Hydro and MHD Solvers in FLASH: How to use them

Dongwook Lee

Flash Center for Computational Science
The University of Chicago

FLASH Tutorial
Rutherford Appleton Laboratory, UK, May 31 – Jun 1, 2012

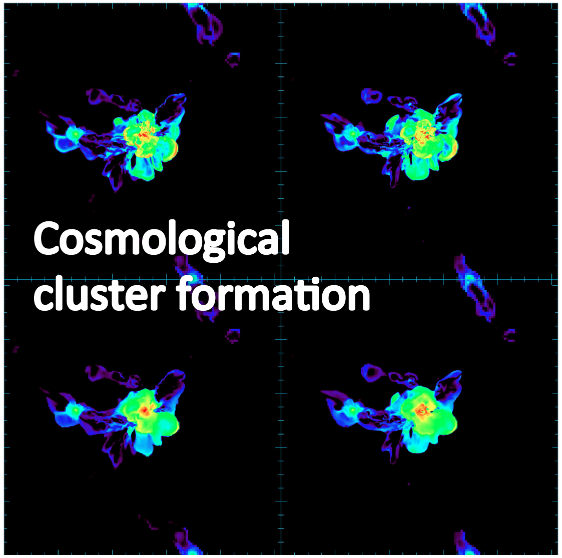
Outline

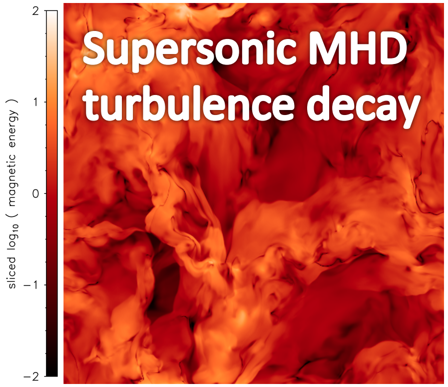
- **Broad Ranges of Applications**
  - Astrophysics and HEDP
- **Algorithms and mathematics**
  - Governing equations, difference equations of PDEs
  - Dimensionally Split vs. Unsplit formulations
  - Hydro and MHD solvers
    - High-order Reconstructions, Riemann problems, physical units
  - Explicit vs. Implicit schemes
  - Beyond the hyperbolic system (diffusion, source terms, Biermann battery, etc)
- **Examples**: setting up problems, how to use various features and switches
- **Summary**

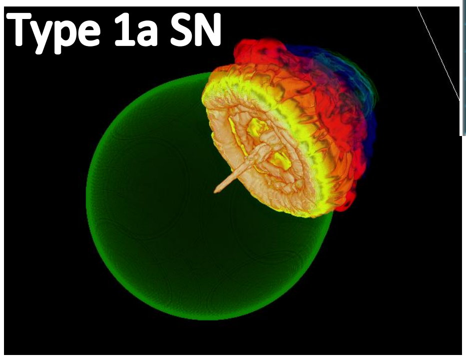
Coupling Hydro/MHD with Astrophysics and HEDP

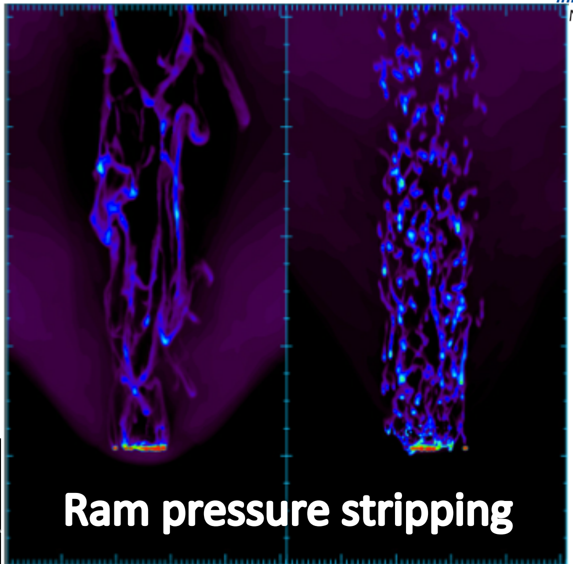
- **Single-fluid description Hydrodynamics, MHD, RHD**
  - *Multitemperature extension for hydro/MHD*
- **Equations of State**
  - Gamma law, multigamma, Helmholtz, reltativistic ideal gamma
  - *Multitemperature (gamma, multigamma, tabulated, multitype)*
- **Nuclear physics and source terms**
  - Burn, ionization, stir, gravity
  - *Laser energy deposition, heat exchange, Biermann battery*
- **Active and passive particles**
- **Material properties**
  - Thermal conductivity, magnetic diffusivity, viscosity
  - *Opacity, operator split (semi) implicit solver for diffusive terms*
- **Cosmology**
- *Radiative tranfer: multigroup diffusion*

