorithm were shown to be important. In particular, spatial reconstruction schemes that contain a directionally-split time advance (such as the piecewise parabolic method, PPM, of Colella and Woodward, 1984, hereafter CW), must include additional source terms in multidimensional MHD compared to the one-dimensional (1D) case. Moreover, the transverse flux gradients used in CTU to correct the interface states in multidimensions also require similar terms. The nature and form of these terms is documented in detail in Sections 3.1 and 4.1.2 of GS05 for two dimensions (2D), and Sections 3.1 and 4.2 in GS08 for three dimensions (3D). In addition, it was shown that simple arithmetic averaging is not sufficient to compute the edge-centered electric fields required by CT from the face-centered fluxes returned by the Riemann solver. Instead, simple reconstruction algorithms were developed and tested (GS05).

More recently, we have presented a detailed description of the implementation of the CTU + CT algorithm in Athena, a new code for astrophysical MHD (Stone et al., in press, hereafter S08). The results of a comprehensive hydrodynamic and MHD test suite were used to demonstrate the fidelity of the CTU + CT method in 1D, 2D, and 3D.

During the course of developing the CTU + CT algorithm in Athena, we have experimented with other dimensionally unsplit integrators for MHD. We have found that a simple predictor–corrector scheme (see the appendix in Falle, 1991) similar to the MUSCL–Hancock scheme described by van Leer (2006; see also Toro 1999) can form the basis of a robust Godunov method for MHD when combined with CT. We refer to this combination as the VL + CT algorithm.

* Corresponding author. Fax: +1 609 258 8226.
E-mail address: jmstone@princeton.edu (J.M. Stone).
[1] Present address: 3915 Rayado Pl NW, Albuquerque, NM 87114, United States.

1384-1076/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.newast.2008.06.003

The purpose of this paper is to provide a comprehensive description of the VL + CT algorithm as implemented in Athena.

The VL + CT method has several advantages over the CTU + CT integrator, including (1) it does not require a time-advance using characteristic tracing in the reconstruction step, thus eliminating the need for a characteristic decomposition of the equations of motion in the primitive variables, (2) for the same reason, it does not require the source terms in the spatial reconstruction step for multidimensional MHD that are needed in the CTU + CT integrator, (3) it does not require the MHD source terms associated with the transverse flux gradients that are needed in the CTU + CT integrator (4) physical source terms are easy to add at second order, and (5) it seems to be more robust than the CTU + CT integrator in extreme parameter regimes. The first three of these advantages make VL + CT integrator particularly attractive for more complex physics than ideal MHD. On the other hand, the tests presented in this paper show that the VL + CT integrator is more diffusive and less accurate than CTU + CT. Nonetheless, we anticipate that VL + CT may be of interest to others developing or testing their own methods, or those who are interested in using Athena.

The outline of this paper is as follows. In the following two sections, we write down the equations of ideal MHD solved by the method, and discuss their finite-volume and finite-area discretization. Although these sections repeat the description in S08 to some extent, we include them here to make this paper self-contained. In Section 4 we describe the second-order spatial reconstruction algorithm needed to compute the interface states used by the Riemann solver to compute upwind fluxes, and how the area-averaged face-centered electric fields returned by the Riemann solver are converted into line-averages at cell-corners as needed by CT to evolve the field. The various Riemann solvers used with the method are also briefly cataloged. In Section 5 a step-by-step description of the algorithm is given, while in Section 6 the results of 3D tests are shown, with the results of the VL + CT method described in this paper compared directly to the results of the CTU + CT integrator described in S08. Finally, in Section 7 we conclude.

2. Equations

The VL + CT algorithm solves the equations of ideal MHD, which when written in conservative form are

$$\frac{\partial \rho}{\partial t} + \nabla\cdot(\rho\mathbf{v}) = 0, \tag{1}$$

$$\frac{\partial \rho\mathbf{v}}{\partial t} + \nabla\cdot(\rho\mathbf{v}\mathbf{v} - \mathbf{B}\mathbf{B} + P^*) = 0, \tag{2}$$

$$\frac{\partial E}{\partial t} + \nabla\cdot((E+P^*)\mathbf{v} - \mathbf{B}(\mathbf{B}\cdot\mathbf{v})) = 0, \tag{3}$$

$$\frac{\partial \mathbf{B}}{\partial t} - \nabla\times(\mathbf{v}\times\mathbf{B}) = 0, \tag{4}$$

where $P^* = P + (\mathbf{B}\cdot\mathbf{B})/2$ is the total pressure (gas plus magnetic), E is the total energy density, and we have chosen a system of units in which the magnetic permeability $\mu = 1$. We use an ideal gas equation of state for which the gas pressure can be written as $P = (\gamma-1)\epsilon$, where γ is the ratio of specific heats. The internal energy density ϵ is related to the total energy E via

$$E \equiv \epsilon + \rho(\mathbf{v}\cdot\mathbf{v})/2 + (\mathbf{B}\cdot\mathbf{B})/2. \tag{5}$$

The method described can be used for any value of γ. For a barotropic equation of state, $P = P(\rho)$, Eqs. (3) and (5) are dropped from the system.

3. Discretization

To develop the discrete form of the MHD equations, we adopt a uniform Cartesian grid. The computational domain is divided into cells of size $\delta x = L_x/N_x$, $\delta y = L_y/N_y$, $\delta z = L_z/N_z$, where for example L_x and N_x are the size of the domain and number of cells in the x-direction respectively. We use the standard notation that the cell denoted by indices (i,j,k) is located at cell-center position (x_i, y_j, z_k). Time in the interval $t \in (t_0, t_f)$ over which the integration is carried out is discretized into N non-uniform steps. A superscript is used to denote time levels, so $t^{n+1} - t^n = \delta t^n$; hereafter we drop the superscript on δt.

We use the finite-volume discretization of Eqs. (1) through (3), and a finite-area discretization of Eq. (4). Integrating over the volume of the cell (i,j,k), and over a time step δt, after application of the divergence theorem, gives

$$\mathbf{U}^{n+1}_{i,j,k} = \mathbf{U}^{n}_{i,j,k} - \frac{\delta t}{\delta x}\left(\mathbf{F}^{n+1/2}_{i+1/2,j,k} - \mathbf{F}^{n+1/2}_{i-1/2,j,k}\right) - \frac{\delta t}{\delta y}\left(\mathbf{G}^{n+1/2}_{i,j+1/2,k} - \mathbf{G}^{n+1/2}_{i,j-1/2,k}\right) - \frac{\delta t}{\delta z}\left(\mathbf{H}^{n+1/2}_{i,j,k+1/2} - \mathbf{H}^{n+1/2}_{i,j,k-1/2}\right), \tag{6}$$

where $\mathbf{U}^n_{i,j,k}$ denotes the volume-average at time level n

$$\mathbf{U}^{n}_{i,j,k} = \frac{1}{\delta x\delta y\delta z}\int_{z_{k-1/2}}^{z_{k+1/2}}\int_{y_{j-1/2}}^{y_{j+1/2}}\int_{x_{i-1/2}}^{x_{i+1/2}} \mathbf{U}(x,y,z,t^n)\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z, \tag{7}$$

while the vector $\mathbf{F}^{n+1/2}_{i-1/2,j,k}$ is the time- and area-averaged flux through the "x-faces" of the cell (surfaces of the cell with unit normal in the x-direction) at the location $x_{i-1/2}$

$$\mathbf{F}^{n+1/2}_{i-1/2,j,k} = \frac{1}{\delta y\delta z\delta t}\int_{t^n}^{t^{n+1}}\int_{z_{k-1/2}}^{z_{k+1/2}}\int_{y_{j-1/2}}^{y_{j+1/2}} \mathbf{F}(x_{i-1/2},y,z,t)\,\mathrm{d}y\,\mathrm{d}z\,\mathrm{d}t. \tag{8}$$

Similar expressions can be written for $\mathbf{G}^{n+1/2}_{i,j-1/2,k}$ and $\mathbf{H}^{n+1/2}_{i,j,k-1/2}$, the time- and area-averaged fluxes through the y- and z-faces of the cell located at $y_{j-1/2}$ and $z_{k-1/2}$. For adiabatic MHD the vectors $\mathbf{U}$ and $\mathbf{F}$ have components

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho v_x \\ \rho v_y \\ \rho v_z \\ E \\ B_x \\ B_y \\ B_z \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \rho v_x \\ \rho v_x^2 + P + B^2/2 - B_x^2 \\ \rho v_x v_y - B_x B_y \\ \rho v_x v_z - B_x B_z \\ (E+P^*)v_x - (\mathbf{B}\cdot\mathbf{v})B_x \\ 0 \\ B_y v_x - B_x v_y \\ B_z v_x - B_x v_z \end{bmatrix}. \tag{9}$$

The components of $\mathbf{G}$ and $\mathbf{H}$ are given in S08.

