Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 226 (2007) 925–946
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

Hybrid block-AMR in cartesian and curvilinear coordinates: MHD applications

B. van der Holst [a,*], R. Keppens [a,b,c]

[a] Centre for Plasma-Astrophysics, K.U. Leuven, Celestijnenlaan 200B, 3001 Heverlee, Belgium
[b] FOM-Institute for Plasma Physics Rijnhuizen, P.O. Box 1207, 3430 BE Nieuwegein, The Netherlands
[c] Astronomical Institute, Utrecht University, P.O. Box 80000, 3508 TA Utrecht, The Netherlands

Received 28 September 2006; received in revised form 2 May 2007; accepted 3 May 2007
Available online 17 May 2007

Abstract

We present a novel, hybrid block-adaptive scheme for use in solving sets of near-conservation laws in general orthogonal coordinate systems. The adaptive mesh refinement (AMR) scheme is block-based, i.e. individual grids have a pre-fixed number of grid cells, and is implemented for any-dimensionality D. Its 'hybrid' character relaxes the common approach where a block that needs refinement triggers 2^D subblocks when grids are refined by a factor of 2. This introduces 'incomplete families' in the grid hierarchy, but approaches the optimal fit to developing flow features inherent in the original patch-based AMR strategy. Our hybrid block-AMR approach is compared with a patch-based AMR one, both exploiting OpenMP parallelism. The implementation is able to handle general curvilinear coordinates, for which restriction and prolongation formulae are presented along with boundary treatments at 'singular' boundaries. Demonstrative examples cover hydro- and magnetohydrodynamic (MHD) applications, including tests on a 2D polar grid, a 2.5D spherical and a 3D cylindrical configuration. The applications cover classical up to relativistic MHD simulations, of particular relevance for astrophysical magnetized jet and stellar wind studies.

Keywords: Adaptive mesh refinement; Magnetohydrodynamics; Curvilinear coordinates

1. Motivation

Adaptive mesh refinement (AMR) provides a cost-effective means for simulating many complex gas and plasma dynamical phenomena where localized (e.g. in the presence of shock fronts) as well as large scale structures (e.g. tubular magnetic flux configurations) play a decisive role in their temporal evolution. In its original application to shock hydrodynamics in 2D [5], the grid consisted of nested rectangular meshes and the created grid hierarchy was characterized by different-sized patches. This patch-based AMR strategy relies on a fairly sophisticated algorithm to organize a collection of individual grid cells – which were somehow identified to

* Corresponding author. Tel.: +32 16327964; fax: +32 16327026.
E-mail addresses: Bart.vanderHolst@wis.kuleuven.be (B. van der Holst), Rony.Keppens@wis.kuleuven.be (R. Keppens).

0021-9991/$ - see front matter © 2007 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2007.05.007

underresolve a flow feature – into properly nested rectangular patches. Berger [4] laid out the basic steps of such an algorithm, and Berger and Rigoutsos [6] introduced pattern recognition techniques to improve on this early patch-based approach. 3D hydro applications emerged in [3], incompressible magnetohydrodynamics problems in [17], and currently, several AMR implementations exist where the dimensionality and/or the application are in essence free parameters. State-of-the-art packages adhering to this patch-based AMR strategy are documented, e.g. in [8], which includes the CCSE application suite and Chombo, or the SAMRAI package at [33]. In AMRVAC [28], an AMR package complementing the Versatile Advection Code (VAC) [38], we followed a similar patch-based strategy until now. These AMR approaches can in principle allow for the use of a different discretization scheme per level (as, e.g. in AMRVAC), or even the coupling of different physics models across the level hierarchy, as pioneered in the Adaptive Mesh and Algorithm Refinement (AMAR) scheme by Garcia et al. [18].

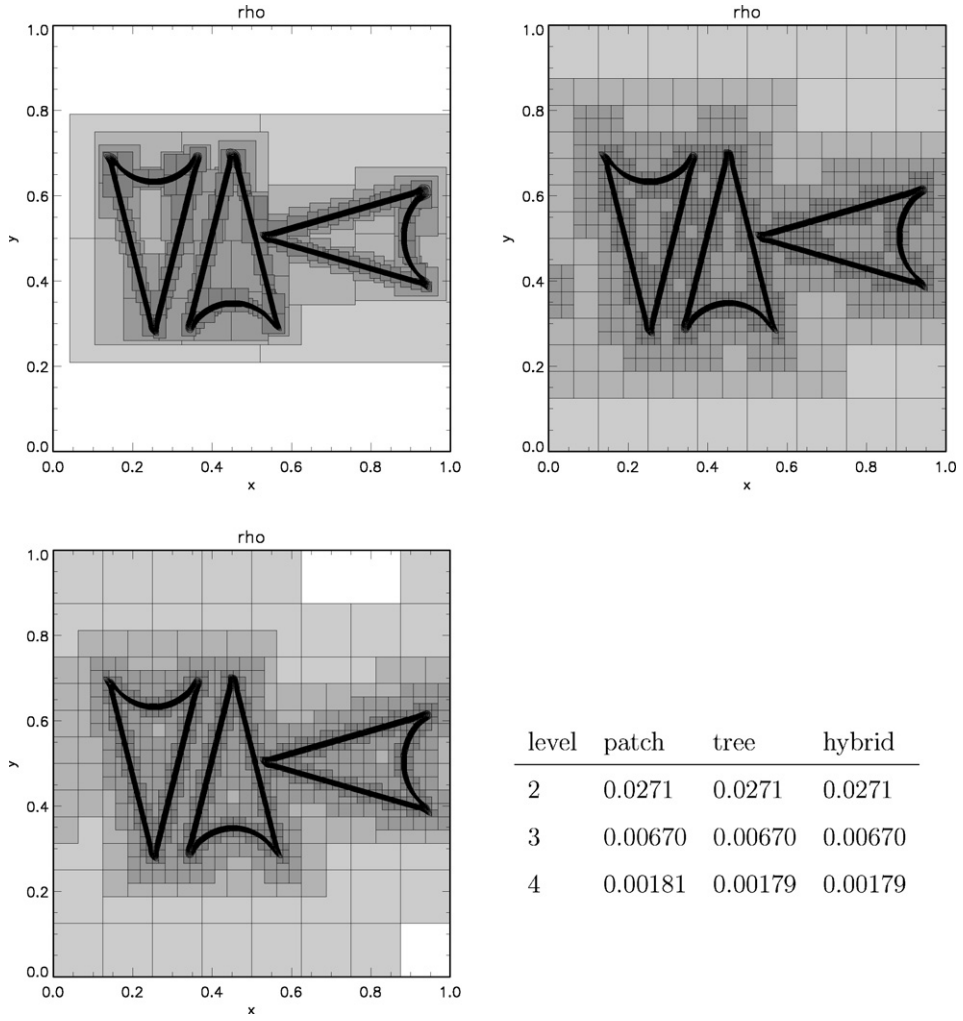
In a good number of AMR frameworks in use today, a block-based AMR technique is exploited, where each grid across the entire grid hierarchy has an equal number of grid cells. The basic data structure is then an octree (for 3D problems), where any block that needs refinement triggers $2^3 = 8$ equal sized blocks when a resolution change of two is assumed. Examples of this approach include the sun-to-earth simulations of space weather phenomena with the BATS-R-US code by Gombosi and coworkers [19], the FLASH code which was validated extensively in [9], the Racoon framework for hyperbolic conservation laws [14] which exploits a hybrid parallelization strategy of POSIX multi-threads and MPI, and the NIRVANA code which has recently been extended with an elliptic solver for handling self-gravitational magnetized plasmas [42]. The advantages of equal block sizes are mostly in the relative ease of creating a properly nested grid structure and the guaranteed identical work load per grid. In this paper, we report on a so-called 'hybrid block-AMR' approach, where the grid size is still constant, but where we relax the full octree structure to allow for incomplete block families (also called 'leaves'). This is meant to incorporate the idea of the patch-based strategy, namely an optimal fit of the grid structure to the dynamical features of interest which was there accomplished at the expense of introducing unequally sized grids per level. On the other hand, the good cache performance of the tree block-based scheme is fully utilized. In contrast to the original Berger approach [4], where grids on a certain grid level l could still overlap, we even modified the patch-based grid approach to eliminate this possibility (as already explained in [28]), and naturally, both hybrid and full octree have this built in: no grid overlap per level. Fig. 1 illustrates both a patch-based, a tree block-based, and a hybrid block-based AMR grid structure exploiting 5 grid levels for a 2D advection test on a doubly periodic domain as introduced in [28]. All three runs are using the same time-stepping per level as the patch-based scheme. On a pentium 4, the timings for this application are 232.6(s) for the patch-based, 195.6(s) for the tree block-based, and 159.2(s) for the hybrid method. The table in Fig. 1 shows the convergence for all three AMR schemes. Here, instead of the discontinuous VAC-logo, we advect a smooth 2D Gaussian bell twice across the doubly periodic domain with a second order numerical scheme and a resolution 48×48 on level 1. This table summarizes the relative error between the end state of the AMR runs for the given maximum allowed refinement level versus the known initial solution. It is clear that all three AMR strategies show a second order convergence behaviour. For more details on the convergence calculations, we refer to [28].

The hybrid block-AMR strategy is made compatible with general curvilinear (orthogonal) coordinate systems, encompassing 2D polar as well as 3D cylindrical and spherical grids. Recent patch-based AMR implementations for Cartesian, cylindrical, and spherical geometries were also demonstrated in [25]. Vector quantities occurring in the equations being solved are then handled in their respective cylindrical or spherical components. We discuss in particular restriction and prolongation formulae suited for curvilinear coordinates, and how we handle boundary conditions at the cylinder axis or the spherical poles in a 'π-periodic' manner.

The applications cover multi-dimensional hydro- (HD), magnetohydrodynamic (MHD) as well as special relativistic magnetohydrodynamic computations. Tests are selected which clearly benefit from AMR strategies. The governing equations can be written as

$$\partial_t \mathbf{U} + \nabla \cdot \mathbf{F}(\mathbf{U}) = \mathbf{S}(\mathbf{U}, \partial_i \mathbf{U}, \partial_i \partial_j \mathbf{U}, \mathbf{x}, t), \tag{1}$$

where $\mathbf{U}$ denotes the vector of conserved quantities, while $\mathbf{F}$ and $\mathbf{S}$ indicate fluxes and source terms, respectively. For the classical Euler and magnetohydrodynamic applications, we have the usual



level	patch	tree	hybrid
2	0.0271	0.0271	0.0271
3	0.00670	0.00670	0.00670
4	0.00181	0.00179	0.00179

Fig. 1. A comparison of the hierarchical grid structure for a patch-based (top-left), a tree block-based (top-right), and a hybrid block-based (bottom-left) approach, exploiting 5 grid levels at the same effective resolution of 768×768. The problem considers the diagonal advection of a 2D 'logo' with constant density 2 within, and 0.5 external to the 'VAC' symbol, and is shown after a full cycle across the diagonal. On the bottom-right panel, the convergence is shown for a double periodic advection of a 2D Gaussian bell versus maximum allowed refinement levels. This indicates that all three AMR strategies remain second order.

$$\mathbf{U} = (\rho, \rho\mathbf{v}, e, \mathbf{B})^{\mathrm{T}}, \tag{2}$$

$$\mathbf{F} = (\mathbf{v}\rho, \mathbf{v}\rho\mathbf{v} - \mathbf{B}\mathbf{B} + Ip_{\mathrm{total}}, \mathbf{v}e + \mathbf{v}p_{\mathrm{total}} - \mathbf{B}\mathbf{B}\cdot\mathbf{v}, \mathbf{v}\mathbf{B} - \mathbf{B}\mathbf{v})^{\mathrm{T}}, \tag{3}$$

with density ρ, velocity $\mathbf{v}$, energy e, magnetic field $\mathbf{B}$ (not for HD) and the total (gas p plus magnetic) pressure p_{total} is for an ideal gas law with ratio of specific heats γ such that

$$p_{\mathrm{total}} = (\gamma - 1)\left(e - \frac{1}{2}\rho\mathbf{v}^2 - \frac{1}{2}\mathbf{B}^2\right) + \frac{1}{2}\mathbf{B}^2 = p + \frac{1}{2}\mathbf{B}^2. \tag{4}$$