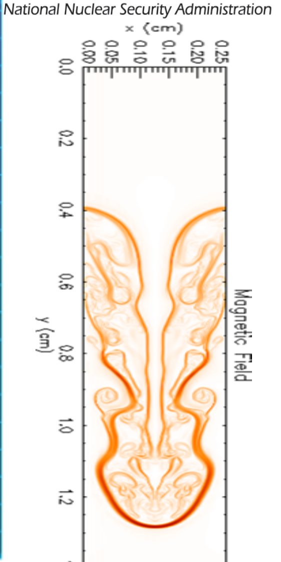
Astrophysical Applications



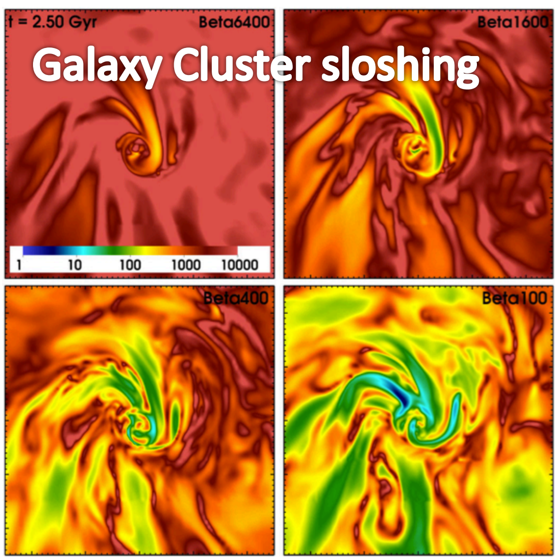


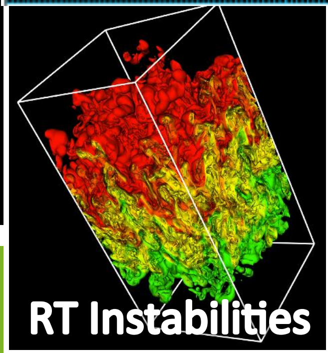


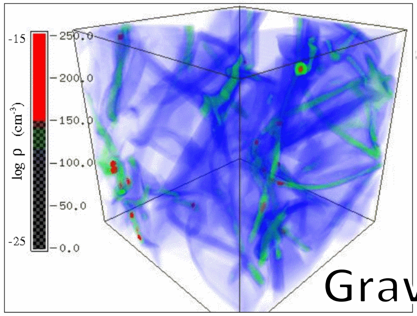




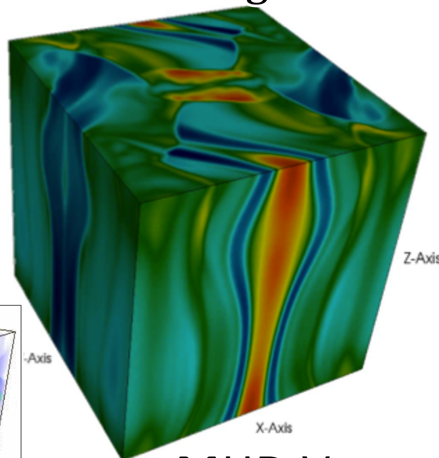
Magnetic RT





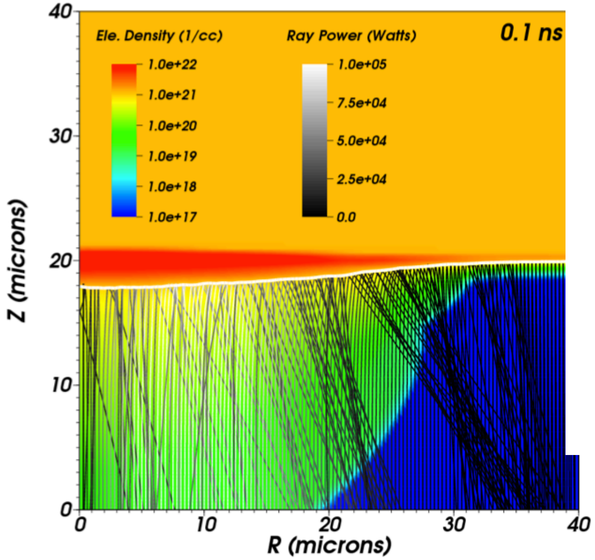


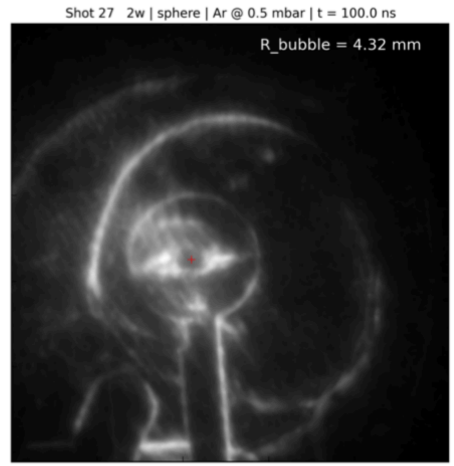
Gravitational Collapse

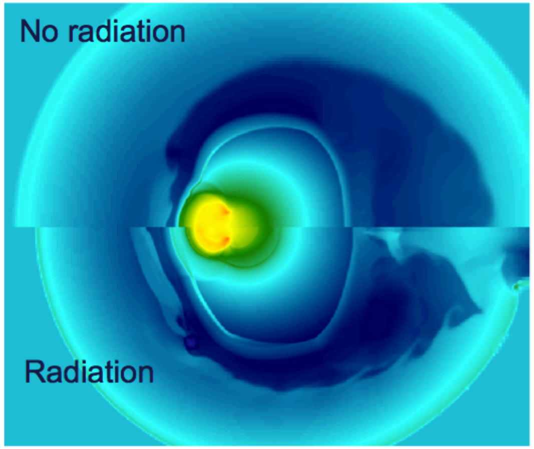


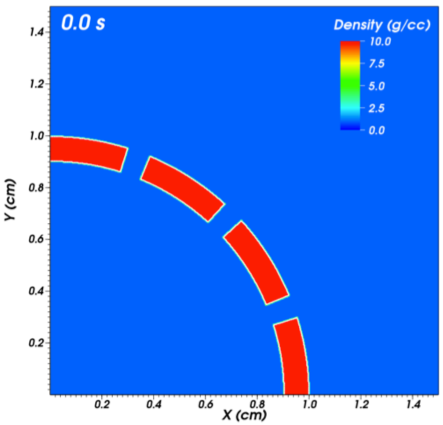
MHD Vortex

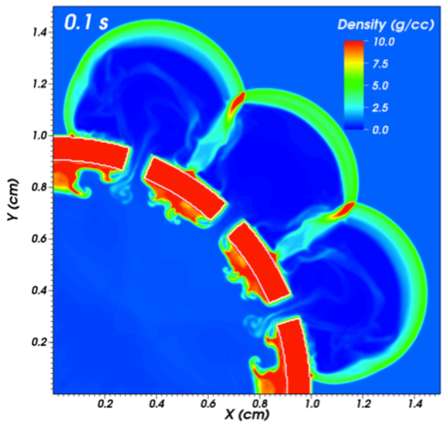
HEDP Applications

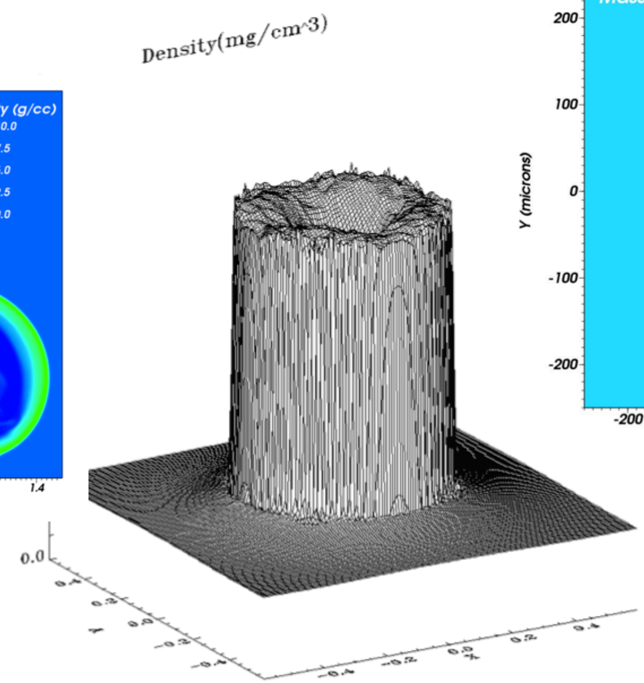


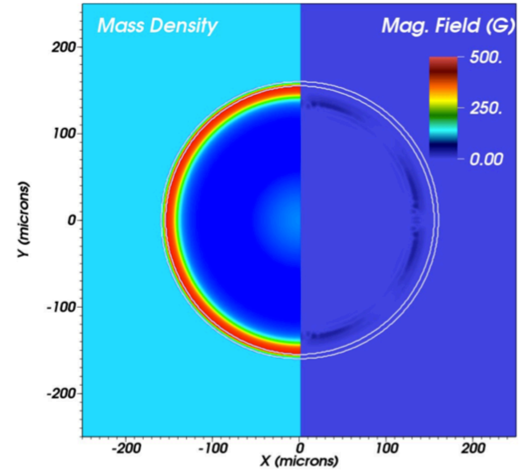




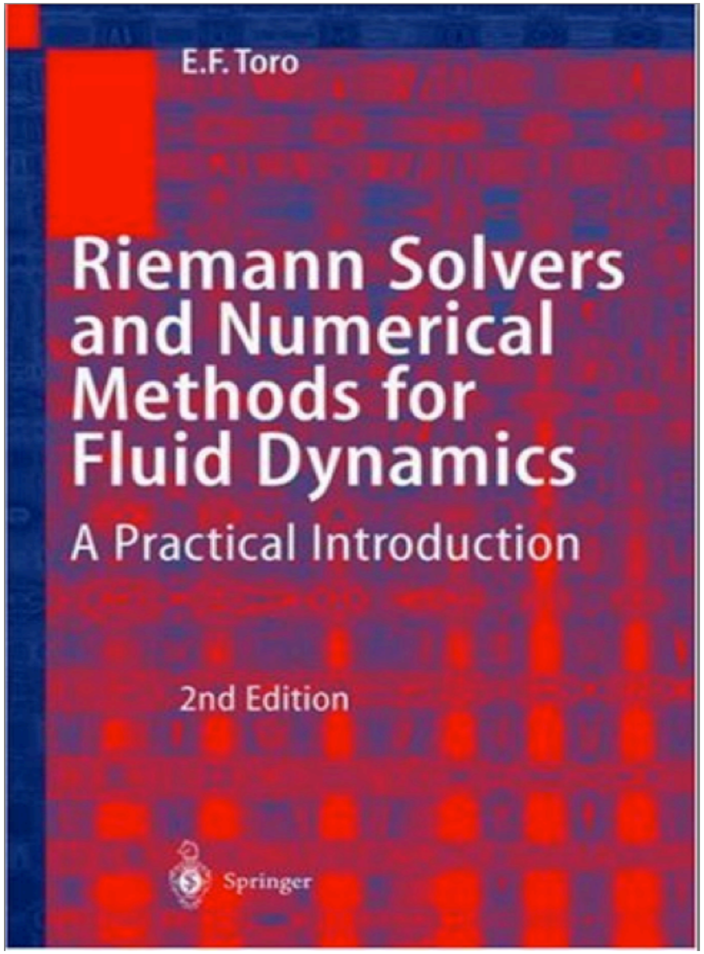


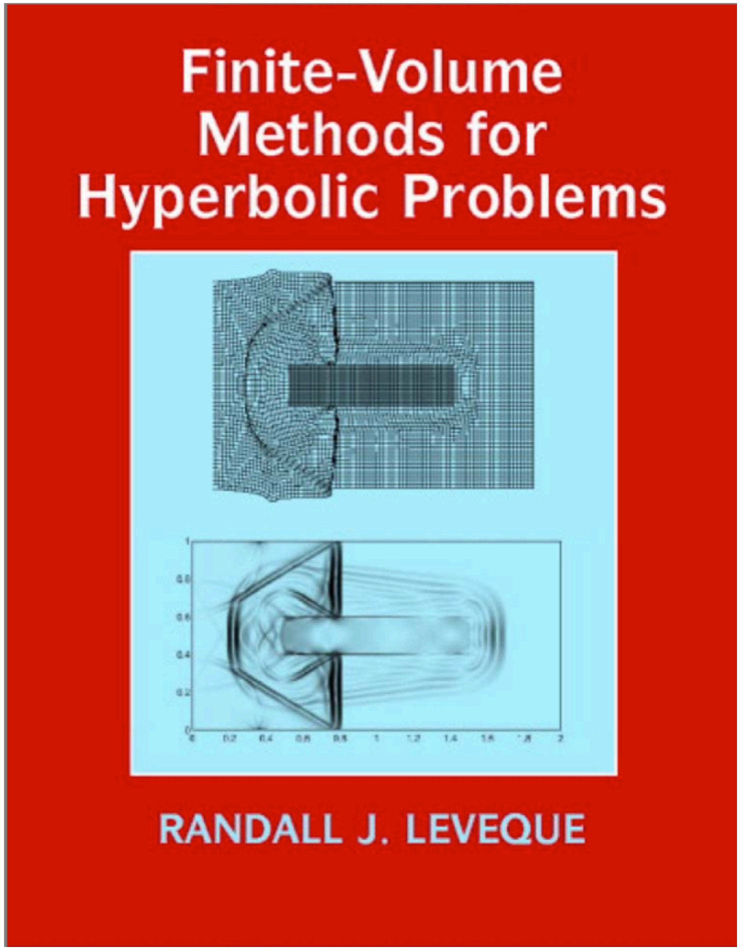






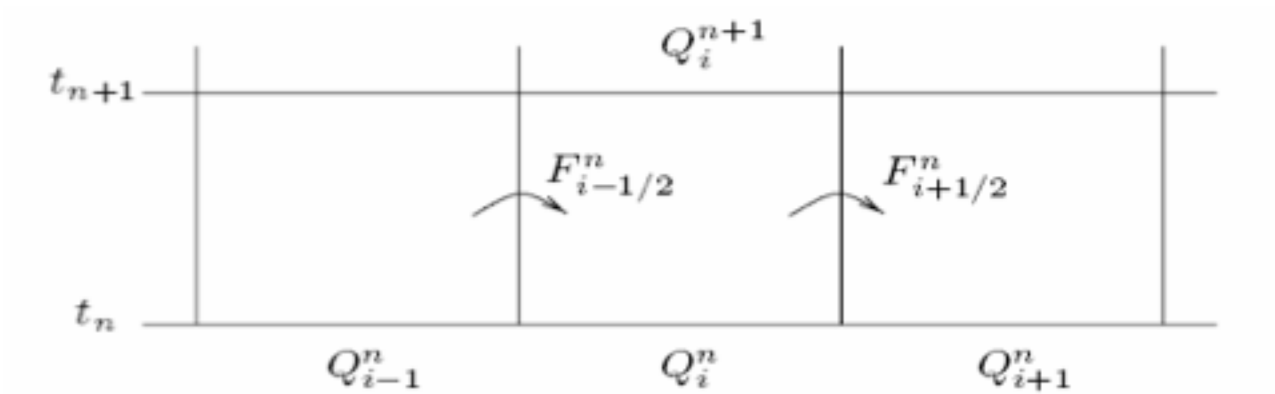
Two Excellent Books





Algorithms and Formulations

- **FLASH's hydro/MHD solves conservation laws using a finite-volume (FV) approach**
  - Highly compressible flows with shocks and discontinuities
  - Differential (smooth) form of PDE becomes invalid
  - Integral form of PDE relaxes the smoothness assumptions and seeks for weak solutions over control volumes and their boundaries
- **Basics of FV formulation (1D):**



$$Q_i^n \approx \frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_{i+1/2}} q(x,t_n)\,dx \equiv \frac{1}{\Delta x}\int_{\mathcal{C}_i} q(x,t_n)\,dx, \qquad F_{i-1/2}^n \approx \frac{1}{\Delta t}\int_{t_n}^{t_{n+1}} f\left(q\left(x_{i-1/2},t\right)\right)dt.$$

Algorithms and Formulations

- **Integral Form of PDE:**

$$\int_{C_i} q(x, t_{n+1})\,dx - \int_{C_i} q(x, t_n)\,dx = \int_{t_n}^{t_{n+1}} f\left(q\left(x_{i-1/2}, t\right)\right) dt - \int_{t_n}^{t_{n+1}} f\left(q\left(x_{i+1/2}, t\right)\right) dt.$$

- **Volume averaged cell centered quantity and time averaged flux:**

$$Q_i^n \approx \frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_{i+1/2}} q(x, t_n)\,dx \equiv \frac{1}{\Delta x}\int_{C_i} q(x, t_n)\,dx,$$

$$F_{i-1/2}^n \approx \frac{1}{\Delta t}\int_{t_n}^{t_{n+1}} f\left(q\left(x_{i-1/2}, t\right)\right) dt.$$

- **Finite wave speed in hyperbolic system:**

$$F_{i-1/2}^n = \mathcal{F}(Q_{i-1}^n, Q_i^n)$$

* High-order reconstruction in space & time

* Riemann problems at each interface

- **General difference equation in conservation form:**

$$Q_i^{n+1} = Q_i^n - \frac{\Delta t}{\Delta x}\left(F_{i+1/2}^n - F_{i-1/2}^n\right),$$

Algorithms and Formulations

- Multidimensional hyperbolic system in conservation laws:

$$Q_{ij}^n \approx \frac{1}{\Delta x\, \Delta y} \int_{y_{j-1/2}}^{y_{j+1/2}} \int_{x_{i-1/2}}^{x_{i+1/2}} q(x, y, t_n)\, dx\, dy.$$

$$F_{i-1/2,j}^n \approx \frac{1}{\Delta t\, \Delta y} \int_{t_n}^{t_{n+1}} \int_{y_{j-1/2}}^{y_{j+1/2}} f(q(x_{i-1/2}, y, t))\, dy\, dt,$$

$$G_{i,j-1/2}^n \approx \frac{1}{\Delta t\, \Delta x} \int_{t_n}^{t_{n+1}} \int_{x_{i-1/2}}^{x_{i+1/2}} g(q(x, y_{j-1/2}, t))\, dx\, dt.$$

- 2D discrete form:

$$Q_{ij}^{n+1} = Q_{ij}^n - \frac{\Delta t}{\Delta x}[F_{i+1/2,j}^n - F_{i-1/2,j}^n] - \frac{\Delta t}{\Delta y}[G_{i,j+1/2}^n - G_{i,j-1/2}^n],$$

- Two approaches:
  - Directionally "split"
  - Directionally "unsplit"

Split Formulation

- Directionally split (1st order Godunov; 2nd order Strang splitting)

$$\begin{aligned} \text{PDE} &: \quad \mathbf{U}_t + \mathbf{F}(\mathbf{U})_x + \mathbf{G}(\mathbf{U})_y = \mathbf{0}\,, \\ \text{IC} &: \quad \mathbf{U}(x,y,t^n) = \mathbf{U}^n\,. \end{aligned}$$

$$\left.\begin{aligned} \text{PDEs} &: \quad \mathbf{U}_t + \mathbf{F}(\mathbf{U})_x = \mathbf{0} \\ \text{ICs} &: \quad \mathbf{U}^n \end{aligned}\right\} \overset{\Delta t}{\Longrightarrow} \mathbf{U}^{n+\frac{1}{2}}$$

$\rightarrow$

$$\left.\begin{aligned} \text{PDEs} &: \quad \mathbf{U}_t + \mathbf{G}(\mathbf{U})_y = \mathbf{0} \\ \text{ICs} &: \quad \mathbf{U}^{n+\frac{1}{2}} \end{aligned}\right\} \overset{\Delta t}{\Longrightarrow} \mathbf{U}^{n+1}\,.$$

  - 1st order:

$$\mathbf{U}^{n+1} = \mathcal{Y}^{(\Delta t)}\mathcal{X}^{(\Delta t)}(\mathbf{U}^n)$$ or

$$\mathbf{U}^{n+1} = \mathcal{X}^{(\Delta t)}\mathcal{Y}^{(\Delta t)}(\mathbf{U}^n)$$

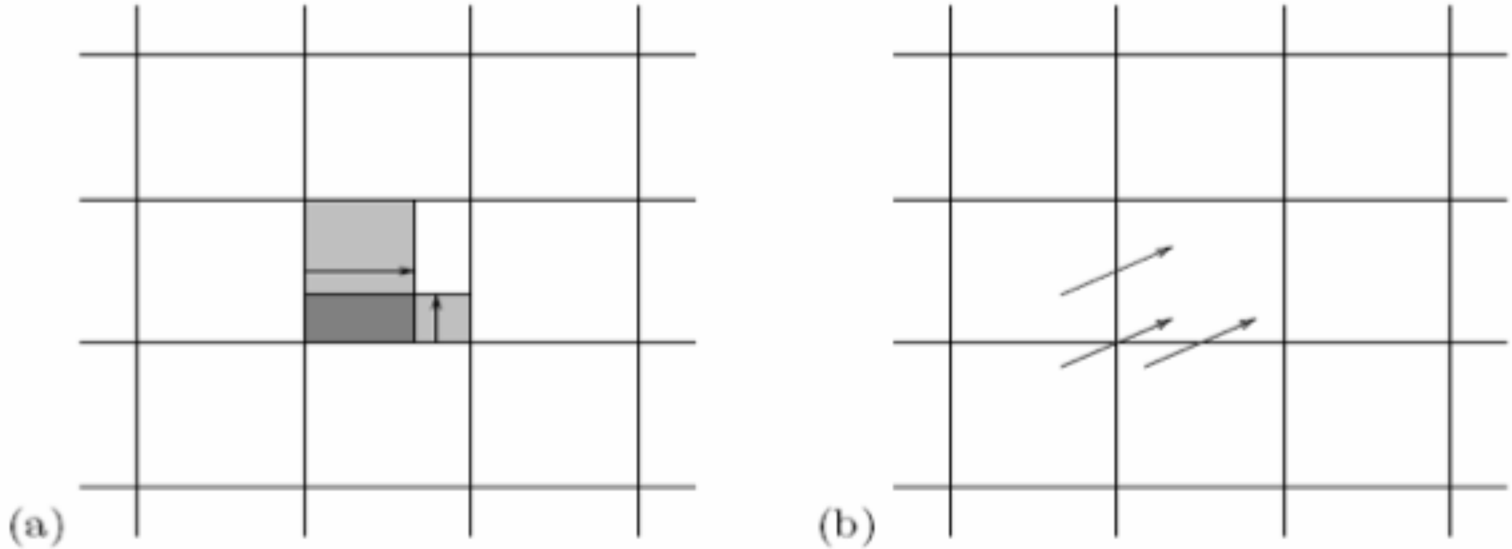
  - 2nd order:

$$\mathbf{U}^{n+1} = \frac{1}{2}\left[\mathcal{X}^{(\Delta t)}\mathcal{Y}^{(\Delta t)} + \mathcal{Y}^{(\Delta t)}\mathcal{X}^{(\Delta t)}\right](\mathbf{U}^n)$$

- Directional splitting scheme is easy to implement to extend 1D to multidimensional scheme. It is robust and accurate in general.

Unsplit Formulations

- Directionally unplit (1st order Donor Cell; 2nd order Corner-Transport-Upwind)
  - 1st order Donor cell vs. 2nd order CTU



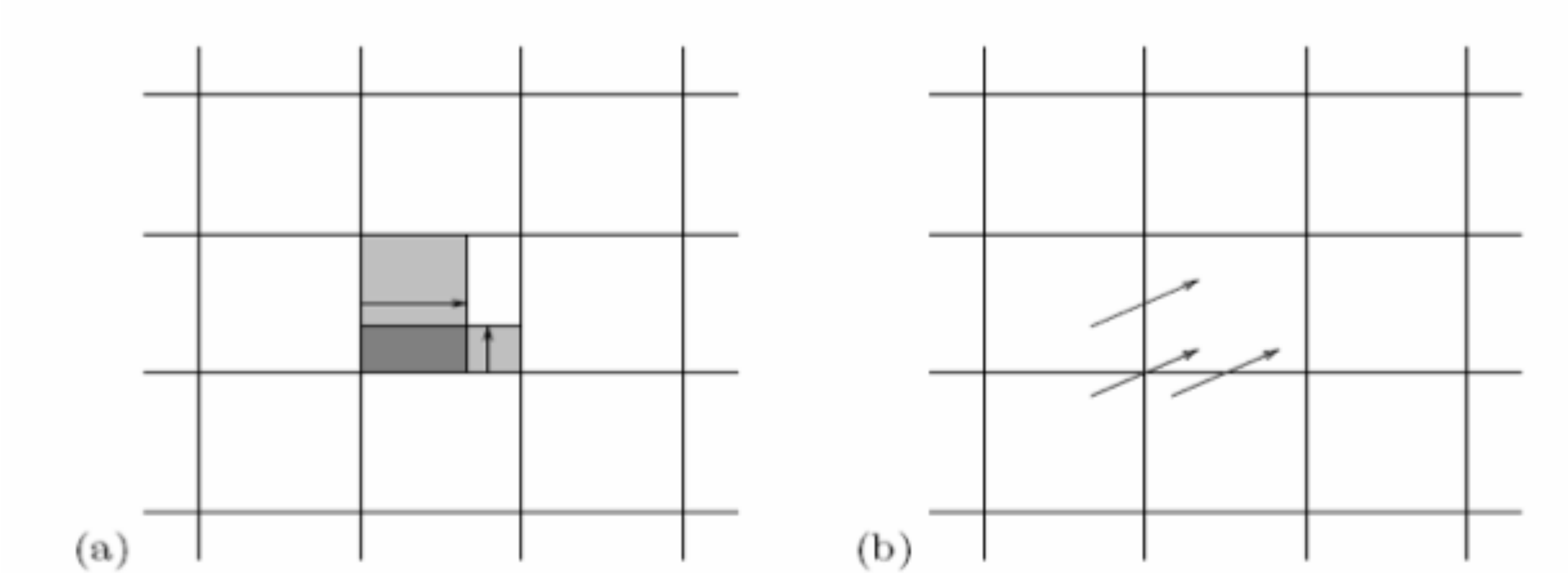
  - Stability limit:

$$\left|\frac{u\,\Delta t}{\Delta x}\right| + \left|\frac{v\,\Delta t}{\Delta y}\right| \leq 1.$$

$$\max\left(\left|\frac{u\,\Delta t}{\Delta x}\right|, \left|\frac{v\,\Delta t}{\Delta y}\right|\right) \leq 1.$$

Unsplit Formulations

- Directionally unplit (1st order Donor Cell; 2nd order Corner-Transport-Upwind)
  - 1st order Donor cell vs. 2nd order CTU