Although the last three components of Eqs. (6) through (9) represent the finite-volume discretization of the induction equation based on the volume-averaged components of the magnetic field, we do not use this form to integrate the magnetic field. Instead, the appropriate integral form of the induction Eq. (4) is given by integration over the *surfaces* rather than the *volume* of the cell. For example, integrating Eq. (4) over the (x,y,z)-faces of the cell located at $(i-1/2,j,k)$, $(i,j-1/2,k)$, and $(i,j,k-1/2)$ respectively gives, after use of Stoke's Law

$$B^{n+1}_{x,i-1/2,j,k} = B^{n}_{x,i-1/2,j,k} - \frac{\delta t}{\delta y}\left(\mathscr{E}^{n+1/2}_{z,i-1/2,j+1/2,k} - \mathscr{E}^{n+1/2}_{z,i-1/2,j-1/2,k}\right) + \frac{\delta t}{\delta z}\left(\mathscr{E}^{n+1/2}_{y,i-1/2,j,k+1/2} - \mathscr{E}^{n+1/2}_{y,i-1/2,j,k-1/2}\right), \tag{10}$$

$$B^{n+1}_{y,i,j-1/2,k} = B^{n}_{y,i,j-1/2,k} + \frac{\delta t}{\delta x}\left(\mathscr{E}^{n+1/2}_{z,i+1/2,j-1/2,k} - \mathscr{E}^{n+1/2}_{z,i-1/2,j-1/2,k}\right) - \frac{\delta t}{\delta z}\left(\mathscr{E}^{n+1/2}_{x,i,j-1/2,k+1/2} - \mathscr{E}^{n+1/2}_{x,i,j-1/2,k-1/2}\right), \tag{11}$$

$$B^{n+1}_{z,i,j,k-1/2} = B^{n}_{z,i,j,k-1/2} - \frac{\delta t}{\delta x}\left(\mathscr{E}^{n+1/2}_{y,i+1/2,j,k-1/2} - \mathscr{E}^{n+1/2}_{y,i-1/2,j,k-1/2}\right) + \frac{\delta t}{\delta y}\left(\mathscr{E}^{n+1/2}_{x,i,j+1/2,k-1/2} - \mathscr{E}^{n+1/2}_{x,i,j-1/2,k-1/2}\right), \tag{12}$$

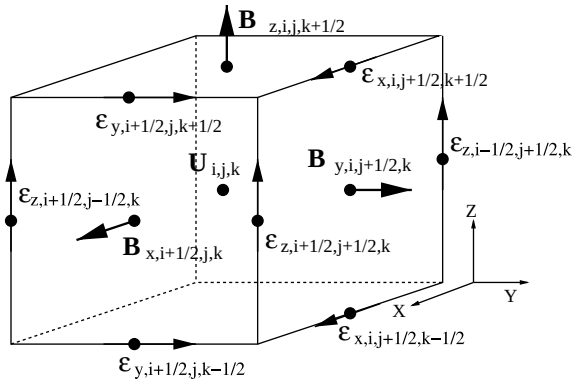

Fig. 1. Centering of the volume-averaged hydrodynamic variables **U**, are-averaged components of the magnetic field **B**, and line-averaged emfs used to evolve the field $\mathscr{E}$ on the computational grid. The area-averaged fluxes of **U** returned by the Riemann solver are co-located with the field **B**.

where, for example,

$$B^n_{x,i-1/2,j,k} = \frac{1}{\delta y \delta z}\int_{z_{k-1/2}}^{z_{k+1/2}}\int_{y_{j-1/2}}^{y_{j+1/2}} B_x(x_{i-1/2}, y, z, t^n)\,\mathrm{d}y\,\mathrm{d}z \tag{13}$$

is the area-averaged x-component of the magnetic field, and

$$\mathscr{E}^{n+1/2}_{x,i,j-1/2,k-1/2} = \frac{1}{\delta x \delta t}\int_{t^n}^{t^{n+1}}\int_{x_{i-1/2}}^{x_{i+1/2}} \mathscr{E}_x(x, y_{j-1/2}, z_{k-1/2}, t)\,\mathrm{d}x\,\mathrm{d}t \tag{14}$$

is the x-component of the electric field $\mathscr{E} = -\mathbf{v}\times\mathbf{B}$ (or emf) averaged along the appropriate line element.

Fig. 1 summarizes the centering of the volume-averaged components of $\mathbf{U}^n_{i,j,k}$, the area-averaged components of the magnetic field at cell faces, and the line-averaged components of the emfs $\mathscr{E}^{n+1/2}_{x,i,j-1/2,k-1/2}$, etc., located at cell edges. The time- and area-averaged fluxes $\mathbf{F}^{n+1/2}_{i-1/2,j,k}, \mathbf{G}^{n+1/2}_{i,j-1/2,k}$, and $\mathbf{H}^{n+1/2}_{i,j,k-1/2}$ are co-located with the components of the magnetic field.

As discussed in S08, in Athena the face-centered fields are evolved with CT, and the volume averaged fields are then computed from the second-order accurate averages

$$B_{x,i,j,k} = \frac{1}{2}(B_{x,i+1/2,j,k} + B_{x,i-1/2,j,k}), \tag{15}$$

$$B_{y,i,j,k} = \frac{1}{2}(B_{y,i,j+1/2,k} + B_{y,i,j-1/2,k}), \tag{16}$$

$$B_{z,i,j,k} = \frac{1}{2}(B_{z,i,j,k+1/2} + B_{z,i,j,k-1/2}). \tag{17}$$

The face-centered values are always considered the primary representation of the magnetic field.

4. Calculating the fluxes

To compute estimates of the fluxes of the volume-averaged variables, for example $\mathbf{F}^{n+1/2}_{i-1/2,j,k}$ at an x-interface, a variety of 1D Riemann solvers can be used. These require values for each variable at the left- and right-side of each cell interface, some of which must be reconstructed from the neighboring cell-centered values. Finally, the area-averaged electric field returned by the Riemann solver must be reconstructed to the edges of the cell to update the magnetic field using CT (see Fig. 1). Each of these ingredients of the algorithm is described in the sections below.

4.1. Riemann solvers

An excellent introduction to the use of a Riemann solver to compute the time-averaged flux through each interface in Godunov methods is given in Toro (1999). Complex and expensive Riemann solvers are not necessary to compute upwind fluxes, inexpensive and approximate solvers that do not require a characteristic decomposition can be used. More complex solvers are *preferred* because they are much more accurate, but they are not *required*.

In Athena, the same set of Riemann solvers that are used with the CTU + CT algorithm (described in Section 4.3 of S08) can be used with the VL + CT method described here. These include nonlinear solvers such as HLLE (Harten et al., 1983; Einfeldt et al., 1991), HLLC for hydrodynamics (Batten et al., 1997; Toro, 1999), and HLLD for MHD (Miyoshi and Kusano, 2005), or linear solvers such as Roe's method (Roe, 1981) extended to MHD (Cargo and Gallice, 1997). Other solvers include Toro's FORCE flux (Toro, 1999), and exact solvers in simple cases (isothermal hydrodynamics). The primary criteria for choosing a solver are speed and accuracy. We have found the HLLC solver for hydrodynamics, the HLLD solver for MHD, or Roe's solver in either case to be the most useful for applications. Implementation of these solvers is discussed in more detail in S08.