In the special relativistic MHD simulations, one still has the set of primitive variables $(\rho, \mathbf{v}, p, \mathbf{B})$, but now the thermodynamic quantities (density and pressure) are measured in the local rest frame (proper rest mass density and pressure). To obtain a $3+1$ split as inherent in system (1), a fixed Lorentzian frame of reference is taken in which the fluid spatial three-velocity is $\mathbf{v}$ with magnetic field $\mathbf{B}$. Introducing the Lorentz factor $\Gamma = (1 - \mathbf{v}^2)^{-1/2}$ (and measuring in a scaling where the light speed $c = 1$), the conserved variables and fluxes become

$$\mathbf{U} = (D, \mathbf{S}, \tau, \mathbf{B})^{\mathrm{T}} \tag{5}$$

$$= (\Gamma\rho, (\xi + B^2)\mathbf{v} - (\mathbf{v}\cdot\mathbf{B})\mathbf{B}, \xi + \frac{B^2}{2} + \frac{1}{2}(v^2B^2 - (\mathbf{v}\cdot\mathbf{B})^2) - p - D, \mathbf{B})^{\mathrm{T}}, \tag{6}$$

$$\mathbf{F} = (D\mathbf{v}, (\xi + B^2)\mathbf{v}\mathbf{v} - \frac{\mathbf{B}\mathbf{B}}{\Gamma^2} - (\mathbf{v}\cdot\mathbf{B})(\mathbf{B}\mathbf{v} + \mathbf{v}\mathbf{B}) + Ip_{\mathrm{total}}, \tau\mathbf{v} + p_{\mathrm{total}}\mathbf{v} - (\mathbf{v}\cdot\mathbf{B})\mathbf{B}, \mathbf{v}\mathbf{B} - \mathbf{B}\mathbf{v})^{\mathrm{T}}. \tag{7}$$

In these equations, we again introduce the total pressure p_{total}, which now reads

$$p_{\mathrm{total}} = p + \frac{(\mathbf{v}\cdot\mathbf{B})^2}{2} + \frac{B^2}{2\Gamma^2}. \tag{8}$$

In the case of an ideal equation of state with constant adiabatic index γ, the still unspecified variable ξ becomes

$$\xi = \Gamma^2\left(\rho + \frac{\gamma p}{\gamma - 1}\right). \tag{9}$$

For special relativistic simulations, the needed conversions from conservative to primitive variables exploit a Newton–Raphson strategy to handle the intrinsically non-linear relation between these two sets. We refer to [7,23] for further details specific to relativistic equation modules available in AMRVAC.

For the purpose of this paper, it suffices to note the unchanged nature of the induction equation for the temporal evolution of the magnetic field $\mathbf{B}$, which in addition must conform to Maxwell's equation $\nabla\cdot\mathbf{B} = 0$. In contrast to several recent approaches [1,25] emphasizing the need for maintaining a pre-chosen divergence-free representation of $\mathbf{B}$ across the AMR grid hierarchy, we continue to exploit a simple source term approach which diffuses errors at the maximal rate compliant with the CFL condition. We briefly summarize its philosophy and its applicability to curvilinear coordinates in Section 4.3. Note that we can exploit the same, simple strategy for both classical as well as special relativistic MHD computations. The collected tests are found in Section 5, and include an assessment of patch-based versus hybrid block-based AMR on a representative case with details on execution times and openMP parallel scaling behavior. Section 6 summarizes and indicates future research opportunities.

2. Hybrid block-based adaptivity

One of the main issues of any AMR scheme is the question how one manages the grid hierarchy and defines their data structure. Specifically, we here describe the data structures used in our hybrid AMR scheme. This new AMR variant has been implemented in the AMRVAC code [28]. Extensive use has been made of the so-called Loop Annotation SYntax (LASY) developed by Tóth [37]. This syntax allows us to handle all data structures in a dimension independent fashion. A tree data structure is – depending on the dimensionality – a binary tree, a quadtree, or an octree variant.

2.1. Data structures

In contrast to AMR schemes based on unequally-sized patches, but similar to block-AMR schemes using a tree data structure, the resolution of all individual AMR grids is kept the same. At the lowest level $l = 1$, this means that the computational domain is decomposed in equal sized grids and they are indexed by global grid indices. These indices are set by the physical extension of the domain, as divided in equal block sizes. If grids are refined to produce children grids at a higher level, the children grids belonging to each parent will also be indexed locally. Since a fixed grid refinement factor of two is assumed in our hybrid AMR scheme (the patch-based implementation allows any even refinement ratio), these local indices simply range from one to two in each physical dimension. The 'hybrid' strategy in our implementation in essence allows that not all children need to be created: only proper nesting of grid levels is required. This relaxation of the usual block-AMR approach where always complete families are used (i.e. each parent always has 2, 4 or 8 children depending

on dimensionality) incorporates the idea of optimally fitting developing flow features with the grid structure, which is central to the patch-based approach.

Our hybrid AMR scheme then effectively uses two means to traverse the grid hierarchy: there is a doubly linked list of grid pointers per level in addition to the tree data structure containing grid pointers between parent and children grids. The necessity of the first linked list originates from the way the time-advances are organized. Here, we use the original time-stepping approach described by Berger and Colella [5]. Grids are refined and coarsened in both cell spacing as well as their associated time steps. The latter means that grids on a certain AMR level are time updated at moments which are generally speaking intermediate to available times at lower-lying levels. With a linked list per level, grids of a certain level are easily found, e.g. when needed for spatio-temporal interpolations to provide boundary values for overlying grids. The tree data structure however contains all required information on grids in the direct neighborhood of a certain grid. It includes the usual pointer to the parent grid as well as pointers to all existing children. If a certain child is not created, its corresponding tree pointer is zero. Each child of a given parent needs to know its spatial location with respect to the parent: this boils down to storing the local grid indices in the tree structure. Our implementation then adds three more pointers to the tree data structure: we provide pointers to grid neighbors at the same level, to neighbors one level down (i.e. parent level) neighbors, and to neighbors one level up (i.e. at child level). While the first two are used throughout the AMR scheme, the latter pointers are in the hybrid AMR scheme only used for fixing the fluxes at coarse cells directly adjacent to fine cells, crucial to ensure the global conservation property.

2.2. Neighbor search algorithm

In the neighbor search algorithm, central to any AMR scheme using a tree data structure, special care has been taken that this algorithm allows for trees with incomplete branched families. On level $l = 1$, the connectivity between all AMR grids is known a-priori (e.g. from the global grid pointers available on that level only), so that the neighbors of that level are easily determined. Assume now that for a given level l the neighbors are already known. Neighbor information for child level $l + 1$ is then obtained via the known neighbors at parent level l as follows. We sweep sequentially through the linked list of level l, and fill for each such parent level l grid a temporary array of size $(3 \times 2)^D$ for a D-dimensional problem. For a 2D case this is schematically illustrated in Fig. 2, where the 6 by 6 array identifies all (maximally four) children of the parent itself, as well as the (maximally 32) children of all 8 surrounding neighbors of the parent. Once such an array is known for each parent level l grid, each grid on level $l + 1$ can easily determine its level $l + 1$ neighbors from these arrays. The parent neighbors of each level $l + 1$ grid, i.e. the tree neighbor connections down to level l, are directly obtained from the previously determined neighbors of the parent. Finally, for the parent grid at level l, the neighboring children connections one level up to $l + 1$, can also be read from the same temporary arrays created. The entire tree neighbor connectivity is in this way reconstructed via a level recursion from any fixed base

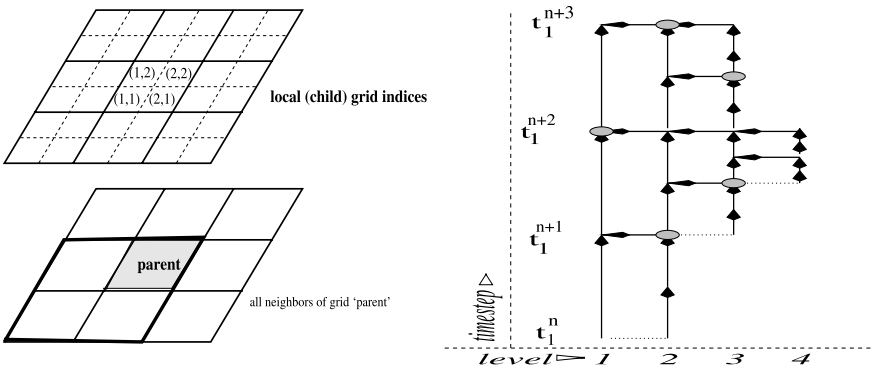


Fig. 2. Left: schematic of the temporary array used to identify parent-children pairs in the neighbor search algorithm for a 2D case. Right: a hypothetical sequence of 3 time integration steps for a 4 level AMR run, as discussed in the text.

level l_{base}. Essential in this whole process is how it conforms with the hybrid AMR approach, where it is allowed that a parent does not have all its four children created.

3. AMR strategy: summary

In the end, we need to time-advance the grid hierarchy while keeping the grids properly nested and maintaining the solution consistent and conservative via an update process described below. As an example, we show in Fig. 2 a hypothetical time-sequence of three time steps on level $l = 1$. Here, the maximum allowed level is assumed to be $l_{\max} = 4$. The vertical arrows correspond to the time advancement of the levels and the horizontal arrows indicate conservation updates across levels. Initially, new grids on level $l = 2$ are triggered. Each new grid on a higher level l will have a finer mesh size Δx_i (where $i = 1, \ldots, D$ for a D-dimensional simulation) and a smaller time step Δt, such that the refinement ratio is

$$r_l \equiv \frac{\Delta x_{i,l-1}}{\Delta x_{i,l}} \equiv \frac{\Delta t_{l-1}}{\Delta t_l}. \tag{10}$$

For the hybrid AMR scheme, we always use a refinement ratio $r_l = 2$. When a particular level $l > 1$ has reached the same time as an underlying level $l - 1$, the horizontal arrows indicate that the unknowns of the underlying coarse cells are then updated with the fine cell information. More importantly, fluxes of the coarse cells directly neighboring the fine cells are updated with the more accurate fluxes used in the fine cell integrations. The latter is standard practice for any AMR scheme maintaining the global conservation properties. We note that as a direct consequence of allowing incomplete grid families, our hybrid AMR scheme will have the outermost boundaries of the collection of fine grids not necessarily coinciding with the boundaries of coarser level grids. Therefore, the already discussed neighbor search algorithm and neighbor information as stored in the tree data structure is again vital for implementing this flux-fix strategy in practice.

The refinement and coarsening operations follow the approach discussed next. Each time a certain level l_{base} has performed (at least) two time steps, as indicated by the grey dots in Fig. 2, all grids on levels $l_{\text{base}} + 1$ up to the currently finest level l_{fine} are evaluated for possible coarsening (i.e. removal of overlying children grids) or refinement (creation of children level grids). All cells on grids from level $l = l_{\text{base}}$ to $l = \min(l_{\text{fine}}, l_{\max} - 1)$ which need refining based on some physically and/or numerically controlled criteria are flagged. We follow the Richardson extrapolation approach as summarized in [28], augmented with possible user-controlled criteria. In a second step, individual grid blocks are identified for refinement or coarsening depending on whether any or none of their cells are flagged. Since grid families do not have to be complete, there are as many refinement and coarsening identifiers per grid as its maximum allowed children, i.e., for each grid in the tree hierarchy there are 2^D flags for a D-dimensional problem. Each identifier can be set independently, and 'refine' introduces a new $l + 1$ grid, i.e., one child grid, while the 'coarsen' flag is meant to remove such a grid. In contrast to the tree block-based method, in the hybrid AMR strategy grids triggered for refinement do not need to be completely covered by their children grids as we can also see in Fig 1. The grid hierarchy created in the end should be properly nested at all times, meaning that all grids on level $l + 1$ have neighbors on level l or $l + 1$. The grid nesting for the hybrid AMR scheme is guaranteed as follows:

(1) Descend from level $l = \min(l_{\text{fine}}, l_{\max} - 1)$ to $l = l_{\text{base}} + 1$. For each candidate $l + 1$ grid identified on a level l flagged for refinement, check if it is properly nested. When it is not, flag all neighbor parent $l - 1$ grids for refinement (i.e. introduce new candidate l grids). Note that not all children of these neighbor parents need to be created, but only those to ensure proper grid nesting for level $l + 1$.
(2) If the result of the previous downward sweep is such that grids on the fixed level $l_{\text{base}} \neq 1$ would end up not properly nested: ascend again from $l = l_{\text{base}}$ to $l = \min(l_{\text{fine}}, l_{\max} - 1)$ and unflag for refinement each grid causing problems for that particular nesting criterion. However, we need to avoid that individual cells flagged by the Richardson-like procedure are in the end not refined. This problem is circumvented by adding two types of buffer zones around each flagged cell as described in the next paragraph.
(3) Finally, descend once more from level $l = l_{\text{fine}} - 1$ to $l = l_{\text{base}}$ and check if coarsening of grids (i.e. removal of any of its children) is allowed in the new grid hierarchy. If it would conflict with ensuring the proper nesting, deactivate the coarsening.