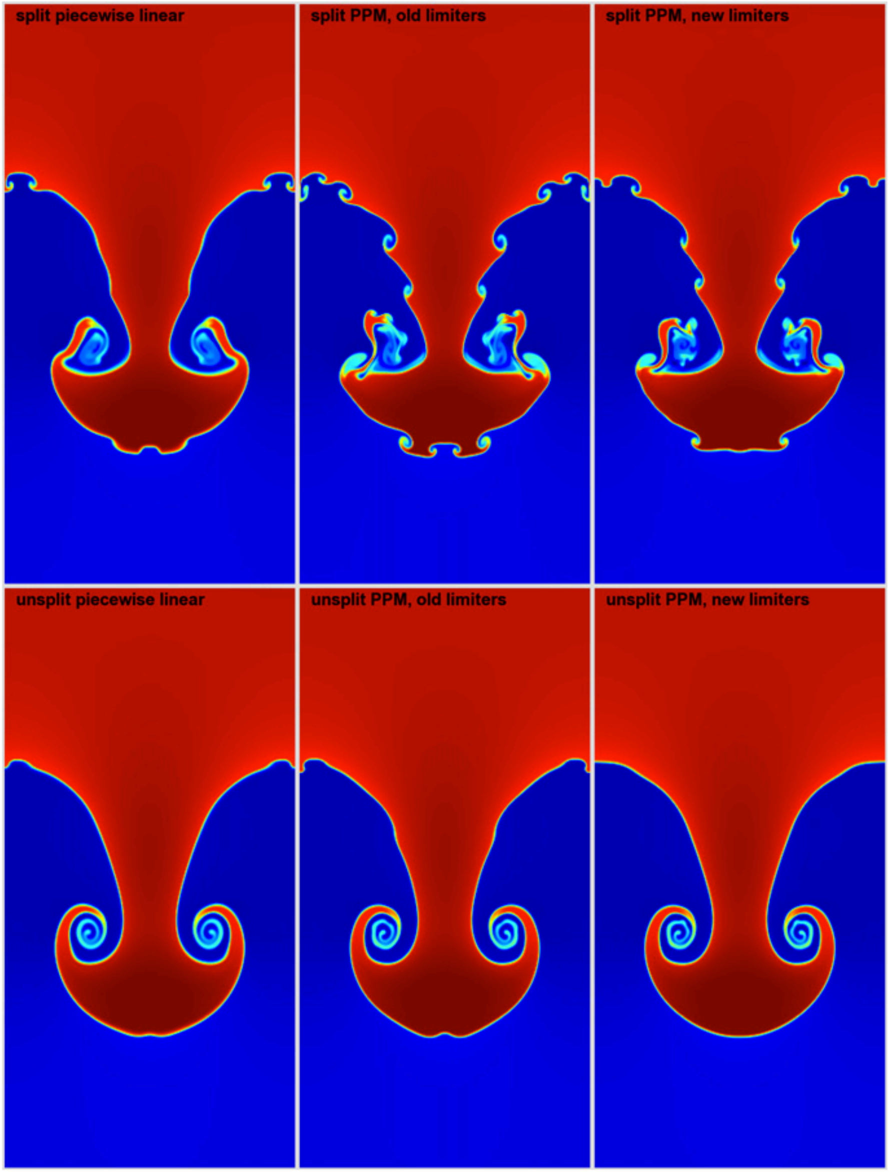


  - Stencil:

$$Q_{i,j}^{n+1} = Q_{i,j}^{n} - \frac{u\Delta t}{\Delta x}\left[Q_{i,j}^{n} - Q_{i-1,j}^{n}\right] - \frac{v\Delta t}{\Delta y}\left[Q_{i,j}^{n} - Q_{i,j-1}^{n}\right]$$

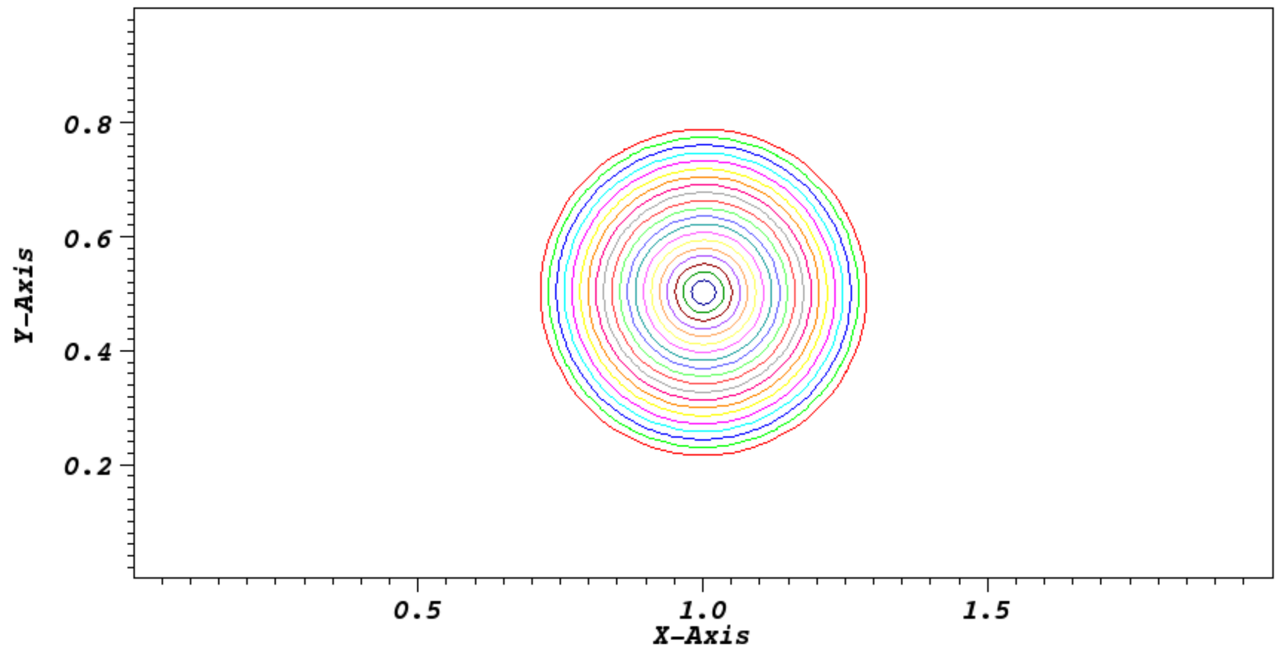
$$Q_{i,j}^{n+1} = Q_{i,j}^{n} - \frac{u\Delta t}{\Delta x}\left[Q_{i,j}^{n} - Q_{i-1,j}^{n}\right] - \frac{v\Delta t}{\Delta y}\left[Q_{i,j}^{n} - Q_{i,j-1}^{n}\right]$$
$$+\frac{\Delta t^2}{2}\left\{\frac{u}{\Delta x}\left[\frac{v}{\Delta y}\left(Q_{i,j}^{n} - Q_{i,j-1}^{n}\right) - \frac{v}{\Delta y}\left(Q_{i-1,j}^{n} - Q_{i-1,j-1}^{n}\right)\right]\right.$$
$$\left.+\frac{v}{\Delta y}\left[\frac{u}{\Delta x}\left(Q_{i,j}^{n} - Q_{i-1,j}^{n}\right) - \frac{u}{\Delta x}\left(Q_{i,j-1}^{n} - Q_{i-1,j-1}^{n}\right)\right]\right\}$$

Hydro Comparison between Split vs. Unsplit



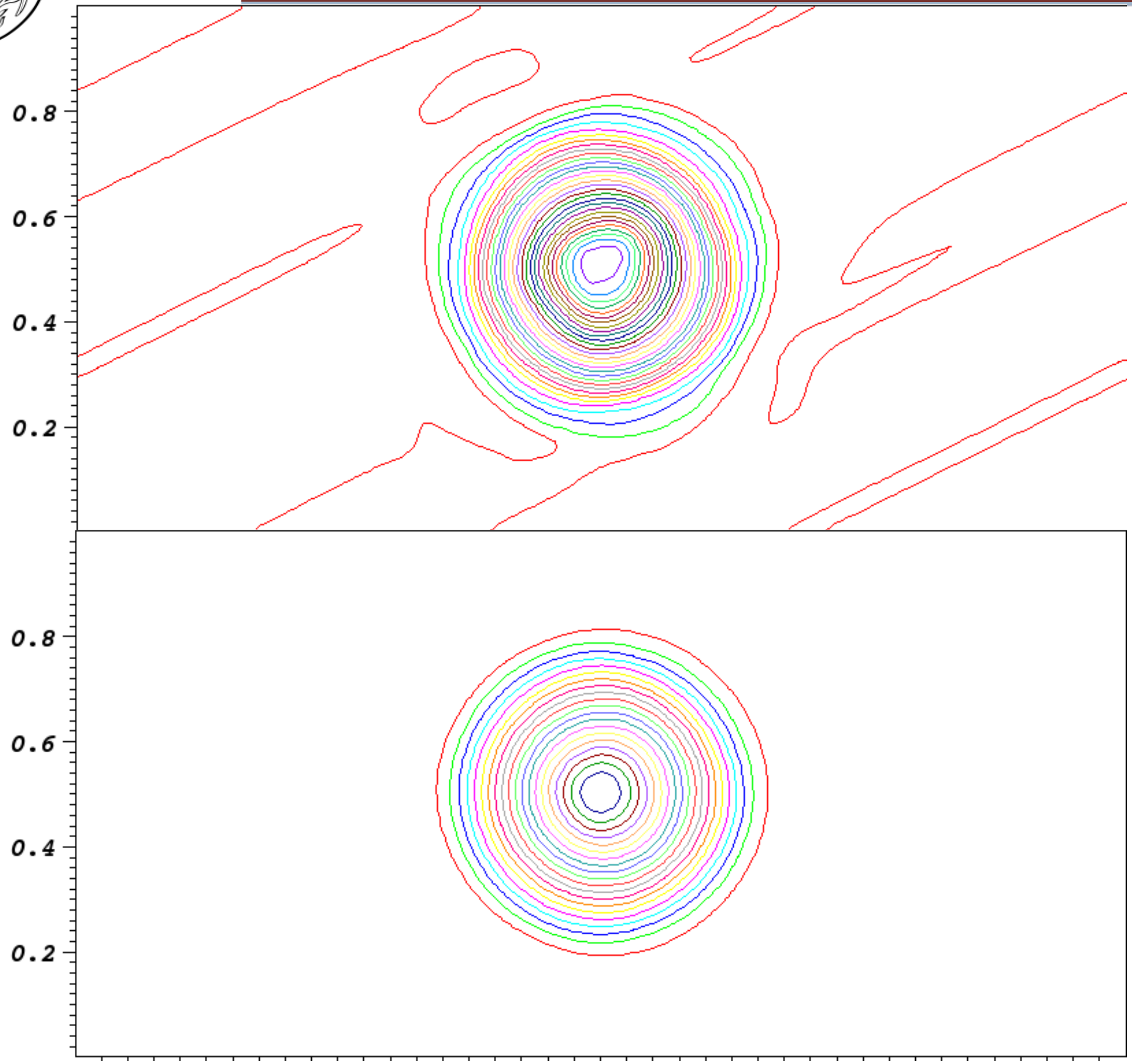
- Single-mode Rayleigh-Taylor instability by Almgren et al (ApJ, 715, 2010)
- Top: split schemes
  - PLM (2nd order)
  - PPM + old slope limiter (3rd order)
  - PPM + new slope limiter (3rd order)
  - high-wavenumber instability grows due to experiencing high compression and expansion in each directional sweep
- Bottom: unsplit schemes
  - PLM
  - PPM + old slope limiter
  - PPM + new slope limiter
  - High-wavenumber instabilities are suppressed

MHD Comparison between Split vs. Unsplit



- Weakly magnetized Field loop advection in 2.5D
- Gardiner & Stone 2005 (JCP); Lee and Deane 2009 (JCP)

MHD Comparison between Split vs. Unsplit



- 8-wave split MHD scheme (Powell et al. 1999) at t=2.0
- Unsplit staggered mesh MHD scheme (Lee and Deane, 2009) at t=2.0

MHD Comparison between Split vs. Unsplit

- What is wrong with the split formulation for MHD?
- In the split formulation, you cannot correctly include terms proportional t $\nabla \cdot \mathbf{B}$
    - Gardiner and Stone (2005)
- Dynamics of in-plane magnetic fields in x and y directions are ruined from erroneous growth of magnetic field in z direction:

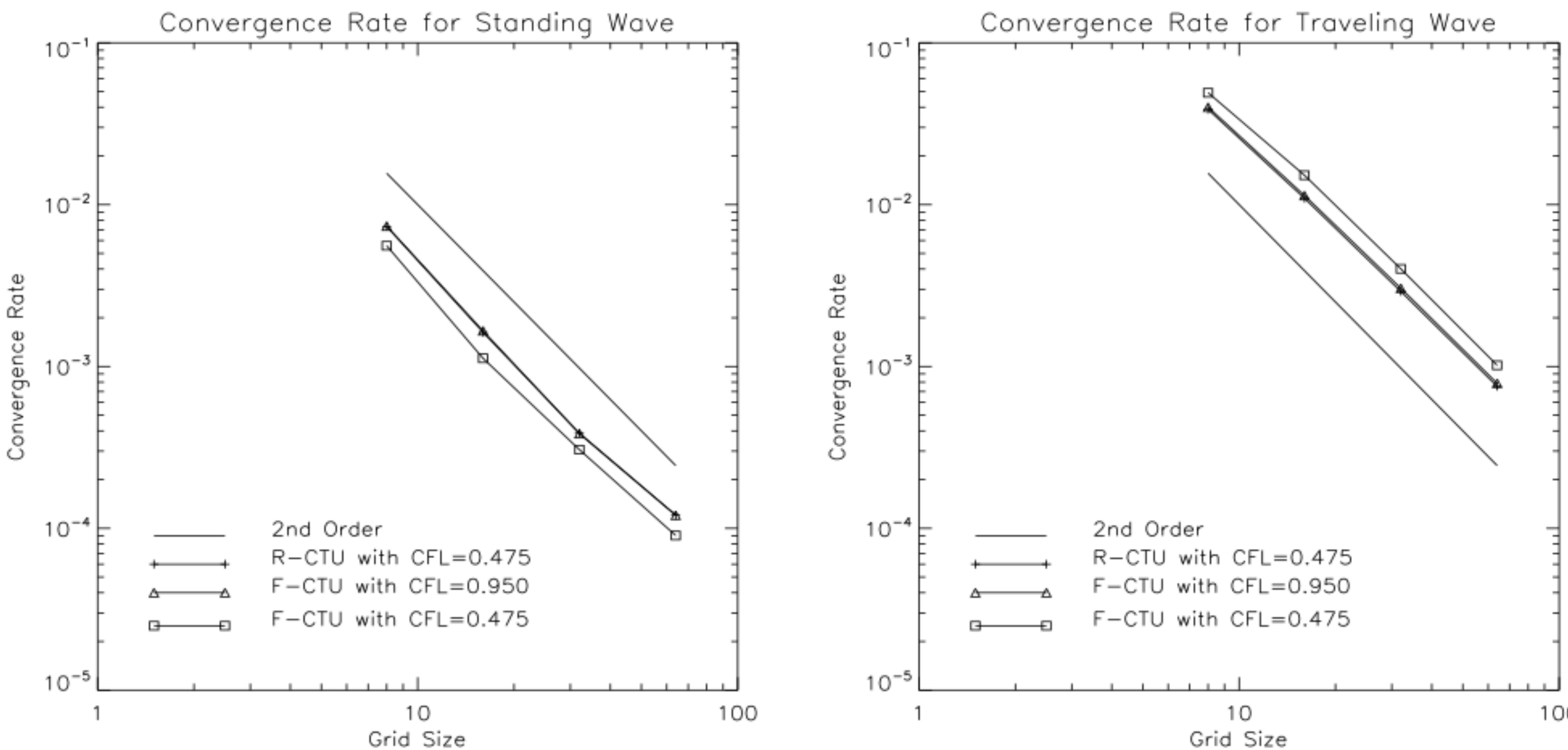
$$\frac{\partial B_z}{\partial t} + B_z \frac{\partial u}{\partial x} - B_x \frac{\partial w}{\partial x} - \boxed{w \frac{\partial B_x}{\partial x}} + u \frac{\partial B_z}{\partial x} + B_z \frac{\partial v}{\partial y} - B_y \frac{\partial w}{\partial y} - \boxed{w \frac{\partial B_y}{\partial y}} + v \frac{\partial B_z}{\partial y} = 0$$

$$\boxed{w \nabla \cdot \mathbf{B} = w(\Delta B_{x,i}/\Delta x + \Delta B_{y,j}/\Delta y).}$$

Convergence, Stability, and Consistency

- **Lax equivalence theorem (for linear problem; 1956)**
  - ***The only convergent schemes are those that are both consistent and stable!***
  - **Hard to show that the numerical solution converges to the original solution of the PDE; relatively easy to show consistency and stability of numerical schemes**
- **In practice, non-linear problems adopts the linear theory as guidance**
  - **Analytical solution if any**
  - **Grid resolution (self-convergent) test**