4.2. Computing the interface states

The Riemann solvers discussed above require values for the conserved variables to the left (denoted by $\mathbf{U}_{L,i-1/2}$) and to the right (denoted by $\mathbf{U}_{R,i-1/2}$) of the cell interface located at $x_{i-1/2}$. These must be reconstructed from cell centered values. With CT, only the components of the magnetic field transverse to the interface need be reconstructed; the longitudinal component is given by the appropriate face-centered value (for example, $B_{x,i-1/2,j,k}$ in the case of reconstruction in the x-direction to the interface located at $(i-1/2, j, k)$). For the transverse fields, the cell-centered averages of the face-centered fields (Eqs. (16) and (17)) are used.

For second-order spatial accuracy, piecewise linear interpolation is used. This interpolation is carried out in the primitive variables, $\mathbf{W} = (\rho, \mathbf{v}, P, \mathbf{B})$ as tests show this makes the reconstruction less oscillatory. For the same reason, slope limiters are required in the reconstruction. The reconstruction can be summarized using the following steps:

Step 1. Convert from conserved to primitive variables at cell centers.

Step 2. Compute the left-, right-, and centered-differences of each of the primitive variables $\mathbf{w}_i$ except the longitudinal component of the magnetic field (e.g. B_x for reconstruction in the x-direction). These differences are defined as

$$\begin{aligned}\delta\mathbf{w}_{L,i} &= \mathbf{w}_i - \mathbf{w}_{i-1},\\ \delta\mathbf{w}_{R,i} &= \mathbf{w}_{i+1} - \mathbf{w}_i,\\ \delta\mathbf{w}_{C,i} &= (\mathbf{w}_{i+1} - \mathbf{w}_{i-1})/2,\end{aligned} \tag{18}$$

where the subscripts L, R, and C refer to locations relative to the cell-center at x_i.

Step 3. Apply monotonicity constraints to the differences in the characteristic variables, so that the characteristic reconstruction is total variation diminishing (TVD), e.g. see LeVeque (2002), for example

$$\delta\mathbf{w}^m_i = \mathrm{SIGN}(\delta\mathbf{w}_{C,i})\min(2\mid\delta\mathbf{w}_{L,i}\mid, 2\mid\delta\mathbf{w}_{R,i}\mid, \mid\delta\mathbf{w}_{C,i}\mid). \tag{19}$$

Note that the minmod limiter used in this step is only one of a wide variety that can be adopted (LeVeque, 2002).

Step 4. Compute the left- and right-interface values using the monotonized difference in the primitive variables

$$\hat{\mathbf{w}}_{L,i+1/2} = \mathbf{w}_i + \delta\mathbf{w}^m_i/2, \tag{20}$$

$$\hat{\mathbf{w}}_{R,i-1/2} = \mathbf{w}_i - \delta\mathbf{w}^m_i/2. \tag{21}$$

It is instructive to compare these steps to those required for the second-order (PLM) reconstruction used with the CTU + CT algorithm described in Section 4.2.2 of S08. Note that the slope limiting in step 3 above is performed in the primitive variables. In fact, limiting in the characteristic variables can improve the method, although it is not essential. If characteristic limiting is used, step 3 above is replaced with steps 3 through 5 in Section 4.2.2 of S08, but using $\delta t = 0$ (no time advance of the interface states is needed). The tests reported in this paper all use slope limiting in the characteristic variables.

We emphasize that the characteristic tracing step required to time-advance the interface states with CTU + CT (step 7 in Section 4.2.2 of S08) is *not* required here. This greatly simplifies the reconstruction. Moreover, it eliminates the need for additional terms for multidimensional MHD that arise due to the use of a directionally-split update in the characteristic tracing step, that is the source terms discussed in Section 5.2 in S08 are also *not* needed.

Finally, it is also possible to combine higher than second-order spatial reconstruction with the VL + CT algorithm described here. For example, the PPM of CW could be used (again, without the characteristic tracing step). Although this does not increase the formal order of accuracy of the method, we have found it can increase the overall accuracy of the solution for smooth flow in some circumstances.

4.3. Calculating the emfs

In GS05, it was shown that a simple arithmetic average of the face-centered fluxes of the magnetic field returned by the Riemann solver (called the "flux-CT" method in T2000) to construct the line-averaged emfs at cell corners used by CT leads to too little dissipation, and results in an unstable algorithm on some tests when combined with certain unsplit integrators (such as CTU). Moreover, arithmetic averaging destroys the upwinded character of the fluxes in the sense that it does not reduce to the appropriate 1D upwind fluxes for plane-parallel grid-aligned flows. Instead, the relation between the face- and cell-centered magnetic fields implies a relation between the CT and Godunov electric fields. This in turn allows one to show that the method of constructing the CT electric fields has implications for the dissipation and stability of the numerical algorithm. In most circumstances, the best method for constructing the CT electric fields was found to be

$$\begin{aligned}\mathscr{E}_{z,i-1/2,j-1/2} = {} & \frac{1}{4}(\mathscr{E}_{z,i-1/2,j} + \mathscr{E}_{z,i-1/2,j-1} + \mathscr{E}_{z,i,j-1/2} + \mathscr{E}_{z,i-1,j-1/2}) \\ & + \frac{\delta y}{8}\left(\left(\frac{\partial \mathscr{E}_z}{\partial y}\right)_{i-1/2,j-1/4} - \left(\frac{\partial \mathscr{E}_z}{\partial y}\right)_{i-1/2,j-3/4}\right) \\ & + \frac{\delta x}{8}\left(\left(\frac{\partial \mathscr{E}_z}{\partial x}\right)_{i-1/4,j-1/2} - \left(\frac{\partial \mathscr{E}_z}{\partial x}\right)_{i-3/4,j-1/2}\right), \end{aligned} \tag{22}$$

where the derivative of $\mathscr{E}_z$ on each grid cell face is computed by selecting the "upwind" direction according to the contact mode, e.g.

$$\left(\frac{\partial \mathscr{E}_z}{\partial y}\right)_{i-1/2} = \begin{cases} (\partial \mathscr{E}_z/\partial y)_{i-1} & \text{for } v_{x,i-1/2} > 0, \\ (\partial \mathscr{E}_z/\partial y)_{i} & \text{for } v_{x,i-1/2} < 0, \\ \frac{1}{2}\left(\left(\frac{\partial \mathscr{E}_z}{\partial y}\right)_{i-1} + \left(\frac{\partial \mathscr{E}_z}{\partial y}\right)_{i}\right) & \text{otherwise} \end{cases} \tag{23}$$

(where the subscript j has been suppressed) with an analogous expression for the $(\partial \mathscr{E}_z/\partial x)$, where

$$\left(\frac{\partial \mathscr{E}_z}{\partial y}\right)_{i,j-1/4} = 2\left(\frac{\mathscr{E}^r_{z,i,j} - \mathscr{E}_{z,i,j-1/2}}{\delta y}\right). \tag{24}$$

The cell center reference electric field $\mathscr{E}^r_{z,i,j}$ appearing in the above expression is computed at the same time level as the face-centered electric field (Godunov fluxes). Fig. 5 in S08 clarifies the relative location between the Godunov fluxes, corner-centered emf, cell-centered reference states, and the derivatives of the electric field. In 3D analogous expressions to the above are required to relate the x- and y-components of the electric field to the appropriate cell corners (see Fig. 1). These expressions follow directly from Eqs. (23) and (24) using a cyclic permutation of the (x, y, z) and (i, j, k).

5. Three-dimensional integration algorithm

The VL + CT algorithm can now be summarized by the following sequence of steps.

Step 1. Using a Riemann solver, construct first-order upwind fluxes

$$\mathbf{F}^*_{i-1/2,j,k} = \mathscr{F}(\mathbf{U}_{i-1,j,k}, \mathbf{U}_{i,j,k}), \tag{25}$$

$$\mathbf{G}^*_{i,j-1/2,k} = \mathscr{F}(\mathbf{U}_{i,j-1,k}, \mathbf{U}_{i,j,k}), \tag{26}$$

$$\mathbf{H}^*_{i,j,k-1/2} = \mathscr{F}(\mathbf{U}_{i,j,k-1}, \mathbf{U}_{i,j,k}), \tag{27}$$

where the longitudinal component of the magnetic field in each vector, $\mathbf{U}_{i,j,k}$ etc., is set equal to the face-centered value at each interface.