Of coarse, level 1 is never coarsened, while the maximal allowed level $l_{\max}$ is never evaluated for refinement.

While the outlined regridding procedure creates a properly nested grid hierarchy, some grids flagged for refinement will not be created unless we add a buffer zone of a certain size around each individual cell identified by the Richardson-like process. This extra buffer should not be unduly large. The size of the buffer should be such that flagged cells are truly refined. We use two types of buffers:

(1) A level dependent buffer zone is added around each flagged cell to ensure that grids can be created on higher AMR levels in a properly nested manner. The size of this buffer is determined as follows. On level $l = l_{\max} - 1$, if it is in use at all, a buffer zone of $1/4$ the grid size (expressed in number of cells) is added. This factor is chosen such that a grid two levels higher up can be triggered (keeping in mind the refinement ratio of two). On level $l = l_{\max} - 2$, if in use, for the same reason a buffer zone of $1/4$ the grid size is added, but also half the $1/4$ buffer zone of one level up. This amounts to a buffer zone on level $l = l_{\max} - 2$ of $1/4 + 1/8 = 3/8$ times the grid size on that level. Continuing this recursive procedure downwards results in a buffer zone for level $l < l_{\max} - 1$ of

$$\frac{2^{(l_{\max}-l-1)} - 1}{2^{(l_{\max}-l)}},$$

times the grid size on level l.

(2) The buffer described in (1) above would be sufficient if the regridding is taking place from level $l = 1$ and above. However, if the regridding is performed from level $l = l_{\text{base}}$ and above, where $l_{\text{base}} > 1$, then individual flagged cells might not be refined (see also [4]). To avoid this problem, we use a buffer zone, as in [4], of n_{buff} (a parameter typically set to 2 when each level is evaluated for refinement after two consecutive timesteps) in each direction. This is usually sufficient to ensure that discontinuities and other regions that need high resolution do not propagate to coarse grids before regridding takes place.

Note that buffer type (1) is small on highest levels and even zero on level $l = l_{\max} - 1$. Since most grids in the grid hierarchy are usually on the highest level, the impact of this buffer on the performance is minimal.

4. Handling curvilinear coordinate systems

The use of curvilinear coordinates introduces relatively obvious changes in the flux formulae, as well as possible geometric source terms specific to the equations. Another important change for an AMR scheme is in the restriction operations from fine to coarse cells and the prolongation formulae from coarse to fine cells. These formulae need to satisfy the conservation property and the order of the interpolation should not be less than that of the numerical scheme in use. Our approach generalizes the one by Balsara [1] with respect to its applicability to more general coordinate systems and a significant simplification in the end formulae. In [1], emphasis was placed on constrained transport, staggered approaches to controlling $\nabla \cdot B$. We use cell-centered based approaches for the latter, and in particular rely on the diffusive approach discussed in [28] and references therein. We briefly mention below how this method can be suitably used in curvilinear coordinates as well. Other related prolongation formulae suited for cylindrical and spherical geometries are presented in [25].

4.1. Restriction and prolongation formulae

In the following, we present the case of 2D, since this already shows the basic idea. The curvilinear coordinates are denoted by (x_1, x_2) and the Jacobian by $J = J(x_1, x_2)$. For a cell indexed by (i, j), the lower and upper coordinate bounds are indicated by $(x_{1;i-1/2}, x_{2;j-1/2})$ and $(x_{1;i+1/2}, x_{2;j+1/2})$, respectively. The volume ΔV_{ij} is obtained by integrating the Jacobian over these coordinate ranges. The conservative restriction formula, for e.g. the density, then simply becomes $\rho_{ij}\Delta V_{ij} = \sum_m \sum_n \rho_{mn} \Delta V_{mn}$, where (m, n) indexes the fine cells covering the coarse (i, j) cell.

The prolongation in curvilinear coordinates is somewhat more involved. We restrict the analysis to coordinate systems for which the Jacobian is separable, i.e. $J = J(x_1, x_2) = J_1(x_1)J_2(x_2)$. Making a change of coor-

dinates from (x_1, x_2) to (y_1, y_2) with $y_i = y_i(x_i)$ such that $\mathrm{d}y_i = J_i(x_i)\mathrm{d}x_i$ for $i = 1, 2$, the lower and upper bounds of a cell (i, j) in y coordinates are denoted by $(y_{1;i-1/2}, y_{2;j-1/2})$ and $(y_{1;i+1/2}, y_{2;j+1/2})$, respectively. Note that the volume of a cell is $\Delta V_{ij} = \Delta y_1 \Delta y_2$ with $\Delta y_1 = y_{1;i+1/2} - y_{1;i-1/2}$. The center of a cell can be defined from the arithmetic average of either the x- or the y-coordinate system, and in general these two centers do not coincide. This is illustrated in Fig. 3. The first one, i.e. $(x_{1;i}, x_{2;j}) = 1/2[(x_{1;i-1/2}, x_{2;j-1/2}) + (x_{1;i+1/2}, x_{2;j+1/2})]$, is used for the division of a coarse cell into fine cells. However, to obtain the density in the fine cell we will use a Taylor expansion about the cell center in y-coordinates, i.e. $(y_{1;i}, y_{2;j}) = 1/2[(y_{1;i-1/2}, y_{2;j-1/2}) + (y_{1;i+1/2}, y_{2;j+1/2})]$, resulting in

$$\rho(y_1, y_2) = \rho_{ij} + \frac{\Delta_1 \rho_{ij}}{\Delta y_1}(y_1 - y_{1;i}) + \frac{\Delta_2 \rho_{ij}}{\Delta y_2}(y_2 - y_{2;j}). \tag{11}$$

Here, $\rho_{ij} = \rho(y_{1;i}, y_{2;j})$ and $\Delta_1 \rho_{ij}$ denotes the minmod limited slope in direction x_1 involving centered, left-sided, and right-sided slopes determined from ρ_{i-1j}, ρ_{ij}, and ρ_{i+1j}, and similarly for $\Delta_2 \rho_{ij}$. The integrated value of $\rho(y_1, y_2)$ over the coarse cell equals $\rho_{ij}\Delta V_{ij}$, consistent with conservation. To compute the density for the fine cells, the evaluation points (y_1, y_2) should be properly chosen. From now on, let the four fine cells be indexed by $i \pm 1/4, j \pm 1/4$. The fine cell $i + 1/4, j + 1/4$, e.g. has lower and upper coordinate bounds $(y_1(x_{1;i}), y_2(x_{2;j}))$ and $(y_{1;i+1/2}, y_{2;j+1/2})$, respectively. This cell has center $(y_{1;i+1/4}, y_{2;j+1/4}) = 1/2[(y_1(x_{1;i}), y_2(x_{2;j})) + (y_{1;i+1/2}, y_{2;j+1/2})]$. Inserting these coordinates in Eq. (11) and writing out the cell centers explicitly in terms of the cell boundary coordinates results in

$$\rho_{i+1/4,j+1/4} = \rho(y_{1;i+1/4}, y_{2;j+1/4}) = \rho_{ij} + \frac{1}{2}\frac{\Delta_1 \rho_{ij}}{\Delta y_1}(y_1(x_{1;i}) - y_{1;i-1/2}) + \frac{1}{2}\frac{\Delta_2 \rho_{ij}}{\Delta y_2}(y_2(x_{2;j}) - y_{2;j-1/2}). \tag{12}$$

Multiplying both the numerator and denominator of the second term on the right hand side by $(y_{2;j+1/2} - y_2(x_{2;j}))$ and of the third term on the right hand side by $(y_{1;i+1/2} - y_1(x_{1;i}))$ then gives the prolongation for the fine cell $i + 1/4, j + 1/4$ as

$$\rho_{i+1/4,j+1/4} = \rho_{ij} + \frac{1}{2}\Delta_1 \rho_{ij}\frac{\Delta V_{i-1/4,j+1/4}}{\Delta V_{i-1/4,j+1/4} + \Delta V_{i+1/4,j+1/4}} + \frac{1}{2}\Delta_2 \rho_{ij}\frac{\Delta V_{i+1/4,j-1/4}}{\Delta V_{i+1/4,j-1/4} + \Delta V_{i+1/4,j+1/4}}. \tag{13}$$

Analogous prolongation formulae can be obtained for the other fine cells. By multiplying the densities of the fine cells with their volumes and adding the results of all four fine cells, it is easily seen that this prolongation is conservative.

4.2. π-periodicity at singular boundaries

For geometries that involve a singular axis, special care has to be taken for handling the boundary conditions there in the tree data structure. The cylindrical coordinate system (r, ϕ, z), e.g. has a coordinate singularity at $r = 0$. The problem of this singularity can be circumvented by identifying each ghostcell at the $r = 0$ boundary at given z and azimuthal angle ϕ with a cell located on the other side of the pole, i.e. at $\phi + \pi$. Since

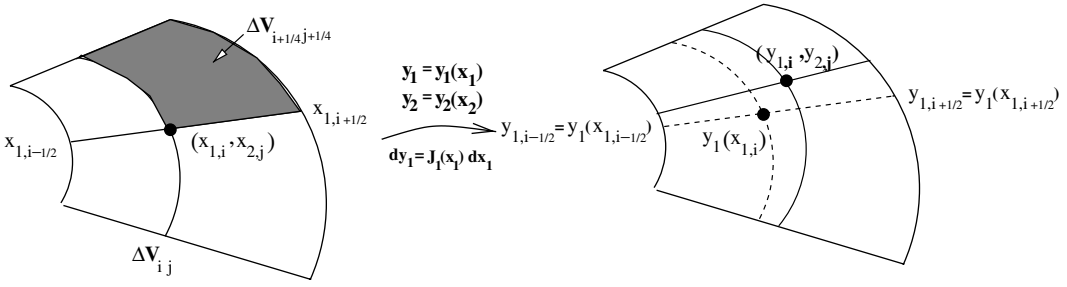


Fig. 3. Division of a 2D cell in curvilinear coordinates into subcells, in both the employed coordinate space (x_1, x_2) and in the corresponding (y_1, y_2) space identified by the Jacobian. The arithmetically determined centers do not coincide generally, and the $(y_{1,i}, y_{2,j})$ center is used in the Taylor expansion, while the $(x_{1,i}, x_{2,j})$ center determines the cell division. See text for details.

this cell is generally located on another grid, the neighbor information in the tree structure therefore makes use of grids π-periodically displaced in ϕ. Moreover, since the unit vectors in the r- and ϕ-direction reverse direction for cells at $\phi + \pi$ compared to the ghost cells at ϕ, the r and ϕ components of the vectors need a sign reversal in the ghost cells. In Section 5, 2D and 3D examples are shown that successfully employ this 'π-periodic boundary condition' at the pole.