Grid resolution test for smooth solution

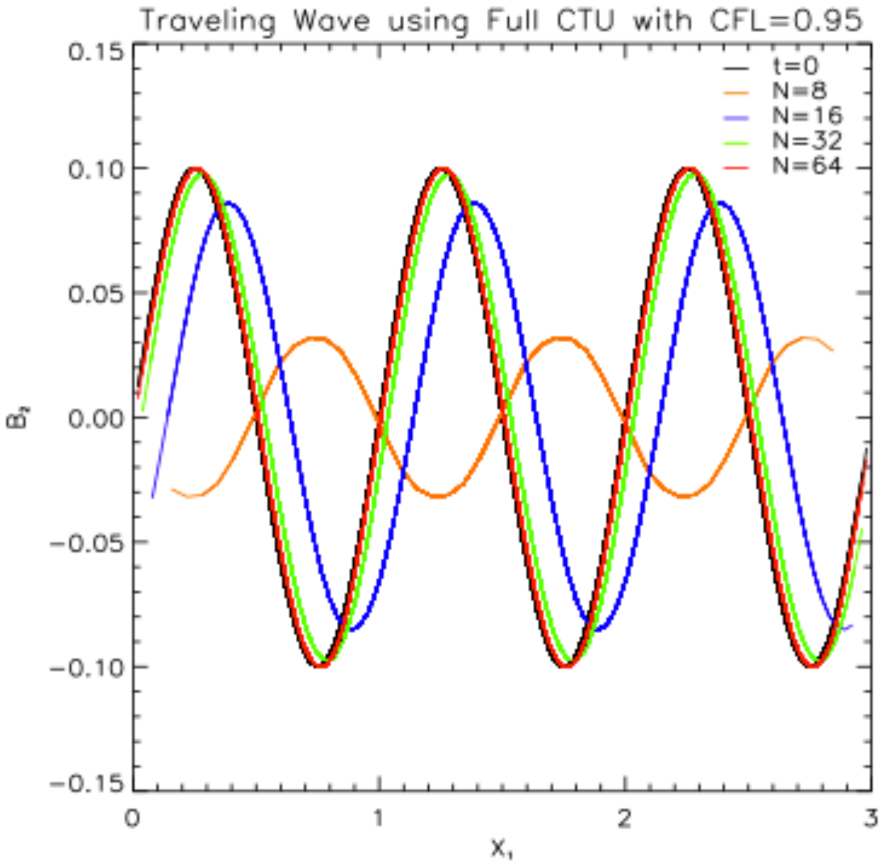


(a) Convergence rate for the standing wave solutions at $t = 1.0$

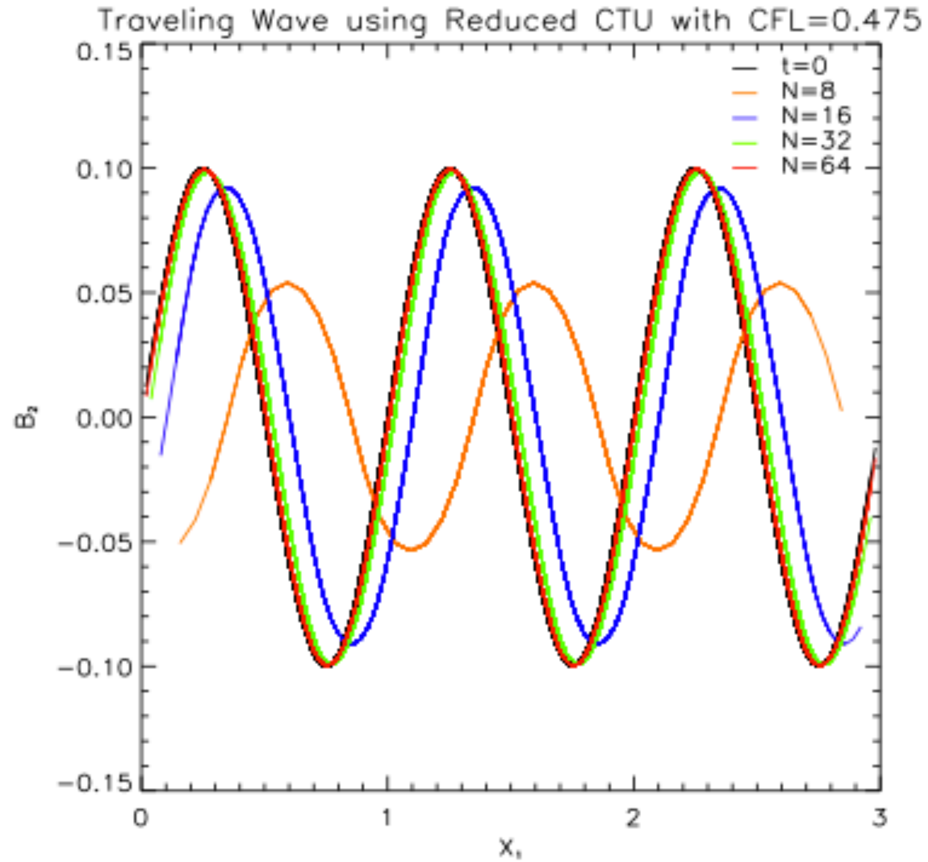
(b) Convergence rate for the traveling wave solutions at $t = 1.0$

Fig. 8. The circularly polarized Alfvén wave convergence rate for both the standing and traveling wave problems. PPM is used along with the HLLD Riemann solver.

Comparison with exact solution



(a) Scatter image of B_2 plotted with respect to x_1 −axis at time $t = 5$.



(b) Scatter image of B_2 plotted with respect to x_1 −axis at time $t = 5$.

Fig. 10. Plot of B_2 versus x_1 for the traveling wave using (a) the full 3D CTU scheme with CFL=0.95, and (b) the reduced 3D CTU scheme with CFL=0.475.The initial condition using $N = 64$ is also plotted.

Riemann Problems and Godunov Method

The Riemann problem:

$$\text{PDEs:} \quad \mathbf{U}_t + \mathbf{A}\mathbf{U}_x = \mathbf{0}\,, \; -\infty < x < \infty\,, \; t > 0\,,$$

$$\text{IC:} \quad \mathbf{U}(x,0) = \mathbf{U}^{(0)}(x) = \begin{cases} \mathbf{U}_\mathrm{L} & x < 0\,, \\ \mathbf{U}_\mathrm{R} & x > 0 \end{cases}$$

Two cases:

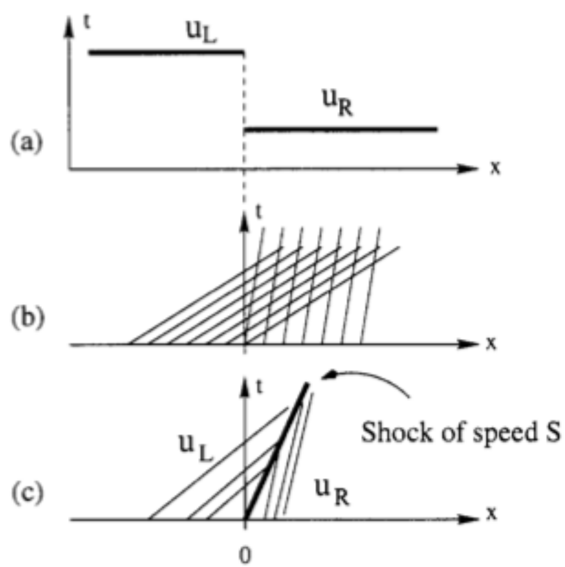


Fig. 2.13. (a) Compressive discontinuous initial data (b) picture of charac and (c) solution on x–t plane

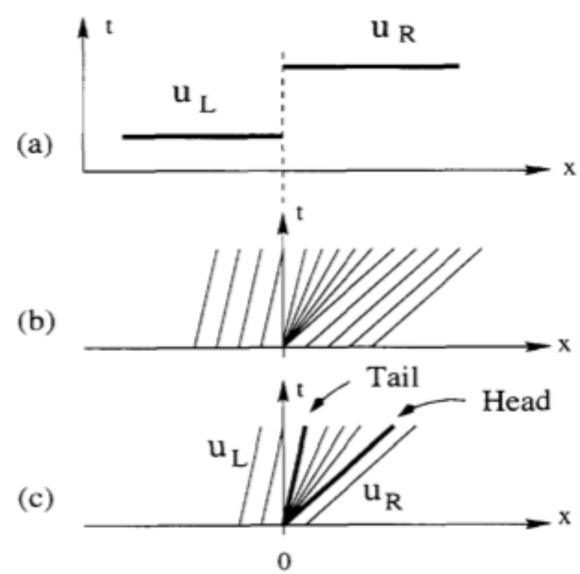
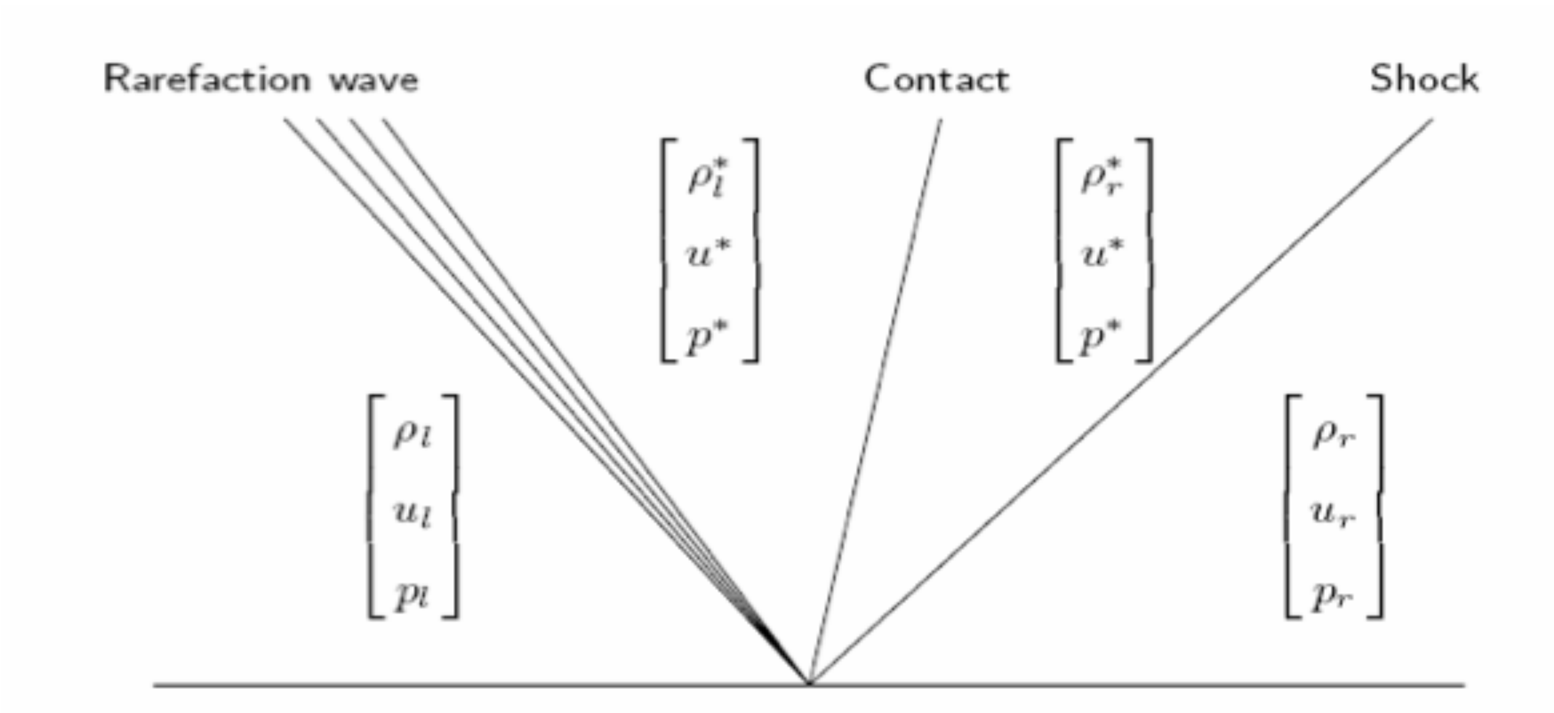


Fig. 2.16. Centred rarefaction wave: (a) expansive discontinuous initial data (b) picture of characteristics (c) entropy satisfying (rarefaction) solution on x–t plane

Riemann Problems and Godunov Method

- The Riemann fan:



Riemann Problems and Godunov Method

- **Godunov's innovative method (1959) to solve non-linear conservative system using the exact solution of the Riemann problem at intercell boundaries**

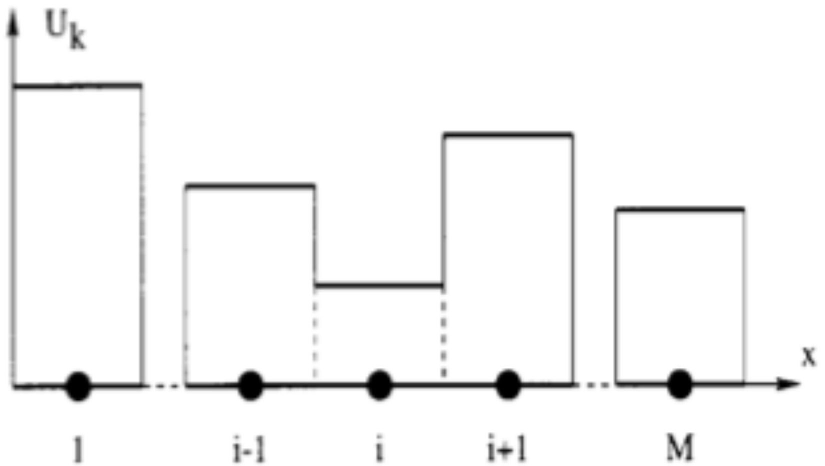


Fig. 6.1. Piece-wise constant distribution of data at time level n, for a single component of the vector **U**

Riemann Problems and Godunov Method

- Godunov's innovative method (1959) to solve non-linear conservative system using the exact solution of the Riemann problem at intercell boundaries

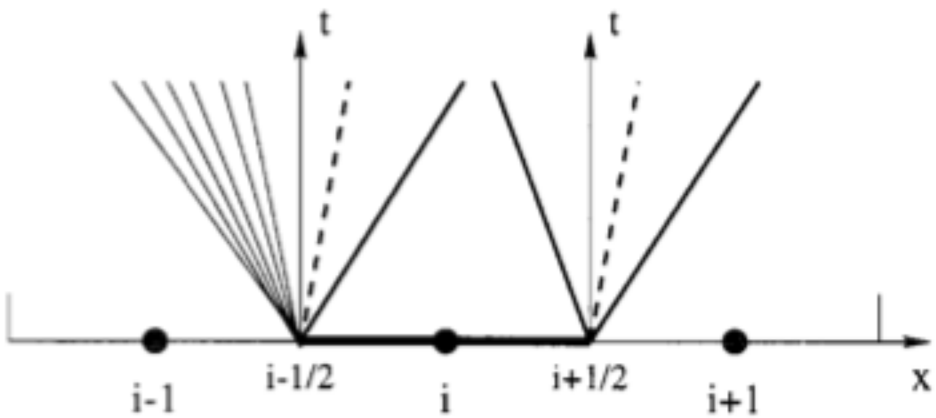


Fig. 6.2. Typical wave patterns emerging from solutions of local Riemann problems at intercell boundaries $i - \frac{1}{2}$ and $i + \frac{1}{2}$

Godunov Theorem and high-order schemes

- Godunov's "order-barrier" theorem (1959) says:
  - If an advection scheme (of PDE) preserves the monotonicity of the solution it is at most first-order accurate!
  - Second or higher order schemes are NOT monotone and will generate oscillations
  - Discouraging and seems to be doomed to improve any advection scheme of PDE!
  - Linear theory is assumed!

- Good! High-resolution scheme became possible using non-linear schemes
  - 70's and 80's: Boris, van Leer, Zalesak, Woodward, Colella, Harten, Shu, Engquist, etc.
  - Use of slope limiters (e.g., Koren, van Leer)
  - MUSCL-Hancock, Piecewise Parabolic Method (PPM), ENO, WENO, etc.