Step 2. Apply the algorithm of Section 4.3 to calculate the CT electric fields at cell-corners, $\mathscr{E}^*_{x,i,j-1/2,k-1/2}, \mathscr{E}^*_{y,i-1/2,j,k-1/2}$ and $\mathscr{E}^*_{z,i-1/2,j-1/2,k}$, from the appropriate components of the face-centered fluxes returned by the Riemann solver in step 1, and a cell center reference electric field calculated using the initial data at time level n, i.e. $\mathscr{E}^{r,n}_{i,j,k} = -(\mathbf{v}^n_{i,j,k} \times \mathbf{B}^n_{i,j,k})$.

Step 3. Update the cell-centered hydrodynamical variables for one-half time step $(\delta t/2)$ using flux differences in all three-dimensions, Eq. (6). Update the face-centered components of the magnetic field for one-half time step using CT (Eqs. (10) through (12)).

Step 4. Compute the cell-centered magnetic field at the half-time step from the average of the face centered field computed in Step 4, using Eqs. (15) through (17).

Step 5. Using the second-order (piecewise linear) reconstruction algorithm described in Section 4.2, compute the left- and right-state quantities at the half time step at cell interfaces in the x-direction $(\mathbf{U}^{n+1/2}_{L,i-1/2,j,k}, \mathbf{U}^{n+1/2}_{R,i-1/2,j,k})$, the y-direction $(\mathbf{U}^{n+1/2}_{L,i,j-1/2,k}, \mathbf{U}^{n+1/2}_{R,i,j-1/2,k})$, and the z-direction $(\mathbf{U}^{n+1/2}_{L,i,j,k-1/2}, \mathbf{U}^{n+1/2}_{R,i,j,k-1/2})$ simultaneously.

Step 6. Using a Riemann solver, construct 1D fluxes at interfaces in all three dimensions

$$\mathbf{F}^{n+1/2}_{i-1/2,j,k} = \mathscr{F}(\mathbf{U}^{n+1/2}_{L,i-1/2,j,k}, \mathbf{U}^{n+1/2}_{R,i-1/2,j,k}), \tag{28}$$

$$\mathbf{G}^{n+1/2}_{i,j-1/2,k} = \mathscr{F}(\mathbf{U}^{n+1/2}_{L,i,j-1/2,k}, \mathbf{U}^{n+1/2}_{R,i,j-1/2,k}), \tag{29}$$

$$\mathbf{H}^{n+1/2}_{i,j,k-1/2} = \mathscr{F}(\mathbf{U}^{n+1/2}_{L,i,j,k-1/2}, \mathbf{U}^{n+1/2}_{R,i,j,k-1/2}), \tag{30}$$

where the longitudinal component of the magnetic field in the vector of left and right states is set equal to the face-centered value at each interface.

Step 7. Compute a cell-centered reference electric field at $t^{n+1/2}$ using the cell-centered velocities and magnetic field at the half timestep computed in steps 3 and 4. Then, apply the algorithm of Section 4.3 to calculate the CT electric fields at cell-corners, $\mathscr{E}^{n+1/2}_{x,i,j-1/2,k-1/2}, \mathscr{E}^{n+1/2}_{y,i-1/2,j,k-1/2}$ and $\mathscr{E}^{n+1/2}_{z,i-1/2,j-1/2,k}$, from the appropriate components of the face-centered fluxes returned by the Riemann solver in step 6 and this reference field.

Step 8. Update the cell-centered hydrodynamical variables for a full timestep, using Eq. (6) and the fluxes from step 6. Update the face-centered components of the magnetic field for full time step using CT (Eqs. (10) through (12)) and the emfs from step 7.

Step 9. Compute the cell-centered components of the magnetic field from the updated face-centered values using Eqs. (15) through (17).

Step 10. Increment the time: $t^{n+1} = t^n + \delta t$. Compute a new time-step that satisfies an estimate of the CFL stability condition based on wavespeeds at cell centers

$$\delta t = C_\circ \min\left(\frac{\delta x}{|v^{n+1}_{x,i,j,k}| + C^{n+1}_{f,i,j,k}}, \frac{\delta y}{|v^{n+1}_{y,i,j,k}| + C^{n+1}_{f,i,j,k}}, \frac{\delta z}{|v^{n+1}_{z,i,j,k}| + C^{n+1}_{f,i,j,k}}\right), \tag{31}$$

where $C_\circ \leqslant 1/2$ is the CFL number, and the denominator of each quantity in the minima is the maximum wavespeed normal to the interface in each dimension (C^{n+1}_f denotes the fast magnetosonic speed evaluated using the updated cell-centered quantities). The minimum is taken over all grid cells. Note this is only an estimate of the CFL stability condition, since the wavespeeds used in the Riemann solver can be different from those computed from the cell-centered values.

Step 11. Repeat steps 1–10 until the stopping criterion is reached, i.e., $t^{n+1} \geqslant t_f$.

The algorithm can be summarized by the flow chart shown in Fig. 2. It is instructive to compare these steps to those in the 3D CTU + CT algorithm described in Section 6.1 in S08. It is also instructive to compare these steps to those of the original MUSCL–Hancock method (van Albada et al., 1982; van Leer, 2006). The primary difference is in the predict step. In the MUSCL–Hancock method, the predict step consists of first updating the cell centered primitive variables by a half time step using a characteristic time advance, and then using linear slopes computed using the old values to reconstruct the variables to the left and right edges of each cell. These left- and right-states are the time averaged values needed by the Riemann solver to compute second-order fluxes needed for the correct step. In the VL + CT scheme described here, the predict step consists of updating the conserved variables using first-order fluxes, and then second-order interface states are computed by spatial reconstruction of these values.

6. Tests

In this section, we show the results from a series of 3D MHD tests of the VL + CT algorithm, using the problems described in S08. The results can be compared directly to those for the CTU + CT integrator shown in S08. In every case the solution has been computed using the HLLD Riemann solver.

6.1. Linear waves

The propagation of linear amplitude, planar waves in a direction which is oblique to the grid provides a quantitative measure of the diffusion and dispersion error for each MHD wave mode. The test uses a computational domain of size $2L \times L \times L$, with $L = 1.5$, and periodic boundary conditions in each direction. Solutions are computed using a grid with $2N \times N \times N$ cells, with N varying from 8 to 128. The background medium is uniform with density $\rho_0 = 1$, pressure $P_0 = 3/5$, and $\gamma = 5/3$. It is easiest to define the magnetic field in a coordinate system in which the 1-direction is aligned with the wavevector, in these coordinates $(B_1, B_2, B_3) = (1, \sqrt{2}, 1/2)$. For the entropy mode the longitudinal component of the velocity is $v_1 = 1$, for the other three modes the background medium is at rest. These choices give $C_f = 2, C_{A,x} = 1$, and $C_s = 1/2$ for the fast, Alfvén, and slow magnetosonic speeds, respectively. Exact eigenfunctions for fast and slow magnetosonic, Alfvén, and contact waves are initialized; the eigenvectors for each mode are cataloged in GS08. The coordinate rotations required to initialize the wave defined in the (x_1, x_2, x_3) coordinate system at an oblique angle to the computational grid are given in GS08. By inclining the wavevector to the grid, we ensure this test is inherently multidimensional, with no symmetries in any direction. For each mode, the wave is allowed to propagate one wavelength, and then the L_1 error with respect to the initial conditions is computed from

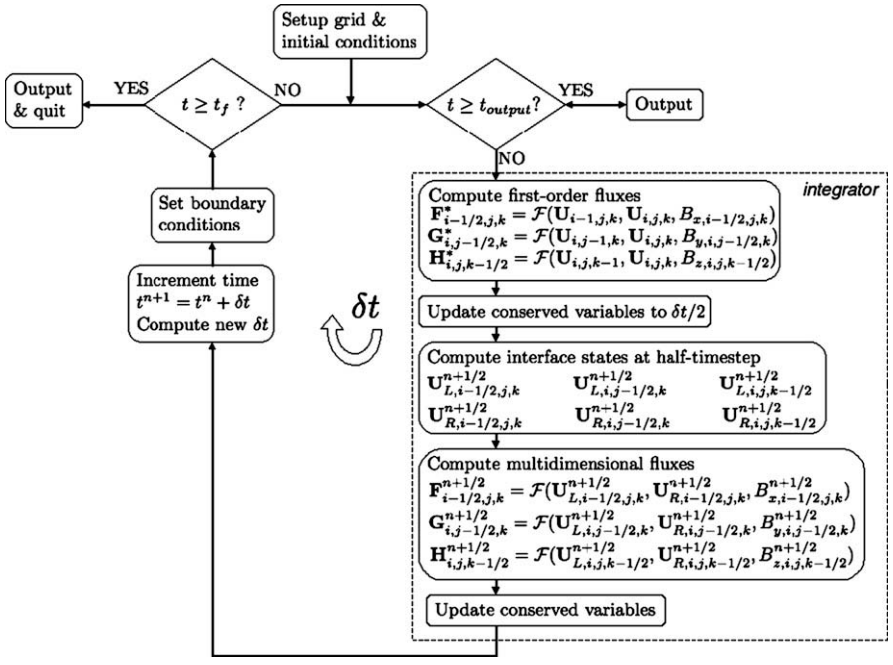

Fig. 2. Schematic flow chart for the 3D VL + CT integrator.