4.3. Diffusive control for numerical magnetic monopoles

In order to handle the $\nabla \cdot \mathbf{B} = 0$ constraint, we use the diffusive approach proposed by Marder [26]. In this method, any numerically generated divergence is diffused away at the maximal rate allowed by the CFL condition. A similar strategy is used in the FLASH code [16]. The induction equation is changed by adding a source term $\eta \nabla \nabla \cdot \mathbf{B}$, where η is an artificial diffusion parameter. By applying a von Neumann stability analysis on the induction equation augmented with this source term, and assuming a central discretization for $\nabla \cdot \mathbf{B}$, it follows that

$$\eta \Delta t \leqslant 2\left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} + \frac{1}{\Delta z^2}\right)^{-1}, \tag{14}$$

for stability. Note the factor 2 instead of the 1/2 which is normal for a diffusion equation, as a consequence of having a gradient of a divergence instead of a divergence of a gradient. By setting the maximal allowed diffusive time step equal to the CFL time step, we obtain the following source term update

$$\mathbf{B} := \mathbf{B} + C_{\mathrm{d}}\left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} + \frac{1}{\Delta z^2}\right)^{-1} \nabla \nabla \cdot \mathbf{B}, \tag{15}$$

where $0 \leqslant C_{\mathrm{d}} \leqslant 2$. For curvilinear coordinates the cell spacings Δx, Δy, and Δz should be replaced by their curvilinear counterparts, e.g. Δr, $r\Delta\phi$, and Δz in cylindrical coordinates. Note that then for large r, the $r\Delta\phi$ contribution vanishes, while near the pole, for small r, this contribution is the dominant one. This behavior is essential for the numerical stability of the diffusive approach.

5. Numerical examples

We present a suite of 7 test problems, ranging from 1D to 3D cases. For each case we describe the complete setup as well as the AMR specific parameter settings. More elaborate descriptions of the various parameters that usually enter the AMR scheme, in particular our way of handling the Richardson type error estimator for flagging grid cells, can be found in [28]. For the purpose of this paper, it suffices to mention which variables have been employed in this estimator, and the value of the tolerance parameter ϵ_{tol} above which refinement is triggered.

5.1. 1D MHD shocktube

Our first test case is taken from Torrilhon [34]. This test is a pure 1D MHD Riemann problem on $x \in [-1, 1.5]$, $\gamma = 5/3$, with left and right states given by

$$x < 0 : (\rho, v_x, v_y, v_z, p, B_x, B_y, B_z) = (1, 0, 0, 0, 1, 1, 1, 0),$$
$$x > 0 := (0.2, 0, 0, 0, 0.2, 1, \cos 3, \sin 3).$$

This initial condition corresponds to an almost coplanar problem (coplanar problem when replacing $3 \to \pi$), which was demonstrated to pseudo-converge to the wrong solution up to few thousand grid points before switching to convergence to the true solution. The wrong solution contains a leftward going fast rarefaction, a slow compound wave (slow shock with rarefaction attached to it), a contact discontinuity, a rightward going slow shock, an Alfvén signal (rotational discontinuity) and a rightward going fast rarefaction. The slow compound wave gets replaced in the true solution by a separate rotational (Alfvén) discontinuity and a slow shock. Only the actual coplanar problem would allow both types of solution, i.e. a solution containing only regular

waves *or* a solution containing the compound structure. However, the almost coplanar test stated above has a unique solution containing only regular waves [35]. A large variety of modern shock-capturing finite volume schemes have been demonstrated to suffer from this non-uniform convergence, and it is therefore of interest to demonstrate that the AMR scheme is capable of recognizing the need for high resolution to find the true solution. In [11], the same test was exploited in a *r*-refinement approach which successfully captured the true solution if again sufficient grid points where allowed to relocate.

This test therefore demonstrates that AMR is in fact vital for all MHD computations, since all modern shock-capturing schemes (TVD, WENO, ...) in use suffer from this pseudo-convergence, and only by having a high effective resolution can one currently circumvent this problem, see [36].

The computed solution at $t = 0.4$ is plotted in Fig. 4 for two AMR runs, exploiting 4 versus 8 refinement levels, respectively. We use a base resolution of 800 grid points, and combined a TVDLF [39,41] discretization on the lower four levels, with a TVD method [21] on the upper four levels. For a more detailed description of these different shock-capturing methods as implemented in VAC and AMRVAC, we refer to [39] and references therein, and also to the Appendix. It was demonstrated earlier that even for problems evolving to more turbulent endstates, the computational efficiency of the level-dependent use of a spatial discretization acts as

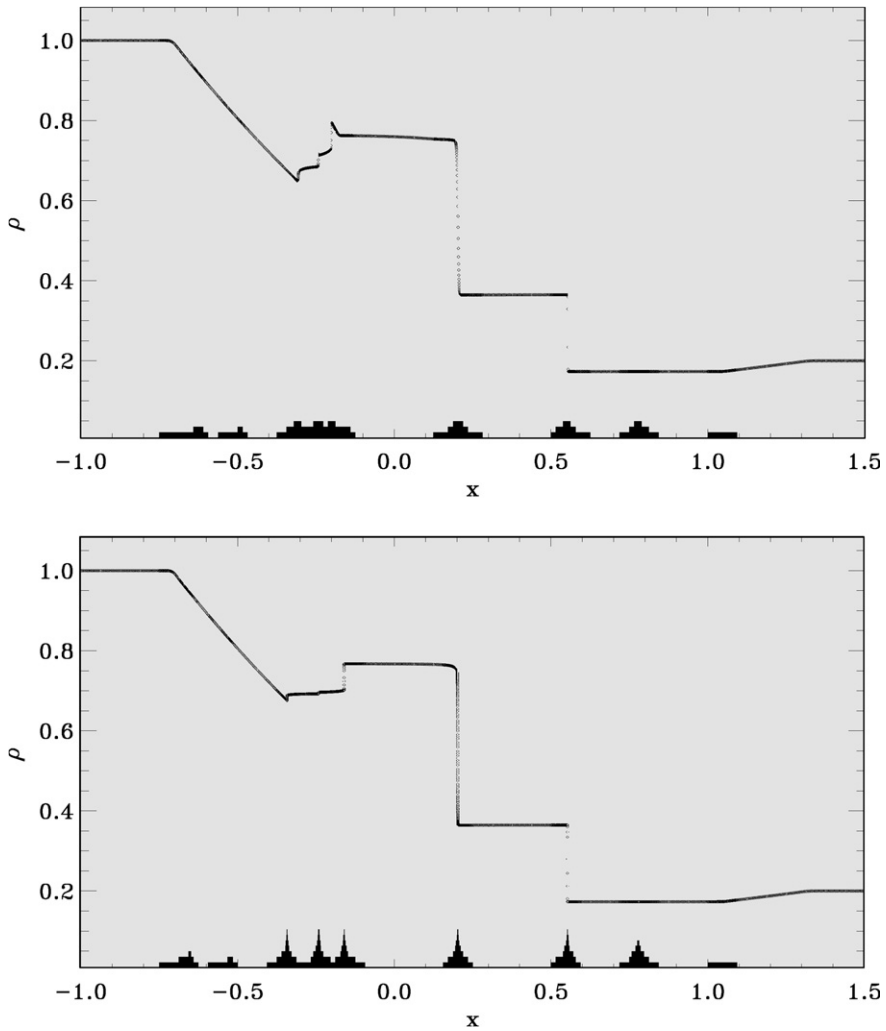


Fig. 4. The Torrilhon shock tube case, for a 4-level AMR run (top) and a 8-level AMR run. The wrong compound structure at about $x \approx -0.25$ only disappears at very large effective resolutions, in accord with previous findings on fixed grids.

an efficient, hybrid scheme. We therefore follow this hybrid approach in some of our tests. In our AMR runs, refinement is triggered on density and B_y in an unequally weighted fashion (ratio 1–9), but the tolerance criterion is set to a relatively low value $\epsilon_{\text{tol}} = 0.0002$. The 4-level run, equivalent to a fixed grid simulation of 6400 cells, takes 21 s (3 GHz Pentium 4 processor) and still has the erroneous compound structure at $x \approx -0.25$. Allowing 8 levels increases the execution time to 341 s to get an effective resolution of 10,2400 cells, but now we recognize the regular solution containing the Alfvén discontinuity (not seen in the density plot of Fig. 4) and slow shock. For this 1D test, there are virtually no differences between patch versus hybrid-block execution times and the results obtained.

5.2. 2D HD Richtmeyer–Meshkov

As a representative planar test problem, we simulate the hydrodynamic interaction of a strong shock with a density discontinuity, which induces the Richtmeyer–Meshkov instability on the perturbed interface. On a domain $(x, y) \in [0, 5] \times [0, 1]$, we place a planar Mach 10 shock aligned with the y-direction at $x = 1/10$ with post and pre-shock states given by

$$(\rho, \rho v_x, \rho v_y, e)_{\text{post}} = (8, 8 \times 8.25, 0, 563.5),$$
$$(\rho, \rho v_x, \rho v_y, e)_{\text{pre}} = (1.4, 0, 0, 2.5),$$

where we take adiabatic index $\gamma = 1.4$. Note that this same planar shock is used in the standard double Mach reflection (shock reflecting of a wedge) from Woodward and Collela [40]. Here, we intentionally align the shock with the y-axis to show that the employed discretization does not suffer from odd-even decoupling, or carbuncle phenomena which may appear for some characteristic-based discretizations [31]. Top and bottom boundaries $y = 0$, 1 are modeled as reflective walls by imposing (a) symmetry conditions on conserved variables. The pre-shock state beyond $x > 1/10$ connects via a contact discontinuity to a state of tenfold lower density $\rho_{\text{low}} = 0.14$, where the discontinuity occurs across the incline $y = (x - 1.5)\tan(\pi/3)$. The boundary conditions at left and right use continuous extrapolation of conserved quantities in the ghost cells, essentially leaving the left border at post-shock state and mimicking an open boundary at right. We simulated the shock interaction until $t = 0.35$; representative snapshots are shown in Fig. 5. We employ MC/Woodward limiting on the primitive variables in TVDLF, allowing for 7 refinement levels at a coarsest resolution of 120×24. The refinement is triggered by a Richardson type extrapolation on density only, with tolerance $\epsilon_{\text{tol}} = 0.005$.

At startup, the planar shock leaves both a small acoustic and entropy error in its wake, but neither influence the dynamics occurring when the shock starts impinging the contact surface. This is ensured by the fairly large standoff distance between shock and contact surface at $t = 0$. As the shock passes over the incline, it accelerates in the lower density medium and that part of the shock front bends forward. At the same time, the perturbed initial contact discontinuity becomes unstable, similar to the familiar Rayleigh–Taylor instability where gravity causes relative accelerations. At the precise time where the initial planar shock passes the top corner of the initial contact discontinuity, the now forwardly bent shock front reflects off the top wall, in a completely similar fashion as the already mentioned Mach reflection problem. However, in the present case the reflection is occurring in the presence of an intersecting unstable interface. In the later snapshots of Fig. 5, we can recognize the reflected shock (curved shock at top behind the unstable interface), as well as the typical configuration where the Mach stem connects via a triple point to the original reflected shock. This Mach stem is trailed by a jet traveling along the top reflecting wall, which, as noted in pure reflection problems computed with AMR [5], in turn demonstrates Kelvin–Helmholtz roll-up along the slip line (connecting to the triple point) and the jet boundary.