Godunov Theorem and high-order schemes

Slope limiters:

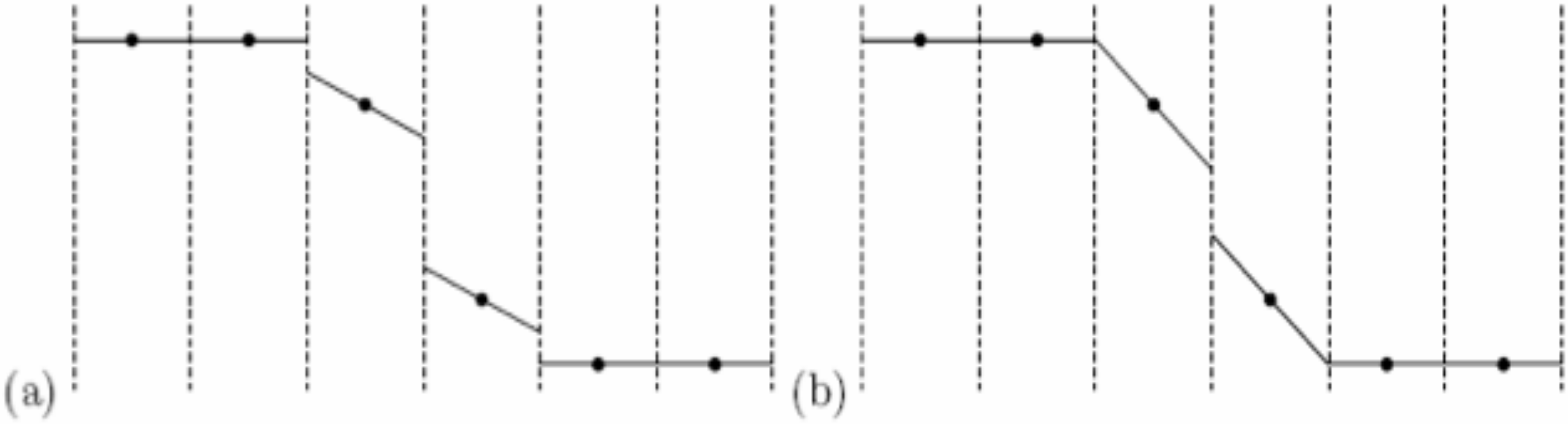
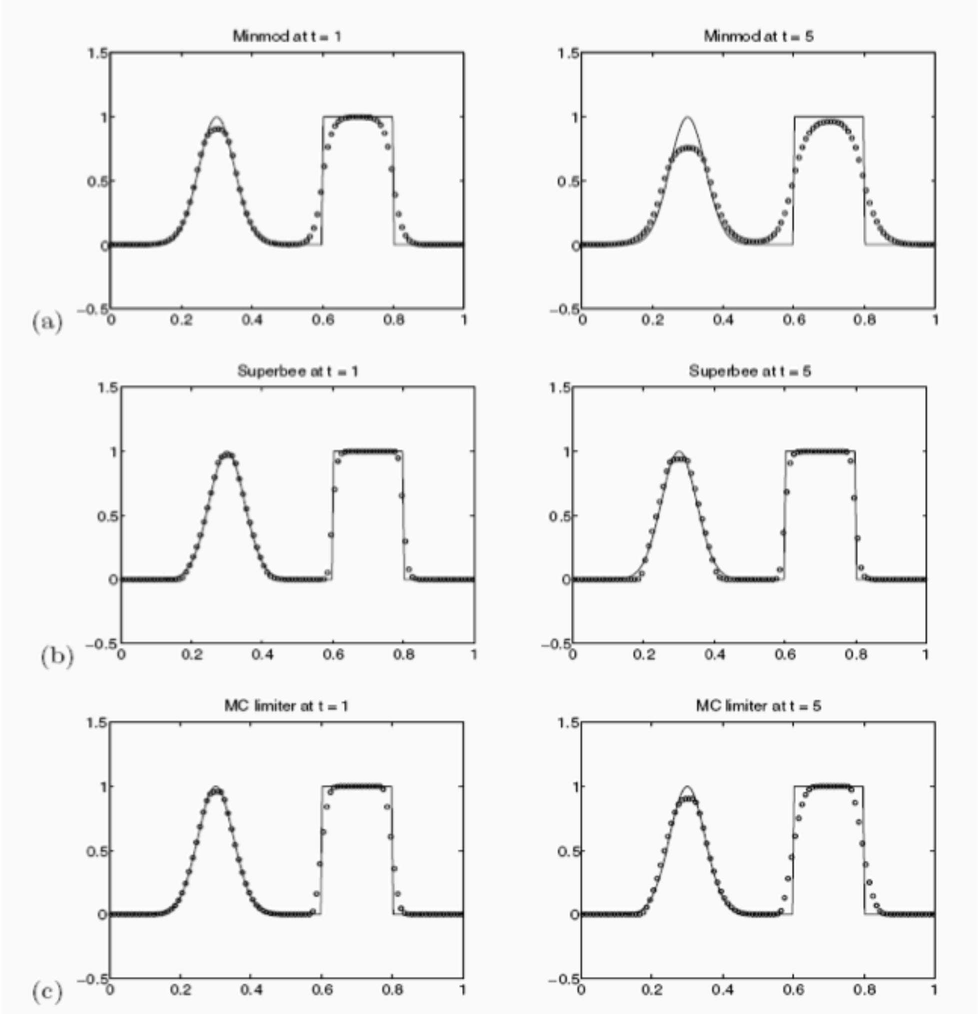


Fig. 6.5. Grid values Q^n and reconstructed $\tilde{q}^n(\cdot, t_n)$ using (a) minmod slopes, (b) superbee or MC slopes. Note that these steeper slopes can be used and still have the TVD property.

Godunov Theorem and high-order schemes

- Slope limiters:



- Examples:
  - Minmod, van Leer, MC, Superbee

Riemann Solvers

- Exact Riemann solvers
  - Involves iterations for pressure to seek for a solution over the Riemann fan
  - Very accurate in general but it can be defective without converging

- Approximate Riemann solvers
  - No iterations needed
  - Rusanov (local Lax-Friedrichs)
  - HLL*-type (HLLE, HLLC for hydro/MHD; HLLD for MHD)
  - Roe solver
  - Hybrid method (e.g., HLL + Roe; HLL + HLLD)
  - Plays a crucial role to determine solution stability and accuracy!

Riemann Solvers

LLF

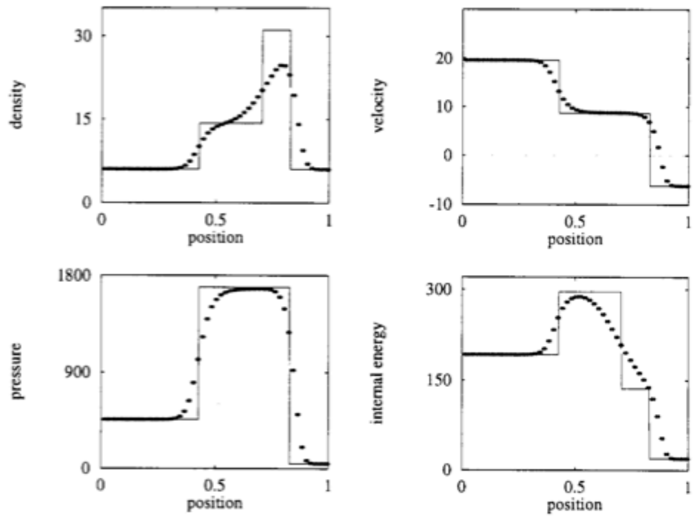


Fig. 6.16. The Lax-Friedrichs method applied to Test 4, with $x_0 = 0.4$. Numerical (symbol) and exact (line) solutions are compared at the output time 0.035 units

HLL

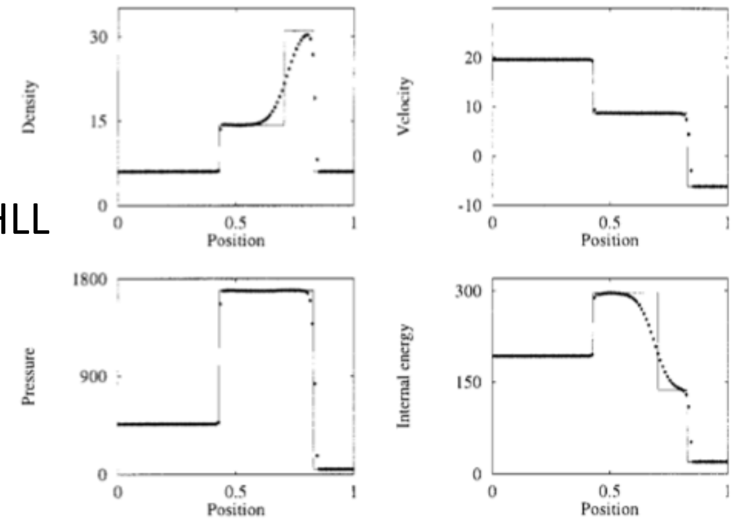


Fig. 10.13. Godunov's method with HLL Riemann solver applied to Test 4, with $x_0 = 0.4$. Numerical (symbol) and exact (line) solutions are compared at time 0.035

HLLC

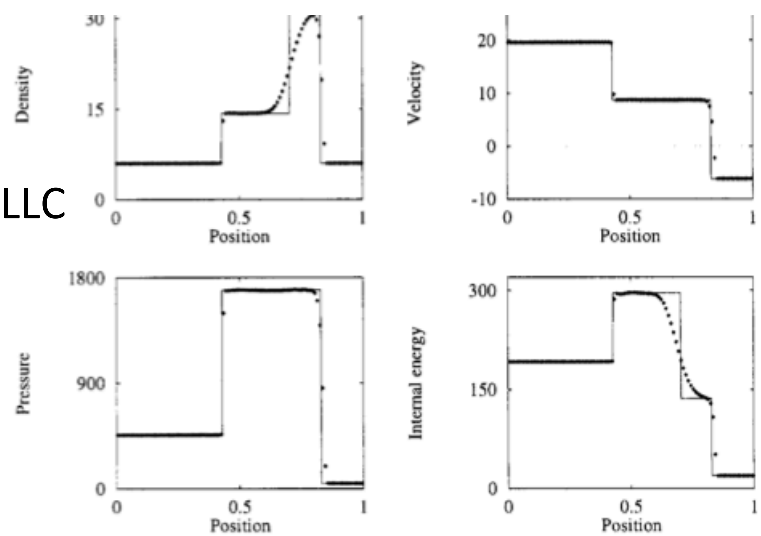


Fig. 10.8. Godunov's method with HLLC Riemann solver applied to Test 4, with $x_0 = 0.4$. Numerical (symbol) and exact (line) solutions are compared at time 0.035

Roe

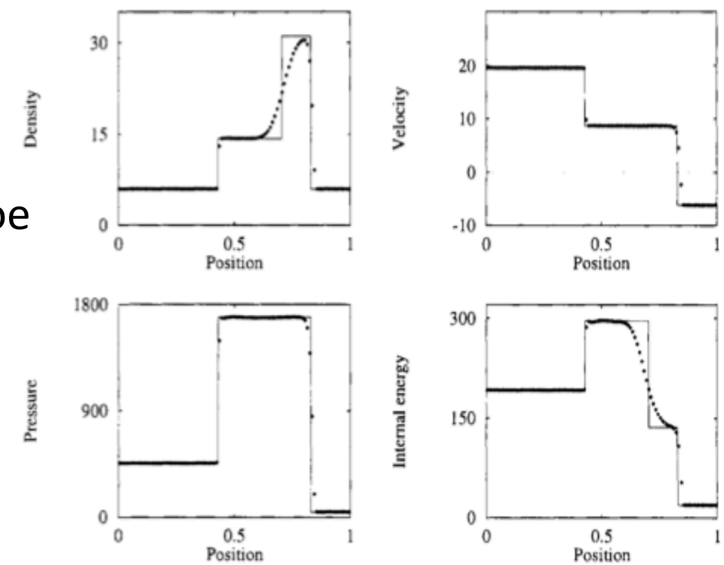


Fig. 11.7. Godunov's method with Roe's Riemann solver applied to Test 4, with $x_0 = 0.4$. Numerical (symbol) and exact (line) solutions are compared at time 0.035

Diffusive Terms and Implicit Treatments

- **Fact:**

$$\Delta t_{\text{Diffusion}} \approx \Delta x^2; \quad \Delta t_{\text{Advection}} \approx \Delta x$$

- **Diffusive time scale dominates as refinement level increases**

- **The need of overcoming small diffusive time scale:**
  - **Explicit super-time-stepping algorithm**
  - **Operator split semi-implicit (using HYPRE)**
  - **Fully implicit time stepping (Jacobian-Free Newton-Krylov) algorithms**

Fully Implicit JFNK Solver in FLASH

- **NSF Grant award (PHY-0903997) for fiscal years 2009 – 2011, $400K**
  - Dongwook Lee (PI), Shravan Gopal

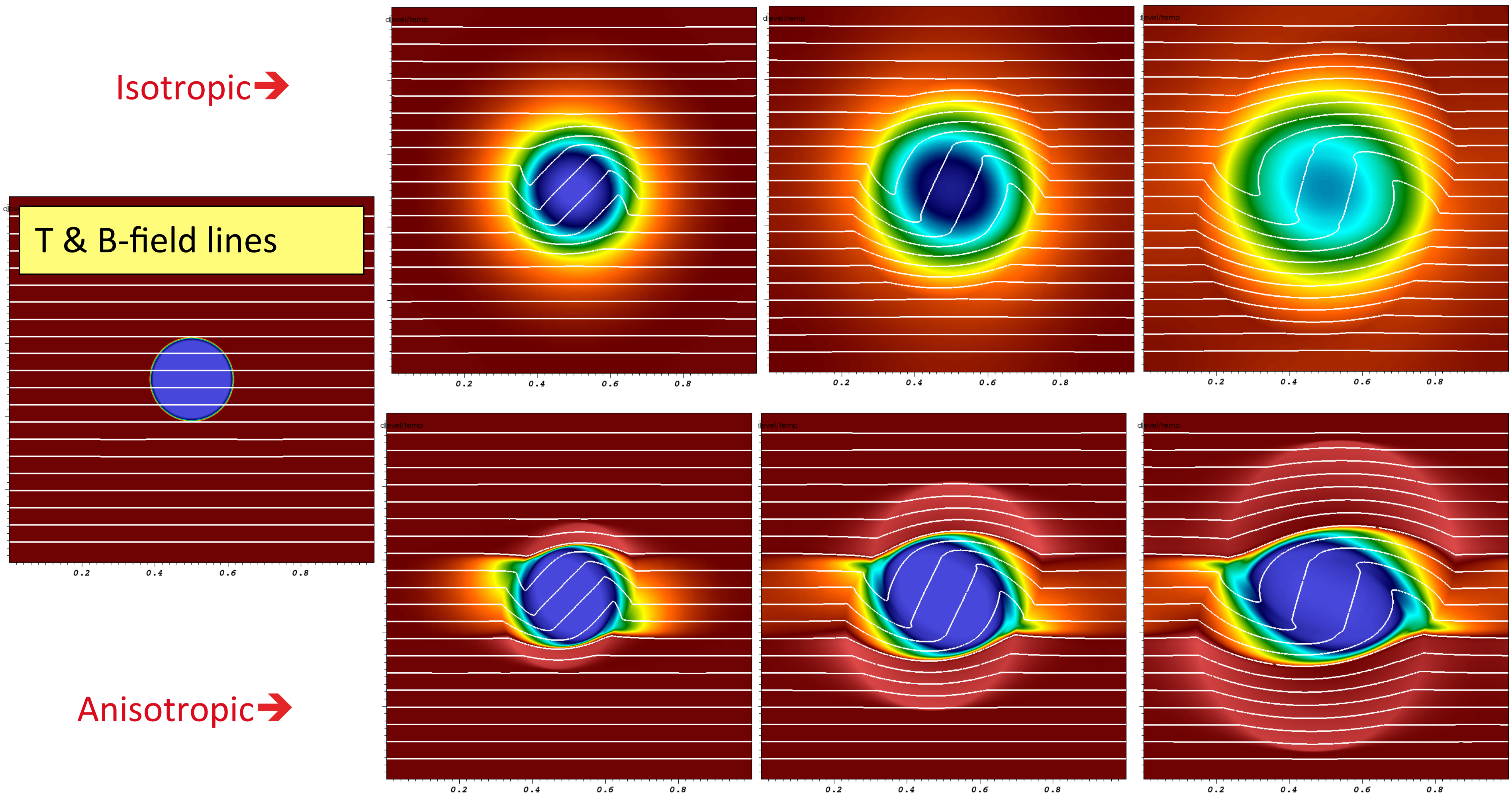
- **Jacobian-Free Newton-Krylov implicit scheme (e.g., Knoll and Keys 2004; Toth et al. 2006)**
  - 2nd order accurate in space and time to solve $\mathrm{A}x = b$
  - GMRES iteration to seek solutions in Krylov subspace
  - Hybrid method of using both Explicit/Implicit blocks in a computational domain
  - Requires load balancing between two different explicit/implicit types of blocks

- **Schwarz-type preconditioner**
  - Preconditioner to accelerate convergence rates for iterative solution
  - Off-processor data may be needed
  - Schwarz-type preconditioner minimizes the need for off-processor data
  - Efficient approach in FLASH's block-structured AMR

- **The implicit solver will extend FLASH's capability to overcome small diffusive time scales in both astrophysical and HEDP applications**