$$\delta \mathbf{U}^n = \frac{1}{N^3} \sum_{i,j,k} | \mathbf{U}^n_{i,j,k} - \mathbf{U}^0_{i,j,k} |, \tag{32}$$

where $\mathbf{U}^0_{i,j,k}$ is the initial solution on the grid. The norm of the L_1 error vector is a useful quantitative measure of the global error in the solution. Results for this identical test computed with the CTU + CT algorithm are given in GS05 in 2D, GS08 in 3D, and S08.

Fig. 3 plots the norm of the L_1 error vector for each of the four MHD wave modes as a function of the numerical resolution N computed using the VL + CT algorithm as a solid line. For direct comparison, the error in the solution computed with the CTU + CT algorithm computed with second-order reconstruction is shown as a dashed line. The errors in both methods are very close, with the largest discrepancy in the Alfvén mode. The error converges at second-order for all modes. Note that lower errors are possible using the CTU + CT method and third-order reconstruction (see S08 Fig. 32). For example, at $n = 128$ the L_1 error using the third-order CTU + CT method is a factor of 1.4 lower for the fast wave, 3 lower for the Alfvén wave, 1.4 lower for the slow wave, and 1.5 lower for the entropy wave.

6.2. Circularly polarized alfven wave

Circularly polarized Alfven waves are an exact solution to the ideal MHD equations, even if they have nonlinear amplitude. Following T2000, we initialize the wave in a uniform medium at rest with $\rho_0 = 1, P_0 = 0.1, \gamma = 5/3$. The longitudinal component of the velocity and magnetic field are $v_1 = 0$ and $B_1 = 1$, respectively. The transverse components of the velocity and magnetic field in this coordinate system are then $(v_2, v_3) = (B_2, B_3) = (A \sin(2\pi x_1/\lambda), A \cos(2\pi x_1/\lambda))$, with $A = 0.1$. As in the linear wave convergence test, the computational domain is of size $2L \times L \times L$, with $L = 1.5$, and periodic boundary conditions in each direction. Solutions are computed using a grid with $2N \times N \times N$ cells, with N varying from 8 to 128. The coordinate rotations used to incline the planar wave solution at an oblique angle to the computational grid are the same as those used for the linear wave convergence test described above (see GS08 for details).

Fig. 4 plots the profile of the transverse component of the magnetic field which, for the particular coordinate rotations adopted for this test, is given by

$$B_2 = (B_y - 2B_x)/\sqrt{5} \tag{33}$$

for resolutions with N from 8 to 64 for traveling waves after propagating five wavelengths. The profiles are dominated by diffusion rather than dispersion error. For $N \geqslant 32$, the wave amplitude is at least 0.9 of the original. Also plotted in Fig. 4 is the norm of the L_1 error vector as a function of resolution after the wave has propagated one wavelength. For comparison, the error in the CTU + CT algorithm, computed using second-order reconstruction, is shown as a dotted line. Fig. 33 of S08 plots the error in the CTU + CT algorithm computed using third-order reconstruction. Comparison of these two figures shows that the third-order CTU + CT algorithm has significantly less diffusion than either the second-order VL + CT or CTU + CT methods; at $N = 8$ the L_1 error is more than a factor of two smaller using the third-order CTU + CT method. This result, when combined with the results from the previous test, indicates the third-order CTU + CT integrator is less diffusive than the VL + CT method described here. It is also instructive to investigate how the numerical damping of Alfvén waves depends on the wave amplitude. Comparison of the second panel in Figs. 3 and 4 demonstrates that when normalized by the initial amplitude, the errors are nearly identical for both linear and nonlinear waves, and that the error converges at second order in both cases. Thus, for smooth solutions, the rate of damping is independent of the initial amplitude, and depends only on resolution.

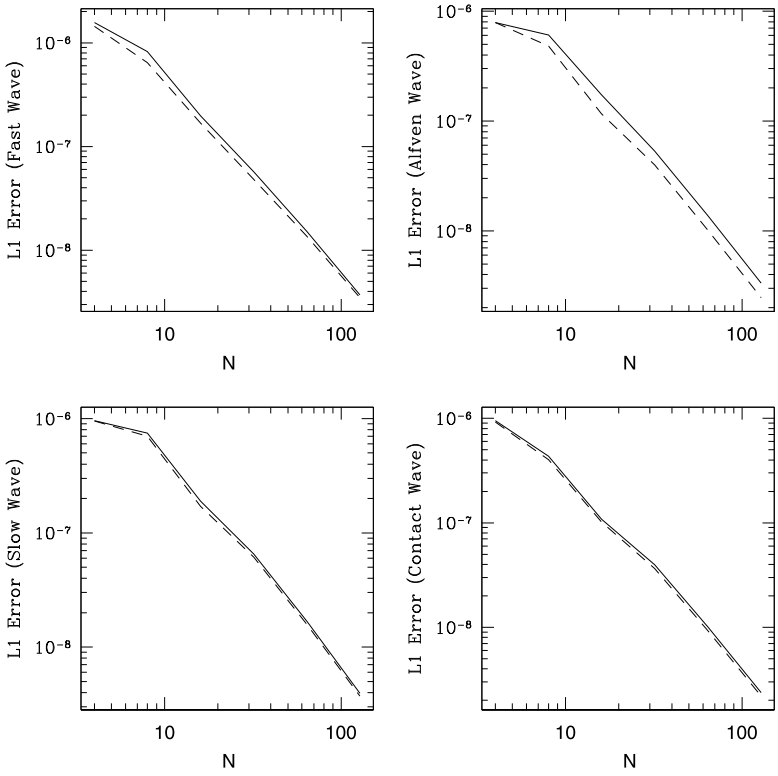

Fig. 3. Convergence of the norm of the L_1 error vector with numerical resolution N for propagation of linear modes of each MHD wave family a distance equal to one wavelength, computed on a 3D grid with resolution $2N \times N \times N$. The solid line shows the errors for the VL + CT integrator, the dashed line shows errors for the CTU + CT integrator with second-order reconstruction. Both converge at second-order.