This test is used to compare the patch-based versus the new hybrid block-AMR approach, in particular to quantify the cost-effectiveness of these two approaches. On a single 2.2 GHz Opteron processor, the execution time of a patch-based approach where grid sizes up to a maximal size of 40×40 are allowed was 17,176 s. The number of grids on the highest AMR level went up to about 870, and an impression of the domain coverage of the $l > 1$ levels as a function of time is shown in Fig. 6. In contrast, a hybrid block run with fixed grid sizes of 12×12 eventually triggers about 2600 grids on the highest level alone, thus implying more internal boundary management. Still, due to its hybrid nature, the grid structure can more optimally adjust to the solution as

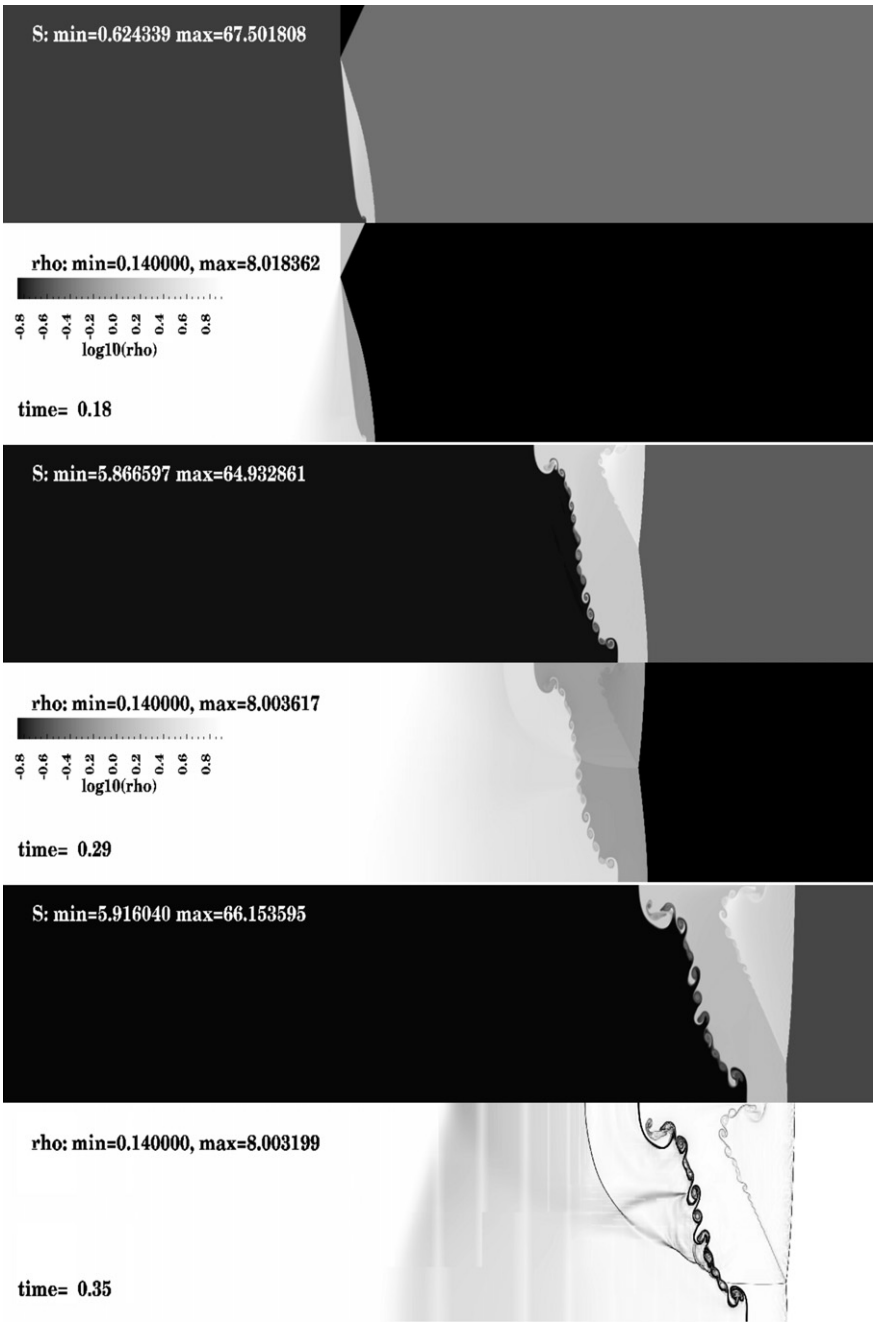


Fig. 5. A Mach 10 shock impinging on an inclined density discontinuity between reflective walls triggers the Richtmeyer–Meshkov instability on the perturbed interface as well as complex shock reflection dynamics. Shown are three consecutive times of the interaction, while impacting the incline (top), and at two later times. We show entropy and density grayscale plots, along with a Schlieren plot of density at $t = 0.35$ below.

reflected by the corresponding domain coverage also shown in Fig. 6. In effect, the execution time thereby decreased to 11,500 s, while the solution itself is qualitatively similar. Another aspect where the hybrid block approach is better than the patch-based one is in its parallel scaling behavior. The right panel of Fig. 6 compares OpenMP scaling behavior for both approaches for the entire simulation up to $t = 0.35$, executed on 1, 2, 4 up to 8 processors. Since the workload per grid is identical in the hybrid block run, its scales better, although the improvement is admittedly marginal.

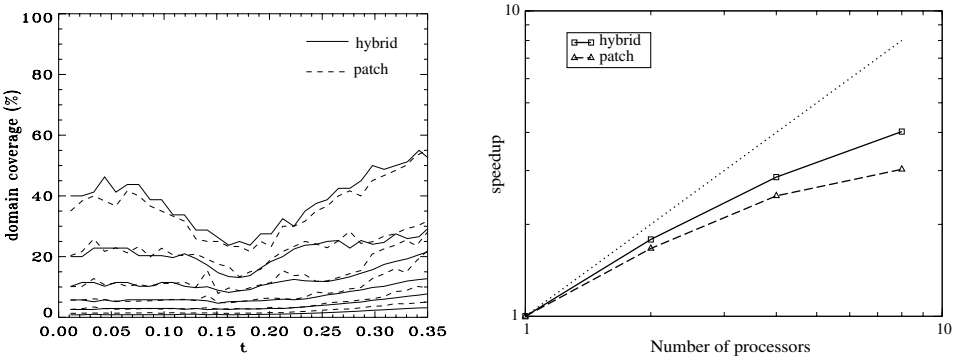


Fig. 6. Left: domain coverage of all levels $l > 1$ for hybrid block-based versus patch-based AMR run of the Richtmeyer–Meshkov application. Right: Comparison of the parallel scaling behavior for patch-based versus hybrid block-based AMR on the 2D Richtmeyer–Meshkov test, simulated until time $t = 0.35$ as shown in Fig. 5. The OpenMP parallelism improves marginally due to the fixed block sizes employed in the latter approach.

5.3. Polar grid shock test

To demonstrate the handling of general orthogonal grids in our hybrid block-based AMR approach, we set up an essentially 1D planar hydrodynamic shock tube problem on a circular domain. We solve the problem in (r, φ) polar coordinates and use the corresponding polar vector components for the momentum representation. The initial condition is taken from the classical Sod problem, where $\gamma = 1.4$ and the 1D Euler–Riemann problem has static left and right states, with $(\rho, p)_{\mathrm{L}} = (1, 1)$ and (0.125,0.1), respectively. We take a circle of radius 2, and place the discontinuity as a straight line at $x = -0.5$ (where $x = r\cos(\varphi)$). The solution is shown at $t = 0.4$, in Fig. 7. At that point, the rightward going shock has traveled across the pole of the circular domain, demonstrating the correct handling of the singular $r = 0$ boundary by the π-periodic boundary conditions. Refinement is triggered on density, and on both momentum components with relative weights (0.5, 1/4, 1/4) and a tolerance of 0.00001. We allow 4 grid levels, with a base resolution of 60^2.

The result of the simulation is shown in Fig. 7, with a visualization of the grid structure in the (r, φ) plane, at the end of the run. At the far $r = 2$ boundary, we continuously extrapolate the Cartesian x-component of momentum along the radial direction, as computed from the polar vector components, while the y-component is set to zero. This reflects the 1D nature of the simulated shock tube. The horizontal cut at $t = 0.4$ shows that the simulation correctly recovers the solution to this Riemann problem. Note that at the time shown, the rightward going shock has already crossed the singular point $r = 0$ without distortion of the straight shock front.

5.4. Planar magnetized shear layer

Returning to Cartesian grids, we present a long-term simulation of a magnetized planar shear layer characterized by a velocity $\mathbf{v} = \frac{V}{2}\tanh(20y)\hat{e}_x + \delta V \exp(-25y^2)\mathrm{ran}(x)\hat{e}_y$. The vertical velocity field contains the random number function $\mathrm{ran}(x) \in [-0.5, 0.5]$ inducing white noise perturbations with amplitude $\delta V = 0.01$. The main flow field is quantified by its Mach number $M = V/c_{\mathrm{s}} = 1$, where density and pressure normalizations take $\rho(t = 0) = 1$ and $p(t = 0) = 1$ with $\gamma = 5/3$. The strength of the initially horizontal and uniform magnetic field is fixed by the plasma beta parameter $\beta = 2p/B^2 = 58.8$. This configuration is Kelvin–Helmholtz unstable and it was used [2] to investigate the long term evolution of extended shear layers. Units are chosen such that the most unstable mode has a horizontal wavelength of order unity, and by taking a domain $(x, y) \in [0, 40], [-4, 4]$ we allow for the development of up to 40 streamwise vortices. These vortices merge and coalesce to larger vortical structures, which under the chosen parameters happens simultaneously with magnetic disruptions inside the vortices due to small-scale reconnection events. As we numerically evolve the ideal MHD equations, the latter is influenced by the effective resolution and associated numerical 'resistivity'. In [2], resistive MHD computations confirmed the physical relevance of sufficiently high resolution 'ideal' simulations. The pairing/merging process is sensitive to relative phase differences between subharmon-

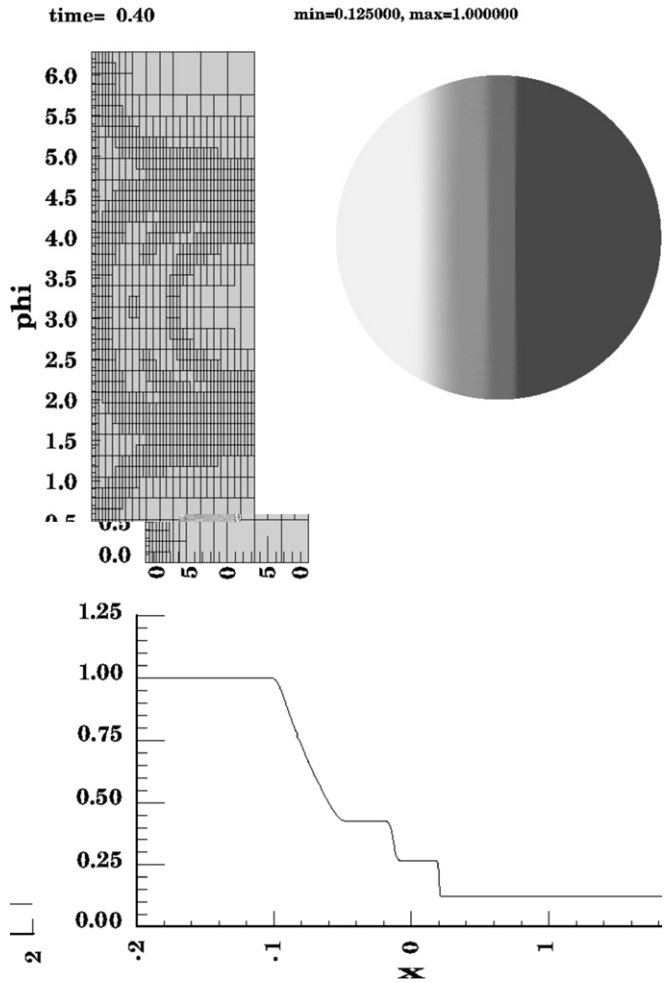


Fig. 7. An essentially 1D shocktube test performed in polar coordinates on a circular domain. We show the grid structure in the (r,φ) coordinate plane and the density in a grayscale on the computational domain (top). A cut along the middle of the grid (bottom), clearly identifying the leftward rarefaction, contact discontinuity and rightward shock.

ics of the most unstable mode, and can be studied separately by inducing deterministic velocity perturbations. Here, we solve the 2D MHD equations with periodic lateral boundaries and open top and bottom boundaries. Fig. 8 shows two snapshots of the simulation, displaying density and velocity fields. Note that in the top panel, corresponding to an early time in the evolution when the most unstable mode has grown to saturation, the vertical direction is artificially increased: the y-extent shown is $[-0.5, 0.5]$, to be compared to the full domain y-size $[-4, 4]$ depicted in the bottom panel. This simulation uses the diffusive approach for handling the numerical $\nabla \cdot \mathbf{B}$ errors, and extends the results of [2] to an even larger computational domain. The AMR run exploited 5 refinement levels, with TVDLF discretization on all but the highest grid level, where a TVD scheme is used. Refining is triggered by a weighted evaluation of density, x- and y-momentum where relative differences of 0.001 in the Richardson scheme activate grid levels. The base resolution uses 216×108 grid points. We simulated until $t = 20$, and at this time the domain coverage of the highest grid level has reached 20%, involving a total number of grids exceeding 12,000. The wallclock execution time on 4 CPUs is 17,833 s. We also ran a similar setup but with a deterministic initial perturbation with both patch and hybrid-tree strategies. This perturbation minimizes the pairing trend to reduce the overall computation cost. The execution time of the hybrid block-based 12,814 (s) shows an improvement over the patch scheme where the timing is 14,734 (s).