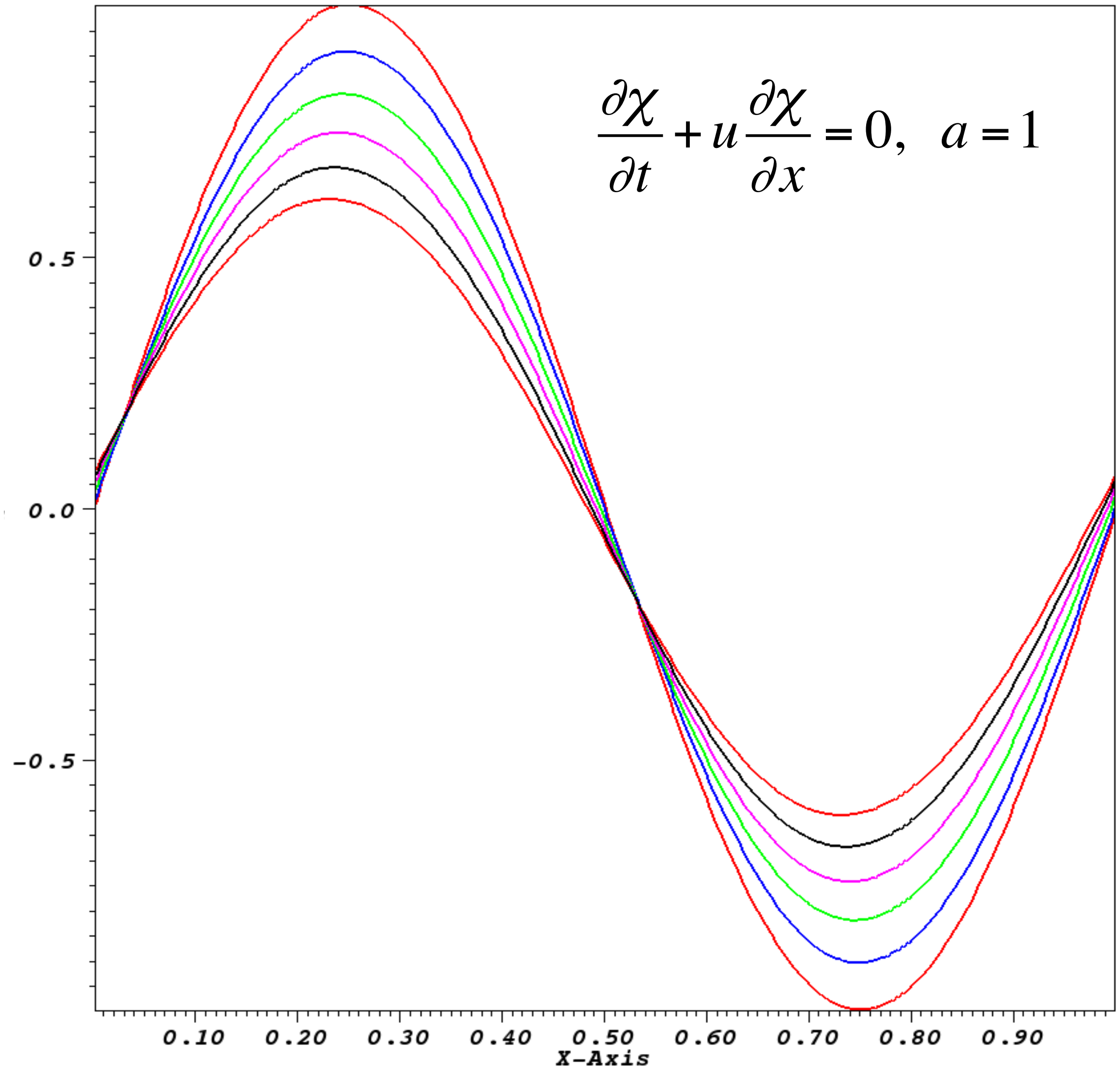
Anisotropic Heat Conduction Test for HEDP

- MHD Rotor test with $\chi_c = 5\times10^7$ for a cold plasma regime



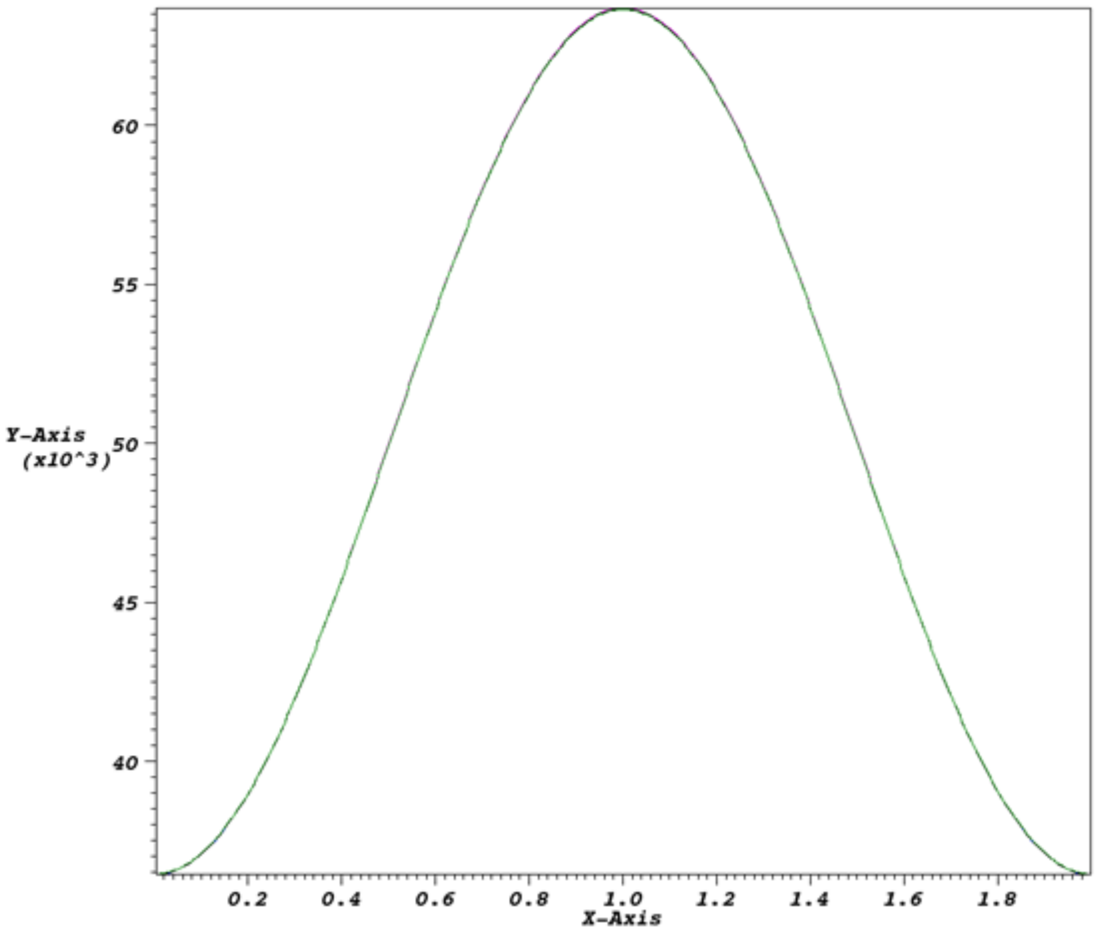
- For hot plasmas, a full Spitzer conductivity can be used

1D Sinusoidal Wave Advection



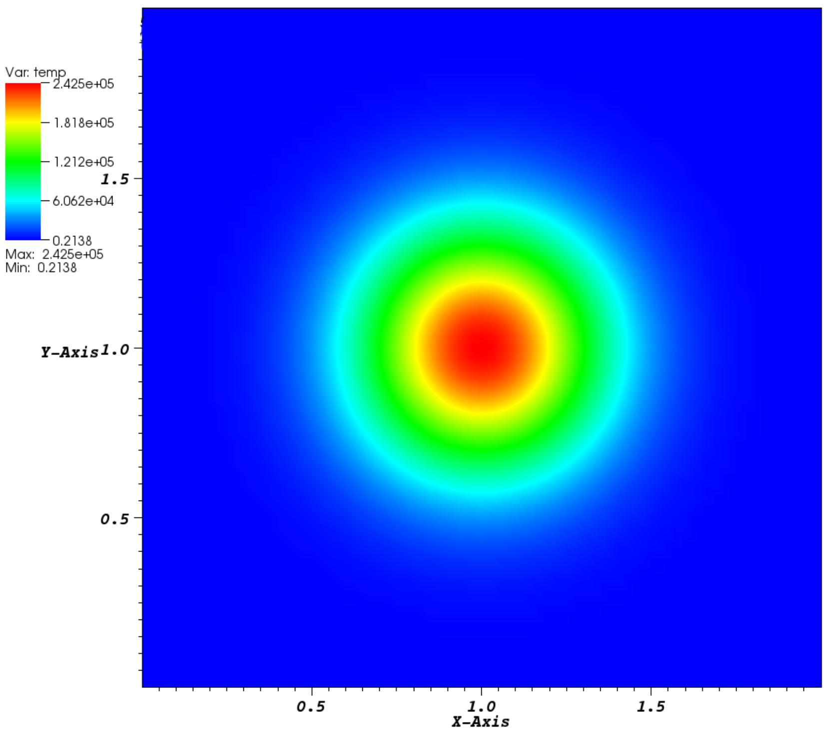
CFL=10 (2 iter with PC)	u=1
Red	t=0
Blue	t=2
Green	t=4
Magenta	t=6
Black	t=8
Red	t=10

1D Thermal Conduction

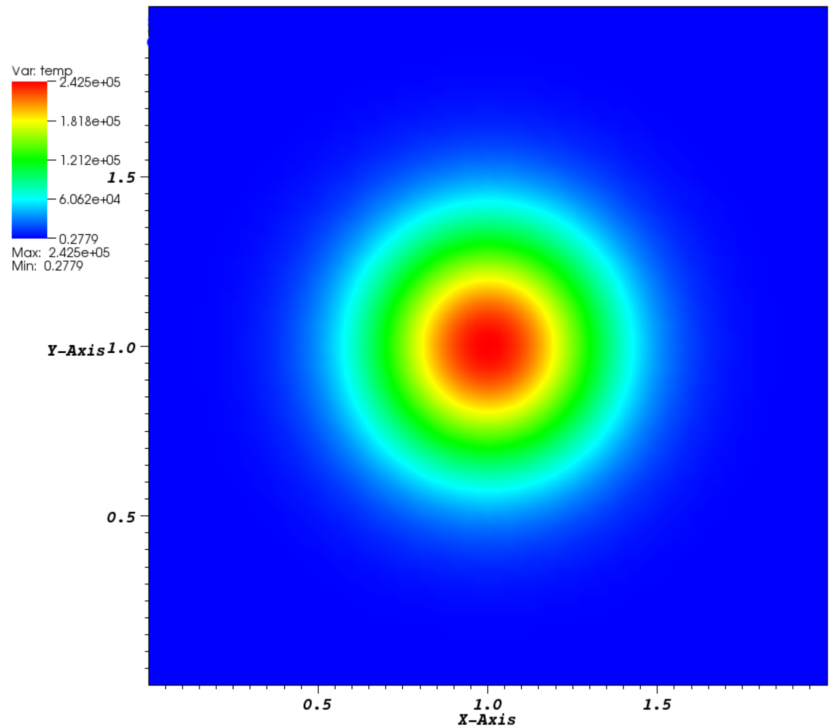


Scheme	CFL	steps
Explicit (Red)	0.8	19,256
Implicit (Blue)	0.8	19,256
Implicit (pink)	10	1556
Implicit (black)	100	174
Implicit (green)	1000	39

2D Thermal Conduction (4 blocks)



Implicit JFNK, GMRES with ILU(0)
L2 Error Norm: 1.95319e-3



HYPRE with PCG and ILU(0)
L2 Error Norm: 1.95318e-3

The Biermann Battery Effects

- Cosmic magnetic fields are ubiquitous, but their origins remain unclear
- Biermann bettery term is important in recreating cosmic conditions within the lab and in computations
- Magnetic fields are important because they modify transport process, accelerate particles and exert body forces
- Battery term can play an important role in seeding magnetic fields in HEDP simulations

$$-\frac{\partial B}{\partial t} = \nabla \times (U \times B) + c\,\frac{\nabla n_e \times \nabla p_e}{n_e^2 e}$$

- In FLASH's single-fluid radiation-hydro model with 3T, a simple battery approximation is available (Kulsrud 2004; Xu 2008) assuming:
  - Charge neutrality, LTE, a constant degree of ionization in space
  - For more complicated HEDP described by non-LTE, two-fluid model is required

Generalized Ohm's Law

- The (yet simplified) generalized Ohm's law can be written assuming high collisions in MHD single fluid theory and dropping the electron inertia term

$$E = \underbrace{-\frac{u \times B}{c}}_{\text{Induction Term}} + \underbrace{\frac{j}{\sigma}}_{\text{Ohmic Term}} + \underbrace{\frac{j \times B}{c n_e e}}_{\text{Hall Term}} \underbrace{- \frac{\nabla p_e}{n_e e}}_{\text{Battery Term}}$$

- Ideal MHD uses IT only (magnetic flux freezing, both ions and electrons are glued to the low-frequency fluid-like field lines)
- OT can be ignored for large magnetic Reynolds number
- Both Hall and Battery terms can be dropped if the Lamor gyration radius of ions is much smaller than the length scale (therefore no charge separation)
- HT becomes more important than BT for high plasma beta:

$$\frac{|\mathrm{BT}|}{|\mathrm{HT}|} = \frac{|\nabla p_e / e n_e|}{|j \times B / e n_e|} \approx \beta \frac{L_B}{L_p}$$

Unit System and Conversions

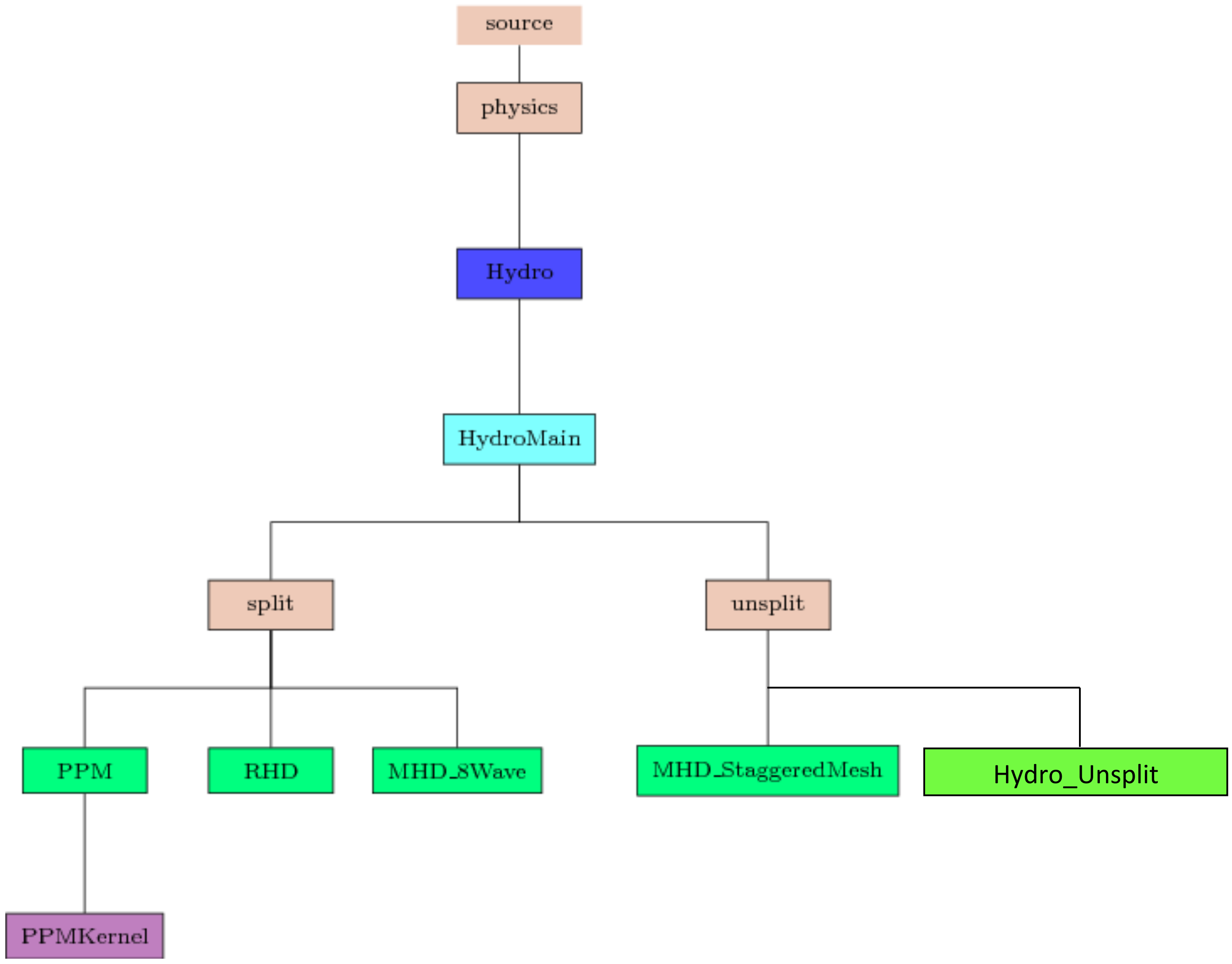
- The default unit in FLASH is cgs for physical variables such as density, pressure, etc.
- For electromagnetic variables, FLASH has a convenient unit in that magnetic permeability, electric permittivity, the speed of light and the factor 4pi are absorbed into the physical variables