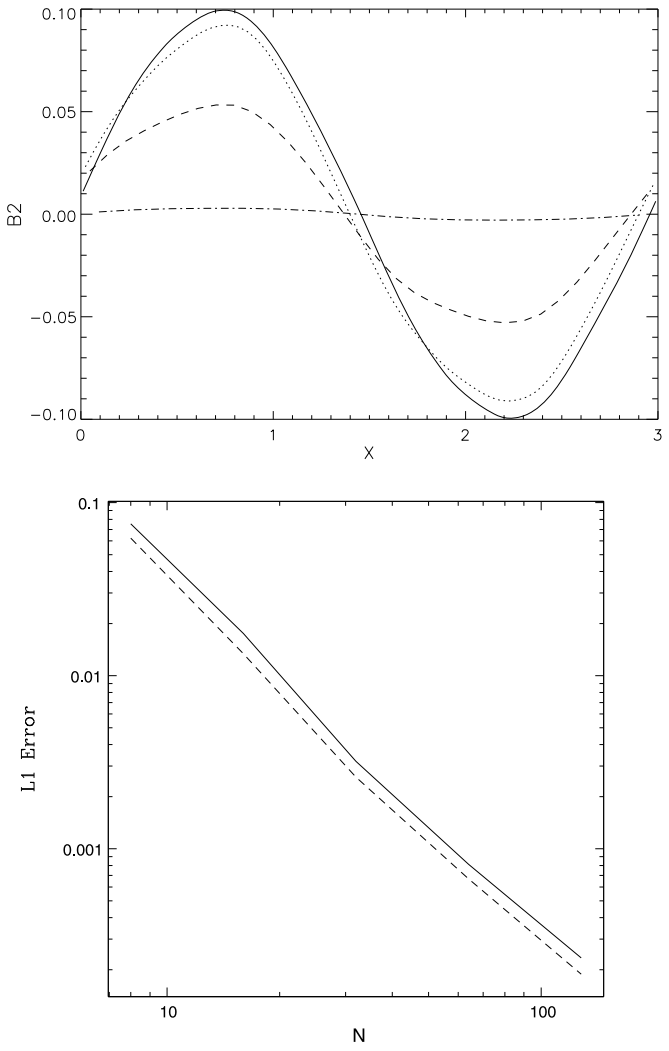

Fig. 4. (Top) Profile of the transverse component of the magnetic field in a nonlinear circularly polarized Alfvén wave after propagating a distance of five wavelengths in a 3D grid with resolution $2N \times N \times N$. The solid line shows $N = 64$, the dotted $N = 32$, the dashed $N = 16$, and the dot-dash $N = 8$. (Bottom) Convergence of the norm of the L_1 error vector after propagating one wavelength. The solid line is for the VL + CT integrator. and the dashed for the CTU + CT integrator with second-order reconstruction. Both converge at second-order.

6.3. Field loop test

The advection of a loop of weak magnetic field has proved to be a sensitive test of the multidimensional integration algorithm for the induction equation. In particular, we have used this test in both GS05 and GS08 to explore methods for constructing the corner-centered emfs from the face-centered fluxes returned by the Riemann solver. In GS08, we also described how this test could be used to investigate whether the numerical algorithm preserves the appropriate discrete representation of $\nabla \cdot \mathbf{B} = 0$. If a field loop in the $x - y$ plane (so that $B_z = 0$) is advected with a uniform 3D velocity field (in particular, with $v_z \neq 0$), then the induction equation for the z-component of the magnetic field reduces to $\partial B_z/\partial t = v_z(\partial B_x/\partial x + \partial B_y/\partial y) = 0$. Thus, B_z will remain exactly zero only if the divergence-free constraint is maintained exactly by the method.

We have performed two versions of this test for the VL + CT method. In the first, the computational domain is of size $2L \times L \times L$, with $L = 1$ and periodic boundary conditions everywhere. The density and pressure are uniform with value unity, and $\gamma = 5/3$.

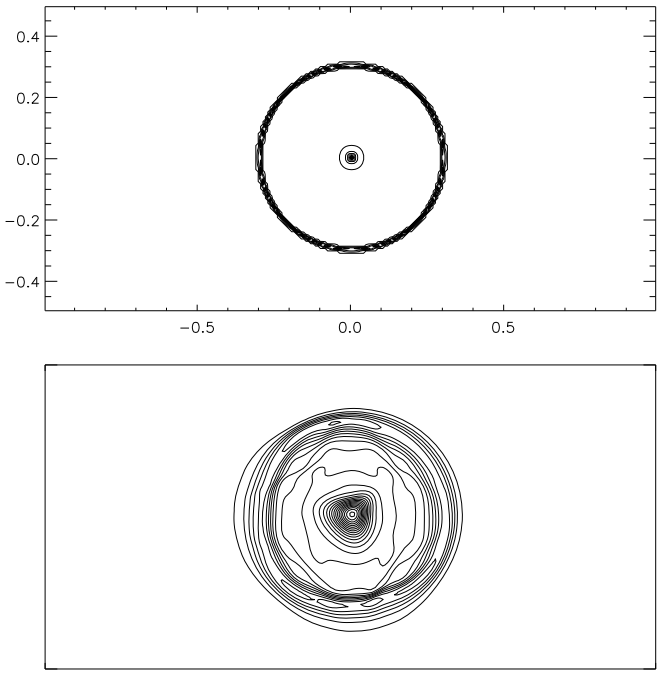

Fig. 5. Contours of the z-component of the current density in the initial conditions (top), and after advection twice around the grid (bottom) for a weak field loop. The contours are plotted for an arbitrary slice in the $x - y$ plane, on a grid with resolution of $256 \times 128 \times 128$.

The velocity is uniform with $(v_x, v_y, v_z) = (2, 1, 1)/\sqrt{6}$. A field loop in the $x - y$ plane is initialized using the vector potential given in GS05; this is a 3D version of the 2D test described there. Fig. 5 plots contours of the out-of-plane component of the current density $\mathbf{J} = \nabla \times \mathbf{B}$ after advection of the loop twice around the domain. The current density is extremely sensitive to oscillations in the field. The loop contains a current sheet at the surface, and a singular current spike at the center. Note the VL + CT method keeps the contours of both of these features smooth and nearly symmetric, despite the lack of any symmetries in the problem. In fact, since the loop has a small but finite magnetic pressure, we expect some evolution in MHD. The contours can be compared to those shown in Fig. 21 in S08 for the CTU + CT method. There is excellent agreement. We have confirmed that the VL + CT preserves $B_z = 0$ to machine round-off error in this test. Thus, although there are many stencils on which the divergence-free constraint can be maintained, this test demonstrates the VL + CT algorithm preserves the constraint on the appropriate stencil to prevent anomalous growth of the field.

In the second test, the computational domain has size $L \times L \times L$, with $L = 1$, and the grid has a resolution of 128^3. Periodic boundary conditions are used in all dimensions. The density and pressure are one, $\gamma = 5/3$, and three components of the velocity are equal to $1/\sqrt{3}$. The magnetic field is initialized using the vector potential given in SG08, it corresponds to a cylindrical column of field loops inclined at an oblique angle to the mesh. Fig. 6 shows the magnetic energy at $t = 1$, after the column has been advected around the grid once. The figure shows the column maintains its structure, and there is no evidence for oscillation or dispersion. In this test the component of the field along the axis of the cylinder should remain zero. We have confirmed the VL + CT method keeps this component zero to within truncation error.

6.4. Three-dimensional shocktube

One-dimensional Riemann problems are a popular test of the ability of a scheme to capture shocks, rarefactions, and contact

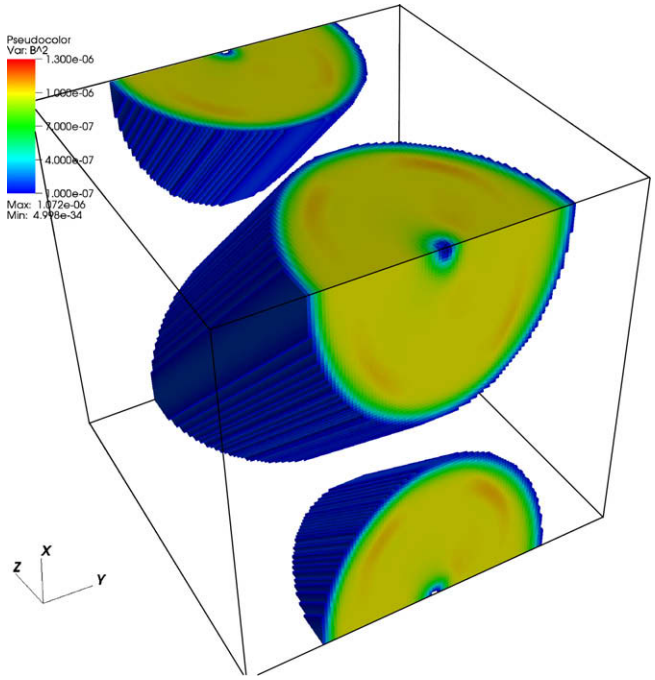

Fig. 6. Image of the magnetic energy density B^2 for the advection of a cylinder of weak magnetic field loops inclined to the grid, after advection once around the grid, computed on a 128^3 mesh.

and rotational discontinuities. For MHD, 1D tests are too restrictive; they cannot reveal errors associated with the development of a non-solenoidal magnetic field. To extend shocktube problems to multidimensions, the 1D problem can simply be rotated at an arbitrary angle to the mesh. Incorrect jump conditions can result, or the longitudinal component of the magnetic field may not be kept constant, if the method does not enforce the divergence-free constraint (Falle et al., 1998, T2000).