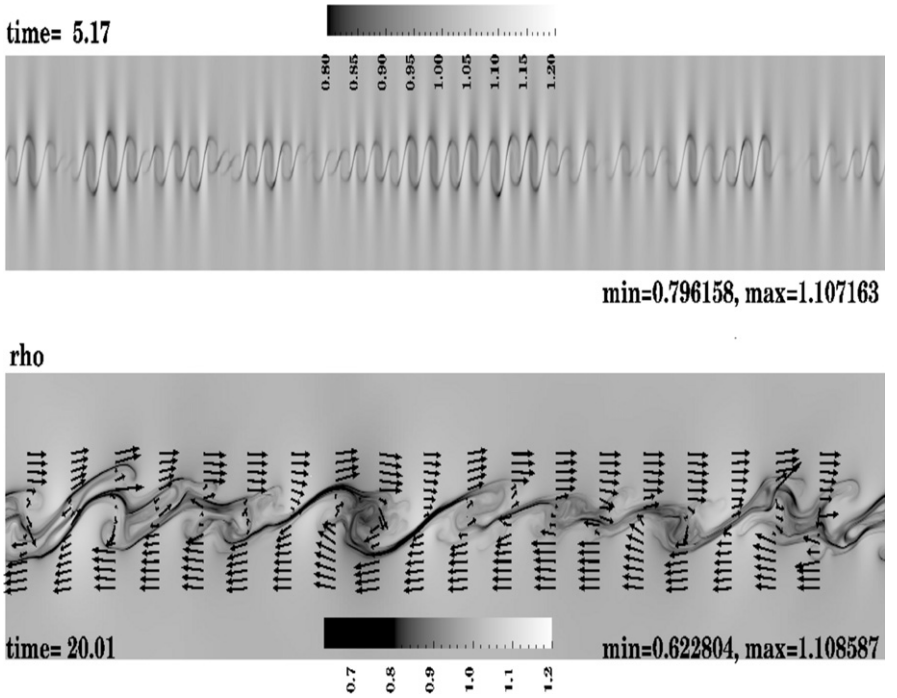


Fig. 8. A 2D magnetized shear flow layer demonstrating Kelvin–Helmholtz instability, mediated by magnetically controlled disruptions. The horizontal domain size spans up to 40 wavelengths of the most unstable mode (top, scaled up in the vertical direction), eventually merging and coalescing to larger-scale structures. Shown is the density as well as the flow field at the final time $t = 20$ (bottom).

In order to quantify more properties of hybrid block-based AMR for this problem, we take a more deterministic initial condition by replacing the $\mathrm{ran}(x)$ with a $\sin(2\pi x)$, thus selecting a single unstable mode on a more restricted domain $[0,1] \times [-2,2]$. Fig. 9 shows the convergence behavior of this simulation for 3, 4, 5, and 6 refinement levels, with the latter runs of similar resolution as used in the extended run from above. Convergence is here quantified by the temporal evolution of the cross-stream kinetic energy. The exponential growth agrees with the growth rate of the Kelvin–Helmholtz instability, and the non-linear saturation behavior is also adequately resolved. The right panel of Fig. 9 quantifies the instantaneous maximal monopole error (in absolute value) that occurs in a central difference evaluation across the grid hierarchy. The trend for dif-

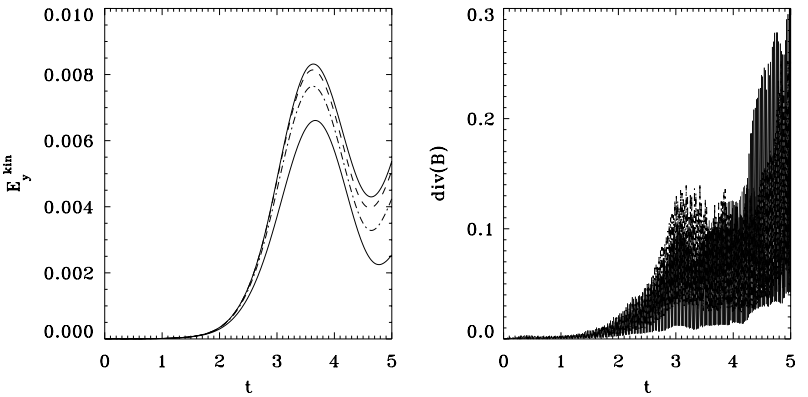


Fig. 9. Development of a Kelvin–Helmholtz instability in a 2D magnetized shear flow layer introduced by one single unstable sinusoidal mode. On the left, the convergence of the cross-stream kinetic energy with respect to the number of refinement levels. The right panel shows that the upper bound on $\nabla \cdot \mathbf{B}$ can reach up to 0.3 during the transient phase.

ferent refinement levels, and hence effective resolution, is rather similar: for the time shown such local errors can reach up to $\mathcal{O}(0.3)$. It should be noted that our diffusive approach is similar in spirit to other source term treatments, and acts primarily to stabilize the overall scheme. In this sense, the AMR run does not differ from a uniform grid computation at the same resolution, and is identical for patch versus (hybrid) block-based methods. Local errors, which can reach order $\mathcal{O}(1)$ near shocks or at the location of strong current sheets, do occur (see a similar observation on steady MHD shock problems using other source terms by [12]), but get averaged out on the larger scales. Quantifying the local errors on the larger run above confirms this observation, with a tendency to decrease in the later, more laminar evolution. At the same time, we reiterate the fact that these 'ideal' MHD runs are numerical proxies for resistive evolutions, and to address the small-scale reconnection events in detail needs also resistive MHD studies.

5.5. Axisymmetric relativistic magnetized jet

A computationally challenging relativistic MHD test is taken from the recent work by Leismann et al. [24], in which the first comprehensive parameter study of axisymmetric relativistic magnetized jets was presented. In contrast to most works on relativistic jet studies where purely toroidal magnetic topologies are considered (hence avoiding any $\nabla \cdot \mathbf{B}$ treatments), these authors also studied purely poloidally magnetized jets and employed a constrained transport type method [15] to handle the solenoidal constraint. We repeat their test case indicated with C2-pol-1 but use our AMR code to achieve a higher effective resolution on a laterally enlarged computational domain. Our method for controlling $\nabla \cdot \mathbf{B}$ uses the diffusive approach in the same fashion as used in the classical MHD test case from Section 5.4. This can be done since the induction equation is identical in form when going from classical to relativistic regimes.

The problem setup uses a domain in cylindrical coordinates $(R, Z) \in [0, 16] \times [0, 50]$, with 6 refinement levels at a base resolution of 40×130 achieving a maximal effective resolution of 1280×4160. The initial condition sets up a pressure-matched jet with Lorentz factor $\Gamma \sim 7$ (the jet inlet velocity is purely axial $v_{Z,\mathrm{jet}} = 0.99$) and characterized by a classical beam Mach number $M_{\mathrm{b}} = v_{Z,\mathrm{jet}}/c_{\mathrm{s,jet}} = 6$. We adopt a constant adiabatic index $\gamma = 5/3$ ideal equation of state, which makes the relativistically relevant sound speed $c_{\mathrm{s,jet}} = \sqrt{\gamma(\gamma-1)p_{\mathrm{jet}}/[(\gamma-1)\rho_{\mathrm{jet}} + \gamma p_{\mathrm{jet}}]}$. The jet density is taken at $\rho_{\mathrm{jet}} = 0.01$ while the static medium in which it is injected has $\rho_{\mathrm{ext}} = 1$. Finally, the jet and external medium are initially pervaded by a uniform axial field with a strength corresponding to unit plasma beta conditions $B_Z = \sqrt{2p_{\mathrm{jet}}}$. Boundary conditions impose the appropriate symmetries at the axis $R = 0$, leave top and lateral boundary open using continuous extrapolation on all conserved quantities to the ghost cells, and impose the jet conditions within its radius $R_{\mathrm{jet}} = 1$. Beyond this radius, we make use of symmetric density $D = \Gamma\rho$, energy density τ, and radial momentum S_{R} and axial magnetic field B_Z. The axial S_Z and radial B_{R} components are treated asymmetrically. This particular combination is typically used in astrophysically relevant jet studies, and it is meant to mimic how the presence of an accretion disk surrounding the jet engine acts as a reflective boundary to recycle the energy channeled in the developing backflows surrounding the jet beam. Refinement is enforced on a domain of size $[0, 1.2] \times [0, 0.1]$ immediately surrounding the jet inlet region, while at $t = 0$ we initialize the region $[0, R_{\mathrm{jet}}] \times [0, 0.5]$ with internal jet conditions. We trigger automated refinement on density D, as well as the energy variable τ and the auxiliary variable ξ. We run the simulation until $t = 126$, using TVDLF with minmod limiting applied on the primitive variables $(\rho, \Gamma\mathbf{v}, p, \mathbf{B})$ (note the use of the spatial part of the relativistic four-velocity). The result is shown in Fig. 10, where lab frame and rest frame densities are plotted in cross-section through the axisymmetric jet. This can be compared with Fig. 5 in [24]. At our higher effective resolution, the jet head is seen to reach $Z = 40$ at the final time, while the bow shock terminates at $Z \approx 42.5$ in their simulation. These quantitative differences are similar to those found in recent comparisons between HLL and HLLC approximate Riemann solvers for purely toroidally magnetized, relativistic jets by Mignone et al. [27].

5.6. 2.5D solar wind in spherical coordinates

A test case of an MHD solar wind application demonstrates the use of spherical coordinates. Here, we reconstruct the solar wind of Manchester et al. [29]. The full set of ideal MHD equations is solved in a cor-

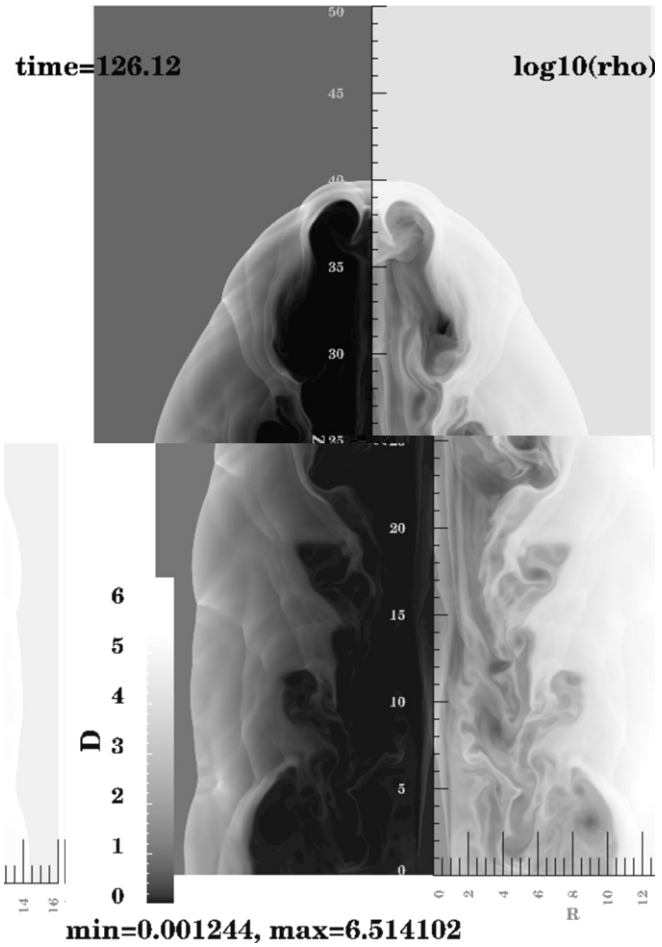


Fig. 10. For a poloidally magnetized, relativistic jet injected at Lorentz factor $\Gamma \sim 7$, we show the density D (left panel), along with the (logarithm) of the rest frame density ρ (right panel). These two quantities are related by $D = \Gamma\rho$, and we can recognize the backflow in the cocoon surrounding the jet proper (width $R_{\rm jet} = 1$), intricately structured due to shear flow instabilities.

otating frame along with an extra added gravitational force with an escape velocity 6.18×10^7 cm s^{-1} as well as an additional heating term that mimics the thermal conduction, radiative losses and other heating mechanisms. The initial condition is the hydrodynamic solar wind solution of Parker [30], with an added magnetic dipole field with field strength of $2.2G$ at the pole. At the solar base we fix the density $\rho = 1.67 \times 10^{-16}$ g cm^{-3}, the temperature $T = 1.5 \times 10^6$ K, the angular velocity such that one solar rotation corresponds with the Carrington rotation of 25.38 days, and the poloidal magnetic field components to the aforementioned dipole field. All other variables at the base are extrapolated. At the outer boundary all unknowns are extrapolated as well. The functional form of the extra heating closely resembles that of Manchester et al. [29],

$$Q = \rho q_0 (T_0 - T) \exp\left[-\frac{(r - R_\odot)^2}{\sigma_0^2}\right]. \tag{16}$$

Here, $q_0 = 10^6$ ergs g^{-1} s^{-1} K^{-1}, $R_\odot$ is the solar radius, and both T_0 and σ_0 have an r and θ dependence. The target temperature, T_0, is from a certain critical angle $\theta_0(r)$ equatorward 1.5×10^6 K and poleward 2.63×10^6 K. For $r < 7R_\odot$ this critical angle is defined by $\sin^2(\theta_0) = \sin^2(17.5^\circ) + \cos^2(17.5^\circ)(r/R_\odot - 1)/8$. For $7R_\odot < r < 47R_\odot$ this changes to $\sin^2(\theta_0) = \sin^2(61.5^\circ) + \cos^2(61.5^\circ)(r/R_\odot - 7)/40$ and for $r > 47R_\odot$ this angle stays fixed at 90°. The heating scale height σ_0 is 4.5 from the critical angle equatorward and poleward it increases as $\sigma_0 = 4.5(2 - \sin^2(\theta)/\sin^2(\theta_0))$. The adiabatic index is $\gamma = 5/3$.

The calculation is set up using a domain in 2D spherical coordinates $(r, \theta) \in [R_\odot, 30R_\odot] \times [0, \pi]$ with a base resolution of 200×30 and 4 refinement levels. The refinement is triggered by a Richardson type extrapolation on ρ, ρv_r, and B_r with weights 0.4, 0.3, and 0.3, respectively, and a tolerance criterion of $\epsilon_{\text{tol}} = 0.0005$. At the equator, $\theta = \pi/2$, we enforce refinement to make sure that the current sheet is accurately captured. The simulation uses the TVDLF scheme with a minmod limiter applied on the primitive variables. The $\nabla \cdot B$ error is handled by the diffusive approach with a diffusive constant $C_d = 1$. The initial condition is relaxed to the steady state solution shown in Fig. 11. This steady state is characterized by fast polar and slow equatorial wind properties, in direct correspondence with measured solar wind properties. Note also that the closed field lines of the helmet streamer are properly captured and correspond to static regions. These and other solar wind models will be used in full 3D AMR simulations of coronal mass ejections, to quantify the effect of the background wind model on the interplanetary CME evolution, extending the work in [22].

5.7. 3D jet in cylindrical coordinates

Our final test case demonstrates the ability to handle 3D cylindrical coordinates on a full 3D MHD problem. A helically magnetized, rotating hot jet penetrates at Mach $M \simeq 7.7$ and Alfvén Mach number $M_A \simeq 10$ into a cloud of 10 times denser material. The plasma $\beta \simeq 1$, i.e. the jet is in equipartition between thermal and magnetic pressures. We take $\gamma = 5/3$ and initialize the jet interior region where $R \leqslant R_{\text{jet}} = 1.5$ and $Z \leqslant Z_{\text{jet}} = 3$ with density $\rho_{\text{jet}} = 100$ and axial velocity $V_{Z,\text{jet}} = 1$. Within the jet region, an azimuthal field $B_\varphi = \tanh(20R)$ is adopted and the centrifugal force due to jet rotation exactly balances the magnetic tension by setting

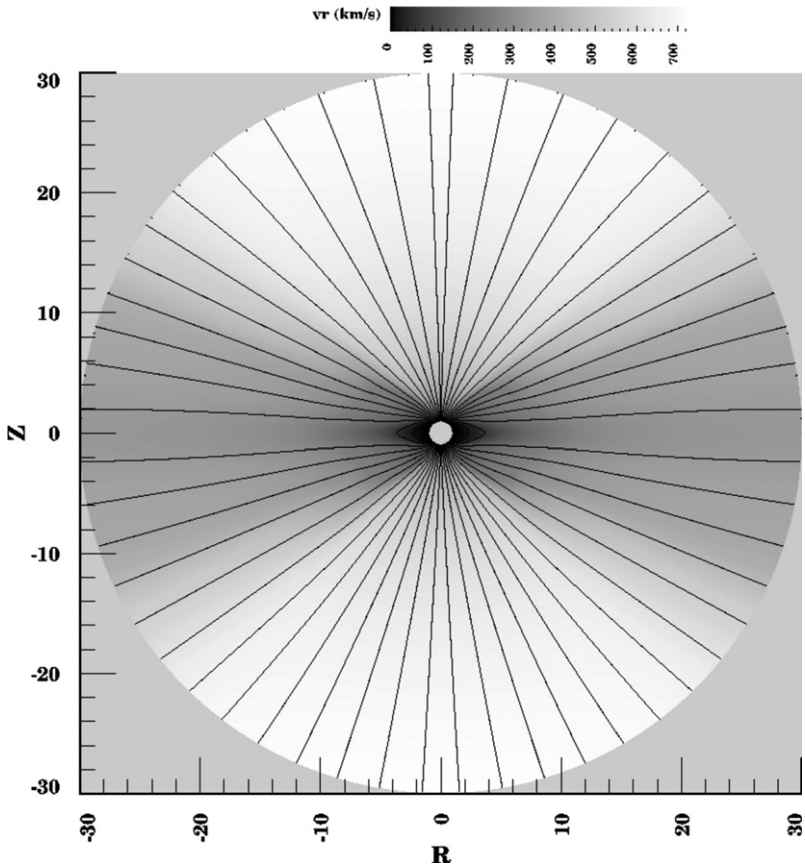


Fig. 11. In a grayscale plot, the radial velocity of the magnetized, transonic wind in a steady MHD configuration, obtained in spherical coordinates using 4 AMR levels. Solid lines indicate fieldlines, and both closed and open regions are incorporated reflecting the predominantly dipolar solar field.

$v_\varphi = B_\varphi/\sqrt{\rho_{\rm jet}}$. In the region outside the jet, the medium is static and $B_\varphi = 0$. The entire domain is pervaded by the magnetic field specified by

$$B_R = B_0 \frac{n\left(\frac{Z}{Z_{\rm jet}}\right)^{n-1} \tanh\left(\frac{Z}{Z_{\rm jet}}\right)^n \tanh\left(\frac{R}{R_{\rm jet}}\right)^2}{2\cosh\left(\frac{Z}{Z_{\rm jet}}\right)^n \frac{R}{R_{\rm jet}}} \frac{R_{\rm jet}}{Z_{\rm jet}},$$

$$B_Z = \frac{B_0}{\cosh\left(\frac{Z}{Z_{\rm jet}}\right)^n \cosh^2\left(\frac{R}{R_{\rm jet}}\right)^2} + B_{\rm c}.$$

This field is most strongly concentrated in the jet interior, and fans out radially from $Z = Z_{\rm jet}$. The pressure is then taken to ensure the remaining force balance at the lower boundary $R = 0$, namely

$$p = p_{\rm jet} + \frac{B_0^2}{2} - \frac{1}{2}(B_\varphi^2 + B_Z^2).$$

The values for the parameters giving the above mentioned Mach and plasma beta conditions are $B_0 = 1$, $B_1 = 1$, $B_{\rm c} = 0.01$ (this is the weak field characterizing the denser cloud), and $p_{\rm jet} = 1$ and we took $n = 4$. The same magnetic topology was used in the axially symmetric simulations by Casse and Marcowith [10], to study astroparticle yields from extragalactic jet terminal shocks.

We set up the simulation on a cylindrical domain of radius $R = 10$ and extended to $Z = 40$. We use 4 refinement levels and 10^3 blocksize and simulate the jet until $t = 80$, with effective resolution of $160 \times 320 \times 160$. The simulation uses TVDLF scheme with a minmod limiting on the primitive variables, and imposes the jet conditions within the bottom region $Z = 0$ and $R \leqslant R_{\rm jet}$. Periodicity applies in the φ direction, while the axis of the cylinder is handled by the π-periodic conditions. All other boundary regions are continuously extrapolated. Fig. 12 shows a snapshot at time $t = 80$. Up to this time the 3D simulation still remains axisymmetric, providing a confirmation of the expected stable evolution of the jet.

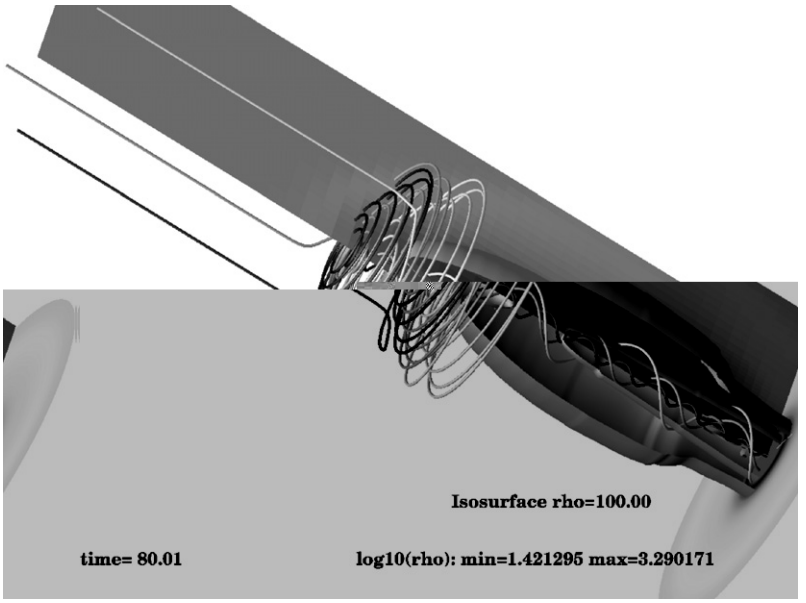


Fig. 12. 3D visualization of the hot magnetized jet penetrating a denser cloud. Shown is an isosurface of the density indicating the jet boundary, and field lines originating in the jet interior which are pushed in the shocked surrounding cloud matter.

6. Conclusions

We have presented a hybrid block-based adaptive mesh refinement scheme for general orthogonal coordinate systems. The hybrid method uses, like any tree block-based AMR scheme, fixed grid sizes in order to achieve the good cache performance on modern computer architecture. On the other hand, the hybrid scheme relaxes the common block-based approach, where a block triggered for refinement creates 2^D sub-blocks for a D-dimensional problem and refinement ratio 2. As a consequence, the grid hierarchy will have incomplete grid families. This method better approaches the optimal fit of the patch-based approach. All three AMR methods maintain the second order convergence for smooth solutions if a second order numerical scheme is used, but the hybrid scheme is the fastest with respect to the execution speed. For timings, we refer to the 2D hydrodynamic Richtmeyer–Meshkov in Section 5.2 and the planar magnetized shear layer in Section 5.4, where it is shown that the speedup is up to 30% in favor of the hybrid scheme.