While a wide variety of Riemann problems are suitable as a test, we present the results for the problem shown in Fig. 2a in Ryu and Jones (1995, hereafter RJ). The left- and right-states that define this problem are given in RJ, and in Table 2 in S08. Initializing these states separated by a discontinuity inclined to the grid in a way which minimizes the generation of noise is not trivial; the method we use is described in detail in GS08. The calculation is performed on a grid of size $(L_x, L_y, L_z) = (3/2, 1/64, 1/64)$, with a resolution of $768 \times 8 \times 8$ cells.

Fig. 7 plots the solution at $t = 0.2$. Direct comparison can be made to Fig. 35 in S08 (computed using second-order reconstruction in the CTU + CT algorithm). The solution computed in 1D is also shown in Fig. 14 of S08. The test is of interest because it develops discontinuities in each MHD wave mode, that is fast and slow magnetosonic shocks, rotational discontinuities, and a contact discontinuity. Each of these is captured well in the solution, indeed the profiles are nearly indistinguishable from the purely 1D solution.

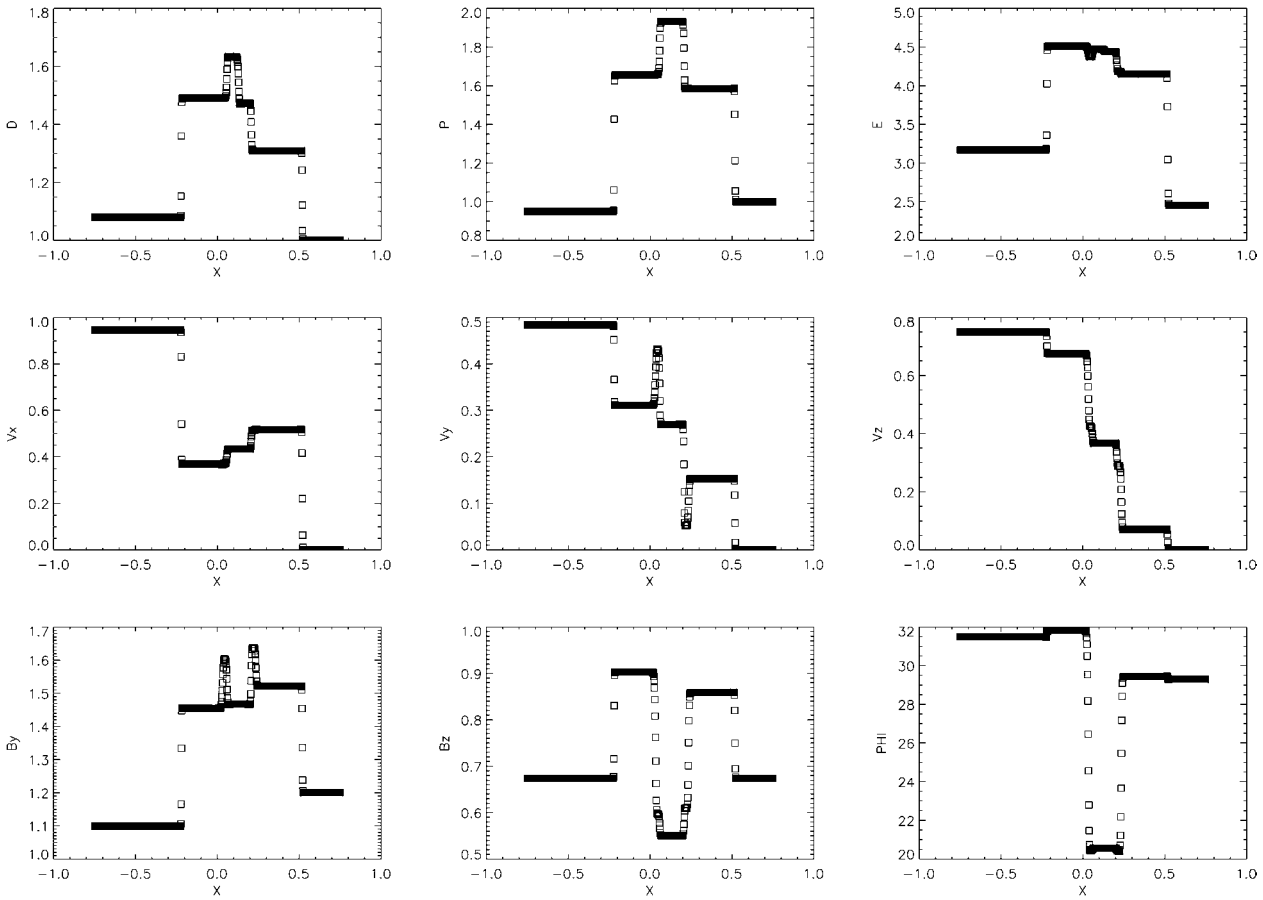

Fig. 7. One dimensional slice of selected variables at $t = 0.2$ from the Riemann problem labeled as 2a from RJ. The calculation is fully 3D with the initial interface inclined at an oblique angle to the mesh.

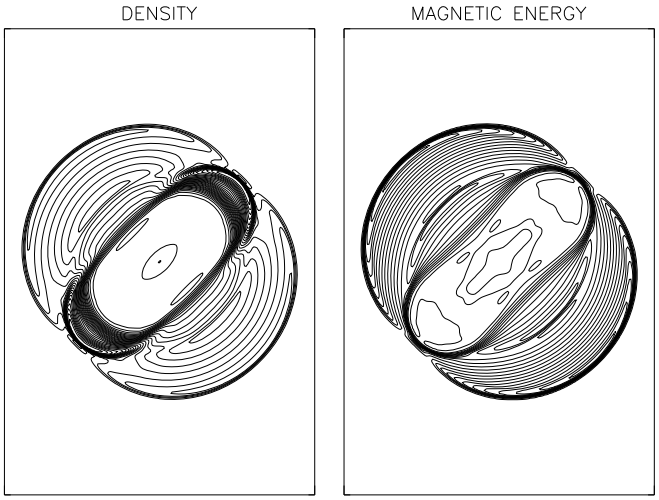

Fig. 8. Contours of the density and magnetic pressure at $t = 0.2$ for the evolution of a spherical blast wave in a strongly magnetized medium in 3D. The contours are shown on an $x - y$ slice taken through the center of the mesh. Thirty equally spaced contours are used between the minimum and maximum of each.

6.5. Blast wave

As a final test of the method, to demonstrate it can handle strong shocks and rarefactions in 3D, we present the evolution of a strong blast wave in a uniform, magnetized medium (this test is also presented in GS08 and S08). The computational domain has size $L \times 1.5L \times L$ with $L = 1$, and a resolution of $200 \times 300 \times 200$. The initial density $\rho_0 = 1$, pressure $P_0 = 0.1$, magnetic field $\mathbf{B}_0 = (1/\sqrt{2}, 1/\sqrt{2}, 0)$, and velocity $\mathbf{v}_0 = 0$. These values give $\beta = 2P_0/B_0^2 = 0.2$ in the background medium. In a region of radius $R = 0.1$, the initial pressure is set to 100. This generates a strong circular blast wave which propagates outward, and a rarefaction which propagates inward, evacuating the overpressurized region. The Mach number of the outward moving shock is about 5. Fig. 8 shows contours of the density and magnetic pressure at $t = 0.2$ (just before the blast wave leaves the domain) on a slice in the $x - y$ plane taken through the center of the grid. The contours are smooth and symmetric. The inner low density bubble is clearly collimated by the strong field into an elliptical region with long axis parallel to the background field. Perpendicular to the field lines, the blast wave propagates rapidly, and there is a large separation between it and the contact discontinuity that bound the inner elliptical bubble. Parallel to the field the blast wave and contact are spaced much more closely. The solution computed with the VL + CT integrator is nearly identical to that shown in Fig. 36 in S08 computed using the CTU + CT method. We have found that increasing the strength of the shock (by increasing the initial pressure in $r < R$) makes little difference to the solution. Moreover, the stability of the VL + CT integrator is independent of the strength of the blast wave. However, we find increasing the magnetic field strength in the ambient gas by a factor of ten (to $\beta = 0.002$) causes the integrator to crash at late times ($t > 0.05$) if an adiabatic equation of state is used. The algorithm runs stably even for such low β with an isothermal equation of state. This behavior is consistent with the well known difficulties associated with integration of the total (rather then internal) energy equation at very low β.

7. Conclusion

We have described a dimensionally unsplit integrator for MHD which is based on: (1) a predictor–corrector algorithm (Falle, 1991) similar to the MUSL–Hancock scheme described by van Leer (2006); Toro, 1999; see also the discussion at the end of Section 5, and (2) the CT algorithm of Evans and Hawley (1988) to preserve the divergence-free constraint. The algorithm is simple. It does not require a characteristic evolution in the reconstruction step, nor does it require the addition of source terms in multidimensional MHD that arise from using directionally-split updates to compute quantities at fractional time steps. Such terms were shown to be necessary for the CTU integrator for MHD in SG05 and SG08. Since the method is so simple, it is easy to extend with additional physics, as pointed out by Falle (1991). For example, source terms can be added at second-order by applying them at the half time-step in the predictor (directly after step 4 described in Section 5). Similarly, since a characteristic decomposition is not required in the reconstruction step, it is easy to extend the method to systems in which the eigenvectors are complex, for example special relativistic MHD.

An important feature of the method is the use of CT to evolve the magnetic field. One disadvantage of the version of CT used here is that it requires a staggered grid for the magnetic field, which complicates the implementation and the interface to AMR drivers. On the other hand, since elliptic solvers are not required for divergence-cleaning, the method is computationally efficient, and scales extremely well on parallel processors. Moreover, we have shown through 3D tests of the advection of weak field loops that the staggered grid formulation of CT prevents spurious growth of magnetic field, and presumably anomalous forces associated with magnetic monopoles, which arise if the constraint is not maintained on the same stencil as is used to compute fluxes. Often it is not clear what is the appropriate discrete representation of the constraint that must be preserved. Our tests demonstrate that the VL + CT integrator maintains the constraint on the appropriate stencil. This test may be useful for other methods.

The VL + CT integrator is implemented as one of two distinct multidimensional unsplit integrators in the C version of the Athena code, which is described in more detail in S08. A simple command line flag to the `configure` script is used to choose between the VL + CT method, or the more complex CTU + CT method described in S08. In this paper, we have presented a series of 3D tests of the VL + CT method, and made direct comparison to the results computed with CTU + CT. Since the identical code is used for the tests (using, for example, the identical Riemann solver), differences in the solutions can only be related to the integrators themselves. We find the VL + CT method converges on smooth solutions of each MHD wave family at second order, and it captures shocks, contact and rotational discontinuities accurately. However, the method is also measurably more diffusive than the CTU + CT method, by about a factor of 1.5–2 for most tests of smooth flow. The VL + CT integrator is not substantially faster than CTU + CT in terms of cpu time (about 20% faster for adiabatic MHD), since it requires the same number of Riemann solves and spatial reconstruction steps per cell as CTU + CT. Since the latter is more accurate, we consider the CTU + CT algorithm the preferred method for most MHD calculations with Athena. On the other hand, we have found the VL + CT method to be more robust in extreme parameter regimes, for example in strongly supersonic MHD turbulence (Lemaster and Stone, in press). In particular, for a CFL number $C_\circ \leqslant 1/3$ and first-order fluxes, it can be shown that the VL + CT method is positive definite. This property can be exploited to build a robust integration step for cells where more accurate, higher-order integration fails. Thus, we have used the VL + CT integrator for recent studies of supersonic MHD turbulence that strain the stability of the methods (Lemaster and Stone, in press). Probably the biggest advantage of the VL + CT integrator is its simplicity. As implemented in Athena, it requires only about one half the number of lines of code as CTU + CT. Thus, it is often our integrator of choice when extending Athena with more complex physics.

The VL + CT integrator can easily be combined with AMR, and therefore we expect it may be useful for many existing code frameworks. It is hoped that the description and tests of the algorithm given in this paper will be useful to others wishing to implement a simple and robust Godunov method for MHD, and that the method will prove useful for the study of a wide variety of problems in astrophysical fluid dynamics.

Acknowledgements

We thank Mike Norman for his encouragement to publish this method, and the referee (Steve Balbus) for suggestions that improved the paper. Simulations were performed on the Teragrid cluster at NCSA, the IBM Blue Gene at Princeton University, and on computational facilities supported by NSF Grants AST-0216105 and AST-0722479. Financial support from DoE Grant DE-FG52-06NA26217 is acknowledged.

References

Balbus, S.A., 2003. ARA&A 41, 555.
Balsara, D.S., Spicer, D.S., 1999. J. Comput. Phys. 149, 270.
Batten, P., Clarke, N., Lambert, C., Causon, D.M., 1997. SIAM J. Sci. Comput. 18, 1553.
Cargo, P., Gallice, G., 1997. J. Comput. Phys. 136, 446.
Colella, P., 1990. J. Comput. Phys. 87, 171.
Colella, P., Woodward, P.R., 1984. J. Comput. Phys. 54, 174.
Crockett, R.K., Colella, P., Fisher, R.T., Klein, R.I., McKee, C.F., 2005. J. Comput. Phys. 203, 255.
Cunningham, A.J., Frank, A., Varniere, P., Mitran, S., Jones, T.W., 2007. Available from: <astro-ph:0710.0424>.
Dai, W., Woodward, P.R., 1998. J. Comput. Phys. 142, 331.
Dedner, A., Kemm, F., Kröner, D., Munz, C.-T., Schnitzer, T., Wesenberg, M.J., 2002. Comput. Phys. 175, 645.
de Gouveia Dal Pino, E., 2005. Adv. Sp. Res. 35, 908.
Einfeldt, B., Munz, C.D., Roe, P.L., SjÖgreen, B.J., 1991. J. Comput. Phys. 92, 273.
Evans, C.R., Hawley, J.F., 1988. ApJ 322, 659.
Falle, S.A.E.G., 1991. MNRAS 250, 581.
Falle, S.A.E.G., Komissarov, S.S., Joarder, P., 1998. MNRAS 297, 26.
Fromang, S., Hennebelle, P., Teyssier, R., 2006. A&A 457, 371.
Gardiner, T., Stone, J.M., 2005. J. Comput. Phys. 205, 509. GS05.
Gardiner, T., Stone, J.M., 2008. J. Comput. Phys. 227, 4123. GS08.
Harten, A., Lax, P.D., van Leer, B., 1983. SIAM Rev. 25, 35.
Lemaster, M.N., Stone, J.M., in press. ApJL 682, L97.
LeVeque, R.J., 2002. Finite Volume Methods for Hyperbolic Problems, Cambridge, CUP.
Londrillo, P., Del Zanna, L., 2004. J. Comput. Phys. 195, 17.
McKee, C.F., Ostriker, E.C., 2007. ARA&A 45, 565.
Mignone, A., Bodo, G., Massaglia, S., Matsakos, T., Tesileanu, O., Zanni, C, Ferrari, A., 2007. Available from: <astro-ph/070185>.
Miyoshi, T., Kusano, K., 2005. J. Comput. Phys. 208, 315.
Pen, U.-L., Arras, P., Wong, S., 2003. ApJ 149 (Suppl.), 447.
Powell, K.G., Roe, P.L., Linde, T.J., Gombosi, T.I., de Zeeuw, D.L., 1999. J. Comput. Phys. 153, 284.
Roe, P.L., 1981. J. Comput. Phys. 43, 357.
Ryu, D., Jones, T.W., 1995. ApJ 442, 228. RJ.
Ryu, D., Miniati, F., Jones, T.W., Frank, A., 1998. ApJ 509, 244.
Stone, J.M., Gardiner, T.A., Teuben, P., Hawley, J.F., Simon, J.B., in press. ApJS 178.
Toro, E.F., 1999. Riemann Solvers and Numerical Methods for Fluid Dynamics. Springer-Verlag.
Tóth, G., 2000. J. Comput. Phys. 161, 605. T2000.
van Albada, G.D., van Leer, B., Roberts, J.W.W., 1982. A&A 108, 76.
van Leer, B., 2006. Commun. Comput. Phys. 1, 192.
Ziegler, U., 2004. J. Comput. Phys. 196, 393.