Our applications have focused mainly on MHD simulations, ranging from classical to special relativistic scenarios. With this new scheme, we plan to study astrophysically relevant multi-dimensional plasma dynamics. The AMRVAC framework will be used to simulate the dynamics of fully 3D relativistic jets with realistically high Lorentz factors. The goal is to investigate the interplay of jet rotation and helical field topologies as well as the stability of the jets with respect to non-axisymmetric perturbations. Another topic of interest is in the field of space weather, where we envisage realtime space weather prediction of arrival time of coronal mass ejections on earth. Here, we need to explore the role of the solar wind on the ejecta evolution in the framework of fully 3D magnetohydrodynamics. The aim is to quantify the CME initiation parameters, such as the initial magnetic polarity, and their effect on the evolution and the geo-effectiveness of CMEs. Future work will improve on the parallel scaling by, e.g. implementing an MPI version of the present scheme or even a hybrid parallel extension using a blend of OpenMP and MPI along the lines of [14]. Another area of interest is to allow for arbitrary embedded boundaries in Cartesian geometry by implementing a cut-cell method, see e.g. [13], to facilitate the space weather calculations from sun to earth as pioneered in [20].

Acknowledgment

We acknowledge coding efforts by Z. Meliani for the relativistic MHD simulations, and fruitful collaborations with S. Poedts, C. Jacobs, H. Baty, F. Casse and D. Kimpe. Computations have been performed on the K.U. Leuven High Performance Computing cluster VIC. R.K. acknowledges financial support from the Netherlands Organization for Scientific Research, NWO Grant 614.000.421, and 'Stichting voor Fundamenteel Onderzoek der Materie' FOM. These results were obtained in the framework of the Projects GOA 2004/01 (K.U. Leuven), G.0451.05 (FWO-Vlaanderen) and C90203 (ESA Prodex 8).

Appendix

For completeness, we briefly recall the second order Total Variation Diminishing Lax–Friedrichs scheme, abbreviated as TVDLF (see [39]). It uses a predictor-corrector approach, using a Hancock predictor and where the corrector step computes time centered numerical fluxes from

$$f^{n+1/2}_{i+\frac{1}{2}} = \frac{1}{2}\left\{F(U^{\mathrm{L}}_{i+\frac{1}{2}}) + F\left(U^{\mathrm{R}}_{i+\frac{1}{2}}\right) - \left|c^{\max}\left(\frac{U^{\mathrm{L}}_{i+\frac{1}{2}} + U^{\mathrm{R}}_{i+\frac{1}{2}}}{2}\right)\right|\left[U^{\mathrm{R}}_{i+\frac{1}{2}} - U^{\mathrm{L}}_{i+\frac{1}{2}}\right]\right\}, \tag{17}$$

where the left and right cell edge states are

$$\begin{aligned} U^{\mathrm{L}}_{i+\frac{1}{2}} &= U^{n+1/2}_i + \overline{\Delta U}^{n+1/2}_i/2, \\ U^{\mathrm{R}}_{i+\frac{1}{2}} &= U^{n+1/2}_{i+1} - \overline{\Delta U}^{n+1/2}_{i+1}/2. \end{aligned} \tag{18}$$

The overbar denotes the slope limiter, with e.g. the 'minmod' limiter given by

$$\overline{\Delta U}_i = \mathrm{sgn}(U_i - U_{i-1})\max\left[0, \min\{|U_i - U_{i-1}|, (U_{i+1} - U_i)\mathrm{sgn}(U_i - U_{i-1})\}\right]. \tag{19}$$

The stencil of this TVDLF scheme is 5 cells wide, through the slope limited linear reconstruction. This scheme is generally applicable to any system of non-linear conservation laws, by inserting the appropriate maximal physical propagation speed $c^{\max}$. For the non-relativistic MHD cases, $c^{\max} = |v| + c$ and c denotes the fast magnetosonic speed. The TVDLF method is also referred to as Local Lax–Friedrichs method, and minor variants use the maximum of each edge state separately instead of the arithmetic average. A first order variant was exploited as early as in 1961 by Rusanov [32], to numerically solve several shock-dominated 2D gas dynamic problems. In the second order variant, slight improvements on its somewhat diffusive character occur when performing the linear reconstruction on the primitive variables ρ, $\mathbf{v}$, p and $\mathbf{B}$ instead of the conservative variables $\mathbf{U}$. The more elaborate TVD method is a dimensionally split, onestep method, which exploits a Roe type approximate Riemann solver. As it exploits a full characteristic based decomposition, it improves the capturing of contact discontinuities and shocks, at a correspondingly higher computational cost.

References

[1] D.S. Balsara, Second-order accurate schemes for magnetohydrodynamics with divergence-free reconstruction, Astrophys. J. Suppl. Ser. 151 (2004) 149.
[2] H. Baty, R. Keppens, P. Comte, The two-dimensional magnetohydrodynamic Kelvin–Helmholtz instability: compressibility and large-scale coalescence effects, Phys. Plasmas. 10 (2003) 4661.
[3] J. Bell, M. Berger, J. Saltzman, M. Welcome, Three-dimensional adaptive mesh refinement for hyperbolic conservation laws, SIAM J. Sci. Comp. 15 (1994) 127.
[4] M.J. Berger, Data structures for adaptive grid generation, SIAM J. Sci. Stat. Comput. 7 (1986) 904.
[5] M.J. Berger, P. Colella, Local adaptive mesh refinement for shock hydrodynamics, J. Comput. Phys. 82 (1989) 64.
[6] M.J. Berger, I. Rigoutsos, An algorithm for point clustering and grid generation, IEEE Trans. Systems, Man Cybernetics 21 (1991) 1278.
[7] J. Bergmans, R. Keppens, D.E.A. van Odyck, A. Achterberg, Simulations of relativistic astrophysical flows, in: T. Plewa, T. Linde, V.G. Weirs (Eds.), Adaptive Mesh Refinement – Theory and Applications, Lect. Not. Comp. Sci. Eng., vol. 41, 2005, pp. 223.
[8] Berkeley Lab AMR homepage. <http://seesar.lbl.gov/>.
[9] A.C. Calder, B. Fryxell, T. Plewa, R. Rosner, L.J. Dursi, V.G. Weirs, T. Dupont, H.F. Robey, J.O. Kane, B.A. Remington, R.P. Drake, G. Dimonte, M. Zingale, F.X. Timmes, K. Olson, P. Ricker, P. MacNeice, H.M. Tufo, On validating an astrophysical simulation code, Astrophys. J. Suppl. Ser. 143 (2002) 201.
[10] F. Casse, A. Marcowith, Astroparticle yield and transport from extragalactic jet terminal shocks, Astroparticle Phys. 23 (2005) 31.
[11] A. van Dam, P.A. Zegeling, A robust moving mesh finite volume method applied to 1D hyperbolic conservation laws from magnetohydrodynamics, J. Comput. Phys. 216 (2006) 526.
[12] H. De Sterck, A. Csik, D. Vanden Abeele, S. Poedts, H. Deconinck, Stationary two-dimensional magnetohydrodynamic flows with shocks: characteristic analysis and grid convergence study, J. Comput. Phys. 166 (2001) 28.
[13] D. De Zeeuw, K.G. Powell, An adaptive refined cartesian mesh solver for the Euler equations, J. Comput. Phys. 104 (1993) 56.
[14] J. Dreher, R. Grauer, Racoon: a parallel mesh-adaptive framework for hyperbolic conservation laws, Par. Comp. 31 (2005) 913.
[15] C.R. Evans, J.F. Hawley, Simulation of magnetohydrodynamic flows – a constrained transport method, Astrophys. J. 332 (1988) 659.
[16] FLASH user guide. <http://flash.uchicago.edu>.
[17] H. Friedel, R. Grauer, C. Marliani, Adaptive mesh refinement for singular current sheets in incompressible magnetohydrodynamic flows, J. Comput. Phys. 134 (1997) 190.
[18] A.L. Garcia, J.B. Bell, W.Y. Crutchfield, B.J. Alder, Adaptive mesh and algorithm refinement using direct simulation Monte Carlo, J. Comput. Phys. 154 (1999) 134.
[19] T.I. Gombosi, K.G. Powell, D.L. de Zeeuw, C.R. Clauer, K.C. Hansen, W.B. Manchester, A.J. Ridley, I.I. Roussev, I.V. Sokolov, Q.F. Stout, G. Tóth, Solution-adaptive magnetohydrodynamics for space plasmas: sun-to-earth simulations, Comput. Sci. Eng. March/April issue, 14 (2004).
[20] C.P.T. Groth, D.L. De Zeeuw, T.I. Gombosi, K.G. Powell, Global three-dimensional MHD simulation of a space weather event: CME formation, interplanetary propagation, and interaction with the magnetosphere, J. Geophys. Res. 105 (2000) 25053.
[21] A. Harten, High resolution schemes for hyperbolic conservation laws, J. Comput. Phys. 49 (1983) 357.
[22] C. Jacobs, S. Poedts, B. van der Holst, The effect of the solar wind on CME triggering by magnetic foot point shearing, Astron. Astrophys. 450 (2006) 793.
[23] R. Keppens, Z. Meliani, Grid-adaptive simulations of relativistic flows, in: Proceedings of ICCFD4, July 2006, Ghent, Belgium.
[24] T. Leismann, L. Antón, M.A. Aloy, E. Müller, J.M. Martí, J.A. Miralles, J.M. Ibáñez, Relativistic MHD simulations of extragalactic jets, Astron. Astrophys. 436 (2005) 503.
[25] S. Li, H. Li, A novel approach of divergence-free reconstruction for adaptive mesh refinement, J. Comput. Phys. 199 (2004) 1.
[26] B. Marder, A method for incorportaing Gauss' law into electromagnetic PIC codes, J. Comput. Phys. 68 (1987) 48.
[27] A. Mignone, S. Massaglia, G. Bodo, Relativistic MHD simulations of jets with toroidal magnetic fields, Space Sci. Rev. 121 (2005) 21.
[28] R. Keppens, M. Nool, G. Tóth, J.P. Goedbloed, Adaptive mesh refinement for conservative systems: multi-dimensional efficiency evaluation, Comput. Phys. Commun. 153 (2003) 317.

[29] W.B. Manchester, T.I. Gombosi, I. Roussev, A. Ridley, D.L. de Zeeuw, I.V. Sokolov, K.G. Powell, G. Tóth, Modeling a space weather event from the sun to the earth: CME generation and interplanetary propagation, J. Geophys. Res. 109 (2004) A02107, doi:10.1029/2003JA010150.
[30] E.N. Parker, Dynamics of the interplanetary gas and magnetic fields, Astrophys. J. 128 (1958) 664.
[31] J.J. Quirk, A contribution to the great Riemann solver debate, Int. J. Numer. Methods Fl. 18 (1994) 555.
[32] V.V. Rusanov, The calculation of the interaction of non-stationary shock waves and obstacles, USSR Comput. Math. Math. Phys. 1 (1961) 304.
[33] SAMRAI Project at LLNL. <http://www.llnl.gov/casc/SAMRAI>.
[34] M. Torrilhon, Non-uniform convergence of finite volume schemes for Riemann problems of ideal magnetohydrodynamics, J. Comput. Phys. 192 (2003) 73.
[35] M. Torrilhon, Uniqueness conditions for Riemann problems of ideal magnetohydrodynamics, J. Plasma Phys. 69 (2003) 253.
[36] M. Torrillhon, D.S. Balsara, High order WENO schemes: investigations on non-uniform convergence for MHD Riemann problems, J. Comput. Phys. 201 (2004) 586.
[37] G. Tóth, The LASY preprocessor and its application to general multi-dimensional codes, J. Comput. Phys. 138 (1997) 981.
[38] G. Tóth, A general code for modeling MHD flows on parallel computers: versatile advection code, Astrophys. Lett. Commun. 34 (1996) 245, See http://www.phys.uu.nl/~toth.
[39] G. Tóth, D. Odstrčil, Comparison of some flux corrected transport and total variation diminishing numerical schemes for hydrodynamic and magnetohydrodynamic problems, J. Comput. Phys. 128 (1996) 82.
[40] P.R. Woodward, P. Colella, The numerical simulation of two-dimensional fluid flow with strong shocks, J. Comput. Phys. 54 (1984) 174.
[41] H.C. Yee, A class of high-resolution explicit and implicit shock-capturing methods, NASA TM-101088, 1989.
[42] U. Ziegler, Self-gravitational adaptive mesh magnetohydrodynamics with the NIRVANA code, Astron. Astrophys. 435 (2005) 385.