Quantity	FLASH's None	Gaussian
magnetic field	$\mathbf{B}$	$\frac{\mathbf{B}}{\sqrt{4\pi}}$
electric field	$\mathbf{E}$	$\frac{c\mathbf{E}}{\sqrt{4\pi}}$
current density	$\mathbf{j}$	$\frac{\sqrt{4\pi}}{c}\mathbf{j}$
vector potential	$\mathbf{A}$	$\frac{\mathbf{A}}{\sqrt{4\pi}}$
magnetic diffusivity (aka magnetic viscosity, or resistivity)	η	$\frac{c^2}{4\pi}\eta$

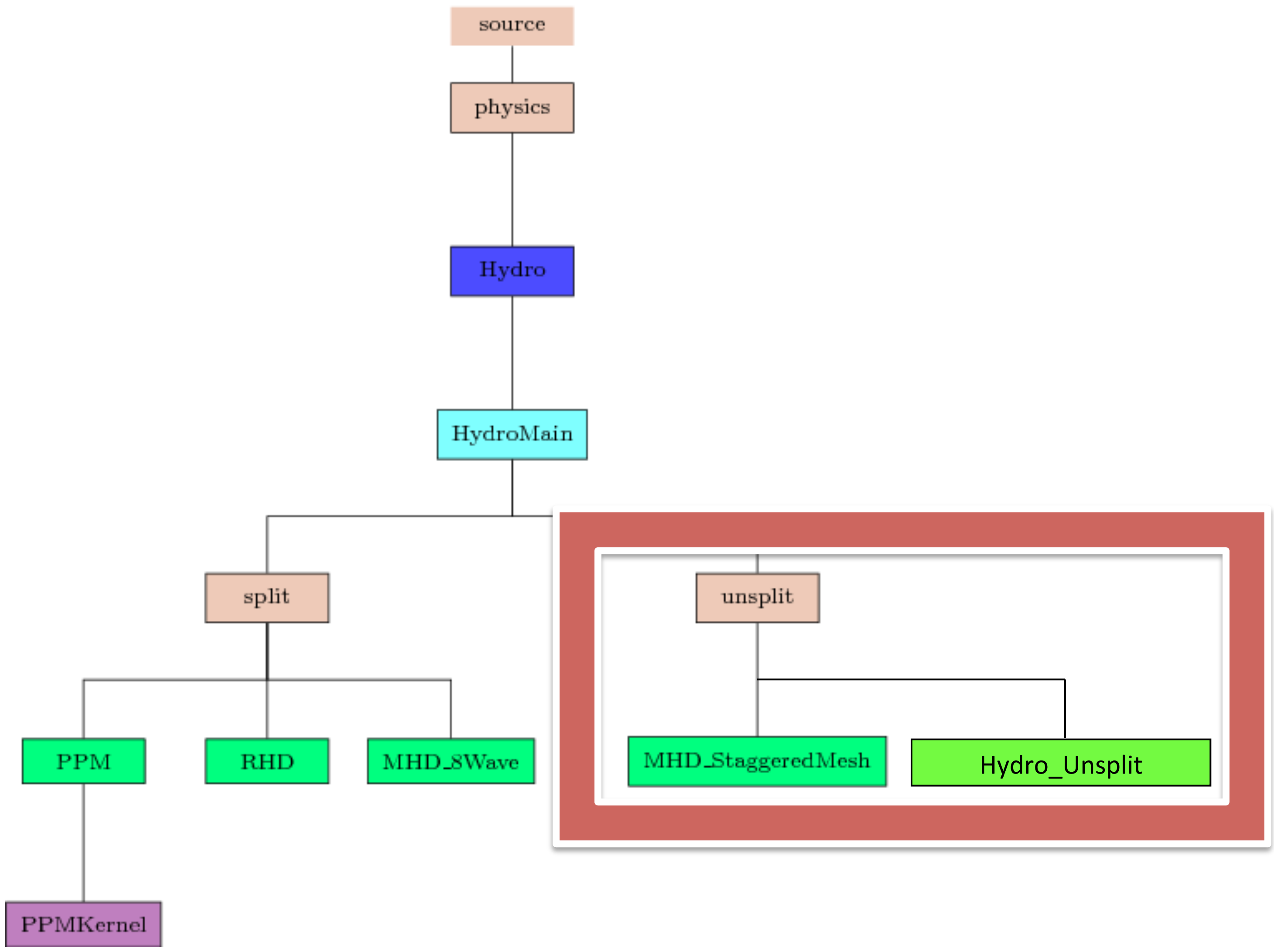
Table 4. Conversion table for electrodynamics quantities from FLASH's none to Gaussian.

- To simulate in gaussian cgs, FLASH needs to:
  - Initialize: $B_{none} \rightarrow B_{none}/\sqrt{4\pi}$
  - Visualize: $B_{chk} \rightarrow \sqrt{4\pi}B_{chk}$, where $B_{chk} = B_{none}/\sqrt{4\pi}$

The FLASH Code: Hydro Unit in FLASH



The FLASH Code: Hydro Unit in FLASH



The FLASH Code: Hydro Unit in FLASH

Journal of Computational Physics 228 (2009) 952–975

Contents lists available at ScienceDirect

Journal of Computational Physics

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

An unsplit staggered mesh scheme for multidimensional magnetohydrodynamics

Dongwook Lee [a], Anil E. Deane [b,*]

[a] ASC FLASH Center, University of Chicago, 5640 S. Ellis, Chicago,
[b] Institute for Physical Science and Technology, University of Ma

ARTICLE INFO

Article history:
Received 3 December 2007
Received in revised form 18 July 2008
Accepted 20 August 2008
Available online 12 September 2008

Keywords:
MHD
Magnetohydrodynamics
Constrained transport
Corner transport upwind
Unsplit scheme
Staggered mesh
High-order Godunov method

ABS

We intr
drodyna
nov flu
order M
"multidi
a single
port up
by the a
tion alg
The sch
tions us
directly
dissipati
ified ele
tion and
excellen
MHD tes
freely av

A Solution Accurate, Efficient and Stable Unsplit Staggered Mesh Scheme for Three Dimensional Magnetohydrodynamics

Dongwook Lee

The Flash Center for Computational Science, University of Chicago, 5747 S. Ellis, Chicago, IL 60637

Submitted to JCP, 2012

Abstract

In this paper, we extend the unsplit staggered mesh scheme (USM) for 2D magnetohydrodynamics (MHD) [D. Lee, A. Deane, An Unsplit Staggered Mesh Scheme for Multidimensional Magnetohydrodynamics, J. Comput. Phys. 228 (2009) 952–975] to a full 3D MHD scheme. The 3D scheme uses the same set of fundamental algorithmic ideas that have been developed in the 2D USM scheme. The scheme is a finite-volume Godunov method consisting of (1) a constrained transport (CT) method for preserving the solenoidal magnetic field evolution on a staggered grid, and (2) an efficient and accurate single-step, directionally unsplit multidimensional data reconstruction-evolution algorithm, which extends Colella's original 2D corner transport upwind (CTU) method [P. Colella, Multidimensional Upwind Methods for Hyperbolic Conservation Laws, J. Comput. Phys. 87 (1990) 446–466]. We present two types of data reconstruction-evolution algorithms for 3D: a reduced CTU scheme and a full CTU scheme. The reduced 3D CTU scheme is a variant of a direct 3D extension of Collela's 2D CTU method, whereas our full 3D CTU approach is a variant of the 3D unsplit CTU method by Saltzman [J. Saltzman, An unsplit 3D upwind method for hyperbolic conservation laws, J. Comput. Phys. 115 (1994) 153–168] for hyperbolic conservation laws. The key novelty in our

Unsplit Staggered Mesh (USM) MHD Solver

- Shock-capturing high-order Godunov Riemann solver (Lee & Deane, JCP, 2009; Lee 2012, submitted)
- Finite volume method, adaptive mesh refinement, uniform grid
- New data reconstruction-evolution algorithm for high-order accuracy
- 1st order Godunov, 2nd order MUSCL-Hancock, 3rd order PPM, 5th Order WENO
- Approximate Riemann solvers: Roe, HLL, HLLC, HLLD, Marquina, modified Marquina, Local Lax-Friedrichs
- Monotonicity preserving upwind PPM slope limiter for MHD (Lee, 2010, Astronum)
- Divergence of magnetic fields is numerically controlled on a staggered grid, using a constrained transport (CT) method (Evans & Hawley, 1998)
- Wide ranges of plasma flows, extended to HEDP
- Full Courant stability limit (CFL ~ 1 for 3D) using corner-transport-upwind (CTU)

MHD Governing Equations

- MHD system of equations:

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

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u}\mathbf{u} - \mathbf{B}\mathbf{B}) + \nabla p_{tot} = 0,$$

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla \cdot (\mathbf{u}\mathbf{B} - \mathbf{B}\mathbf{u}) = 0,$$

$$\frac{\partial E}{\partial t} + \nabla \cdot (\mathbf{u}E + \mathbf{u}p_{tot} - \mathbf{B}\mathbf{B} \cdot \mathbf{u}) = 0.$$

- This can be written in a simple matrix form:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} + \frac{\partial \mathbf{H}}{\partial z} = 0,$$

MHD Governing Equations

- Conservative variables and fluxes:

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ B_x \\ B_y \\ B_z \\ E \end{pmatrix}, \quad \mathbf{F} = \begin{pmatrix} \rho u \\ \rho u^2 + p_{tot} - B_x^2 \\ \rho u v - B_y B_x \\ \rho u w - B_z B_x \\ 0 \\ u B_y - v B_x (= -E_z) \\ u B_z - w B_x (= E_y) \\ (E + p_{tot}) u - B_x (u B_x + v B_y + w B_z) \end{pmatrix}, \tag{7}$$

$$\mathbf{G} = \begin{pmatrix} \rho v \\ \rho v u - B_x B_y \\ \rho v^2 + p_{tot} - B_y^2 \\ \rho v w - B_z B_y \\ v B_x - u B_y (= E_z) \\ 0 \\ v B_z - w B_y (= -E_x) \\ (E + p_{tot}) v - B_y (u B_x + v B_y + w B_z) \end{pmatrix}, \quad \mathbf{H} = \begin{pmatrix} \rho w \\ \rho w u - B_x B_z \\ \rho w v - B_y B_z \\ \rho w^2 + p_{tot} - B_z^2 \\ w B_x - u B_z (= -E_y) \\ w B_y - v B_z (= E_x) \\ 0 \\ (E + p_{tot}) w - B_z (u B_x + v B_y + w B_z) \end{pmatrix}. \tag{8}$$

General Features

- New approach of using characteristic tracing for BOTH normal predictor and transverse corrector

- Reduced 3D CTU
  - A direct extension of 2D CTU to 3D
  - Requires 3 Riemann solves for 3D (6-ctu needs 6 Riemann solves)
  - Only including second cross derivatives
  - CFL limit ~ 0.5

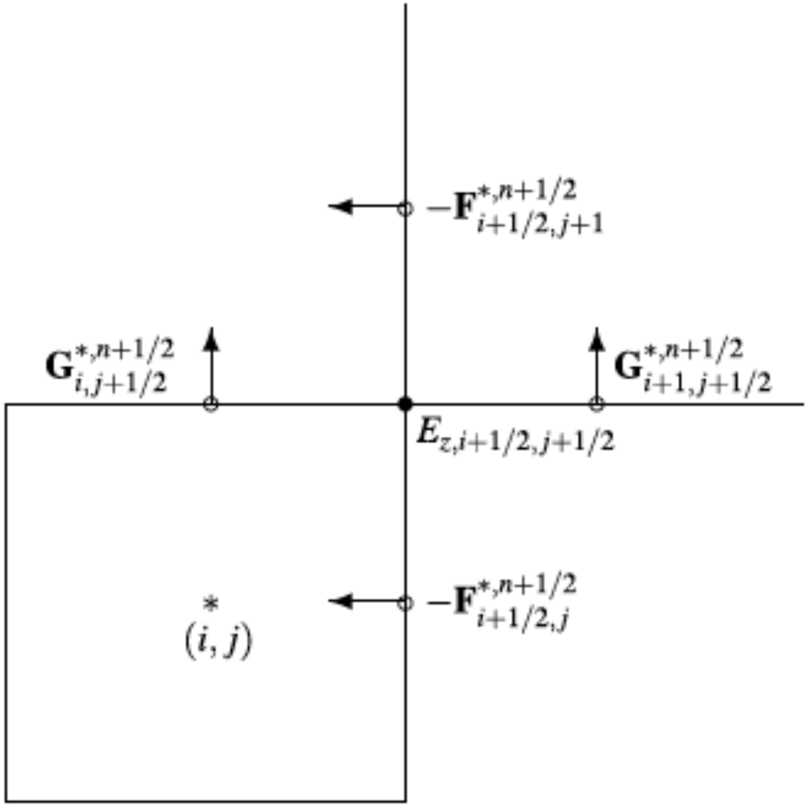
- Full 3D CTU
  - Full considerations of accounting for third cross derivatives
  - Requires 3 Riemann solves for 3D (12-ctu needs 12 Riemann solves)
  - CFL limit ~ 1.0
  - 20% relative performance gain compared to reduced 3D CTU

Divergence-Free fields:
Constrained Transport (CT) MHD

Constraint Transport Method

- CT scheme by Balsara and Spicer, 1998:

$$E^{n+1/2}_{z,i+1/2,j+1/2,k} = \frac{1}{4}\left(E^{*,n+1/2}_{z,i+1/2,j,k} + E^{*,n+1/2}_{z,i+1/2,j+1,k} + E^{*,n+1/2}_{z,i,j+1/2,k} + E^{*,n+1/2}_{z,i+1,j+1/2,k}\right)$$



Constraint Transport Method: Recall…

- Conservative variables and fluxes:

$$
\mathbf{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ B_x \\ B_y \\ B_z \\ E \end{pmatrix}, \quad
\mathbf{F} = \begin{pmatrix} \rho u \\ \rho u^2 + p_{tot} - B_x^2 \\ \rho u v - B_y B_x \\ \rho u w - B_z B_x \\ 0 \\ u B_y - v B_x (= -E_z) \\ u B_z - w B_x (= E_y) \\ (E + p_{tot}) u - B_x (u B_x + v B_y + w B_z) \end{pmatrix}, \qquad (7)
$$

$$
\mathbf{G} = \begin{pmatrix} \rho v \\ \rho v u - B_x B_y \\ \rho v^2 + p_{tot} - B_y^2 \\ v B_x - u B_y (= E_z) \\ 0 \\ v B_z - w B_y (= -E_x) \\ (E + p_{tot}) v - B_y (u B_x + v B_y + w B_z) \end{pmatrix}, \quad
\mathbf{H} = \begin{pmatrix} \rho w \\ \rho w u - B_x B_z \\ \rho w v - B_y B_z \\ \rho w^2 + p_{tot} - B_z^2 \\ w B_x - u B_z (= -E_y) \\ w B_y - v B_z (= E_x) \\ 0 \\ (E + p_{tot}) w - B_z (u B_x + v B_y + w B_z) \end{pmatrix}. \qquad (8)
$$

A New Upwind Constraint Transport Method

- New upwind biased modified electric field construction(upwind-MEC), Lee 2012:

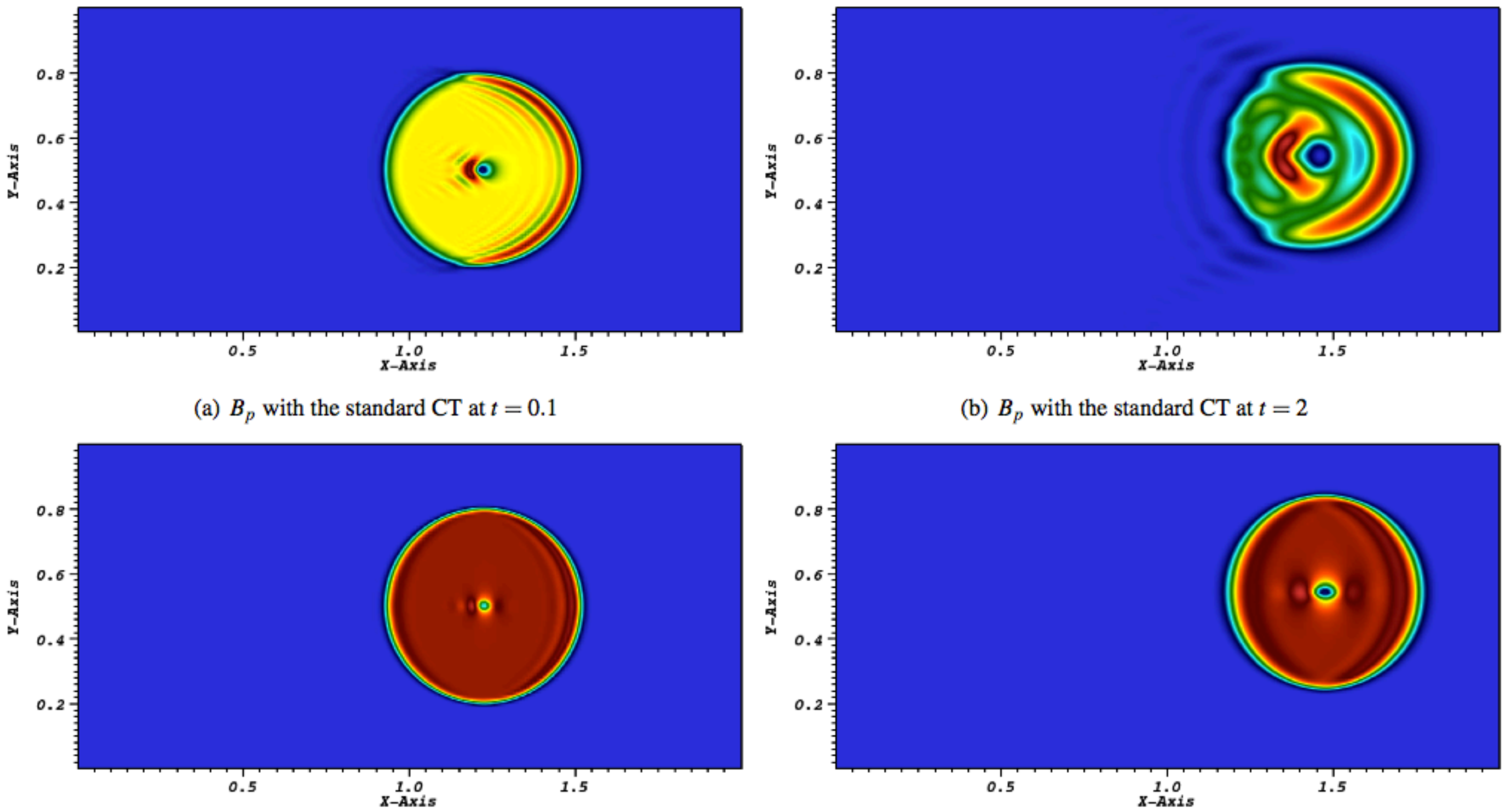
$$
\begin{aligned}
E^{n+1/2}_{z,i+1/2,j+1/2,k} = \alpha \Bigg[& v_P \left(E^{*,n+1/2}_{z,i+1/2,j,k} + \frac{\Delta y}{2} \frac{\partial E^{*,n+1/2}_{z,i+1/2,j,k}}{\partial y} + \frac{\Delta y^2}{8} \frac{\partial^2 E^{*,n+1/2}_{z,i+1/2,j,k}}{\partial y^2} \right) + \\
& v_N \left(E^{*,n+1/2}_{z,i+1/2,j+1,k} - \frac{\Delta y}{2} \frac{\partial E^{*,n+1/2}_{z,i+1/2,j+1,k}}{\partial y} + \frac{\Delta y^2}{8} \frac{\partial^2 E^{*,n+1/2}_{z,i+1/2,j+1,k}}{\partial y^2} \right) + \\
& u_P \left(E^{*,n+1/2}_{z,i,j+1/2,k} + \frac{\Delta x}{2} \frac{\partial E^{*,n+1/2}_{z,i,j+1/2,k}}{\partial x} + \frac{\Delta x^2}{8} \frac{\partial^2 E^{*,n+1/2}_{z,i,j+1/2,k}}{\partial x^2} \right) + \\
& u_N \left(E^{*,n+1/2}_{z,i+1,j+1/2,k} - \frac{\Delta x}{2} \frac{\partial E^{*,n+1/2}_{z,i+1,j+1/2,k}}{\partial x} + \frac{\Delta x^2}{8} \frac{\partial^2 E^{*,n+1/2}_{z,i+1,j+1/2,k}}{\partial x^2} \right) \Bigg].
\end{aligned}
$$

$$
\begin{aligned}
u_P &= \frac{1}{2}(1+\mathrm{sign}(u_{i+1/2,j+1/2}))|\mathrm{sign}(u_{i+1/2,j+1/2})|, \\
u_N &= \frac{1}{2}(1-\mathrm{sign}(u_{i+1/2,j+1/2}))|\mathrm{sign}(u_{i+1/2,j+1/2})|, \\
v_P &= \frac{1}{2}(1+\mathrm{sign}(v_{i+1/2,j+1/2}))|\mathrm{sign}(v_{i+1/2,j+1/2})|, \\
v_N &= \frac{1}{2}(1-\mathrm{sign}(v_{i+1/2,j+1/2}))|\mathrm{sign}(v_{i+1/2,j+1/2})|,
\end{aligned}
$$

$$
\mathrm{sign}(x) = \begin{cases} 1 & \text{if } x > 0, \\ 0 & \text{if } x = 0, \\ -1 & \text{if } x < 0. \end{cases}
$$

A New Upwind Constraint Transport Method

- Small angle advection of the 2D field loop:



(a) B_p with the standard CT at $t = 0.1$

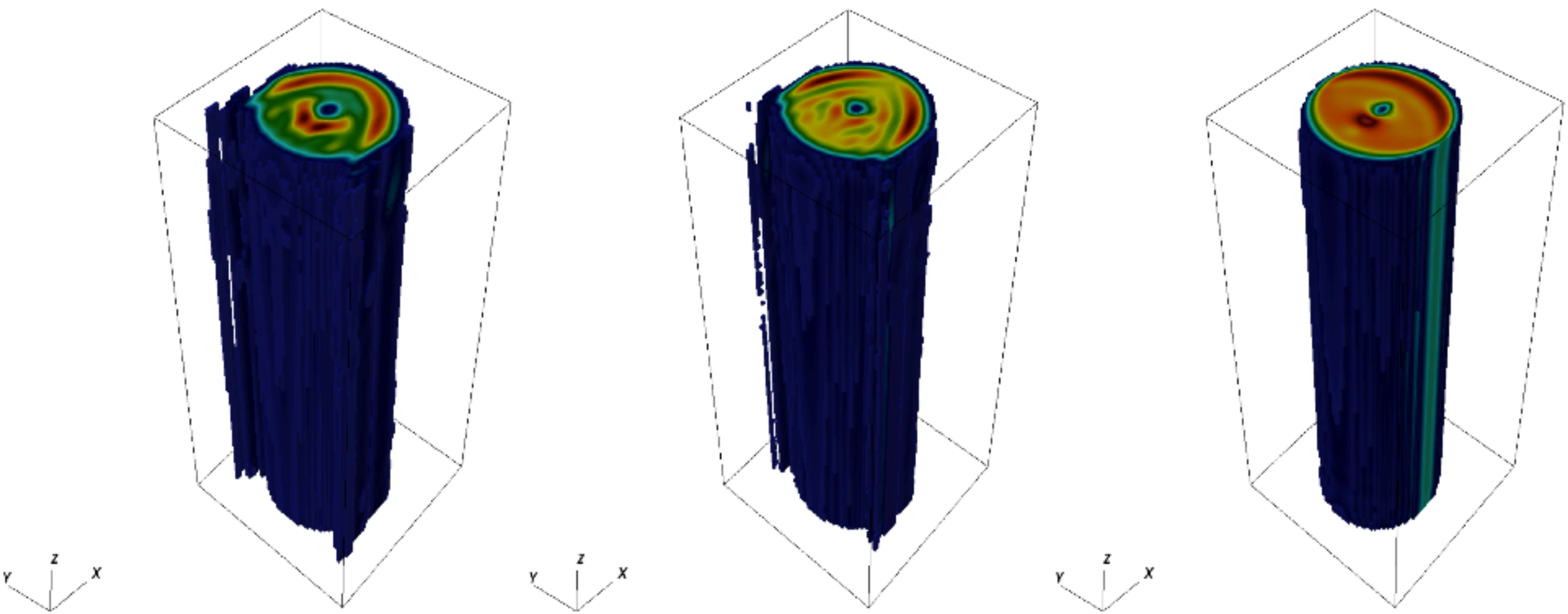
(b) B_p with the standard CT at $t = 2$

(c) B_p with the upwind-MEC at $t = 0.1$

(d) B_p with the upwind-MEC at $t = 2$

A New Upwind Constraint Transport Method

- Small angle advection of the 3D field loop:

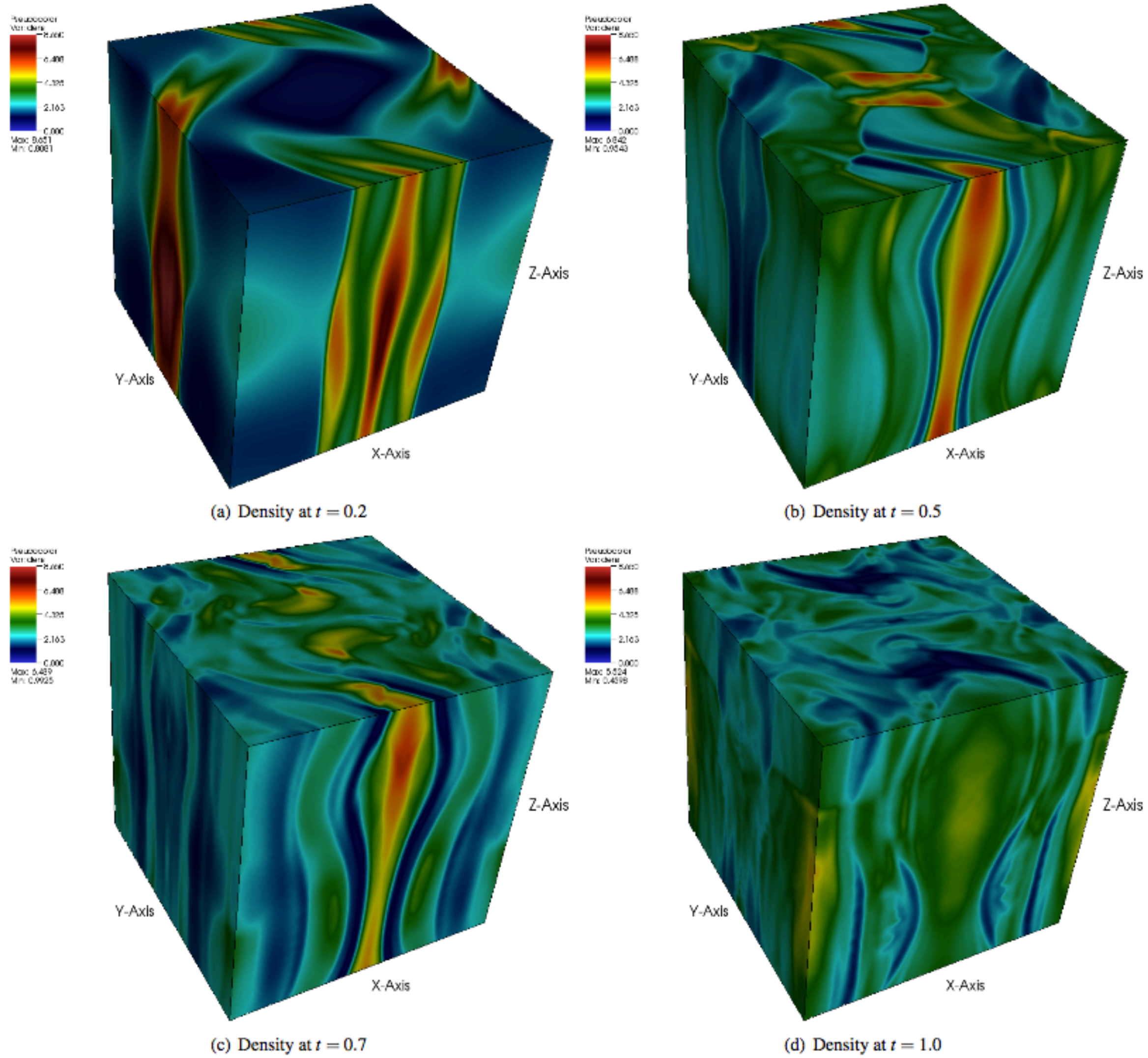


(a) B_p using the standard CT at $t = 2$. (b) B_p using the standard MEC at $t = 2$. (c) B_p using the upwind MEC at $t = 2$.

Intermediate Summary I

- Three CT schemes discussed:
  - **Standard CT scheme by Balsara and Spicer, 1998:**
    - Takes a simple arithmetic averaging
    - Lacks numerical diffusion for magnetic fields advection
  - **Modified electric field construction (MEC) scheme by Lee and Deane, 2009:**
    - 3rd order accurate in space
    - Not enough numerical diffusion for field advection
  - **Upwind biased MEC (upwind-MEC) scheme by Lee, 2012 (submitted)**
    - Upwind scheme of MEC
    - Added numerical diffusion to stabilize field advection

Numerical Tests

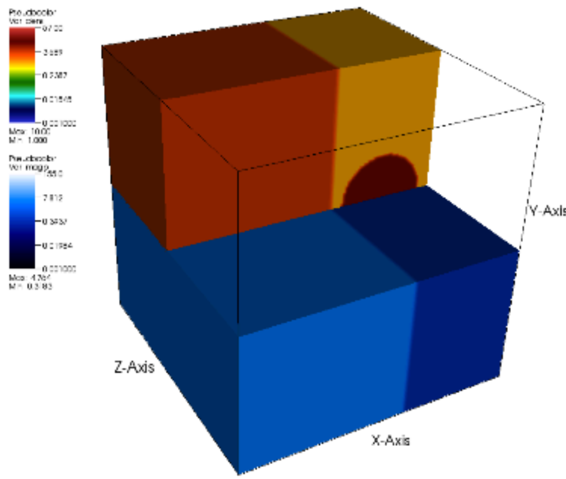


(a) Density at $t = 0.2$ (b) Density at $t = 0.5$

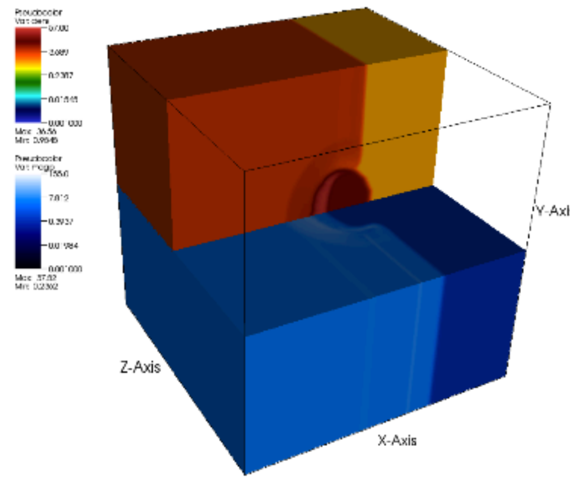
(c) Density at $t = 0.7$ (d) Density at $t = 1.0$

Fig. 11. Density plots of the Orszag-Tang problem at a resolution of 128^3 using the Roe Riemann solver.

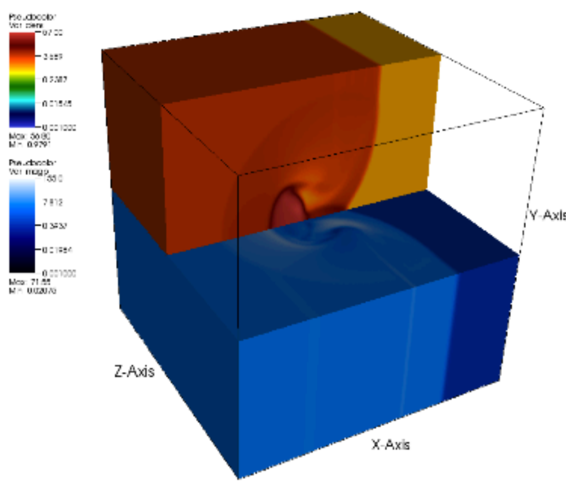
Numerical Tests



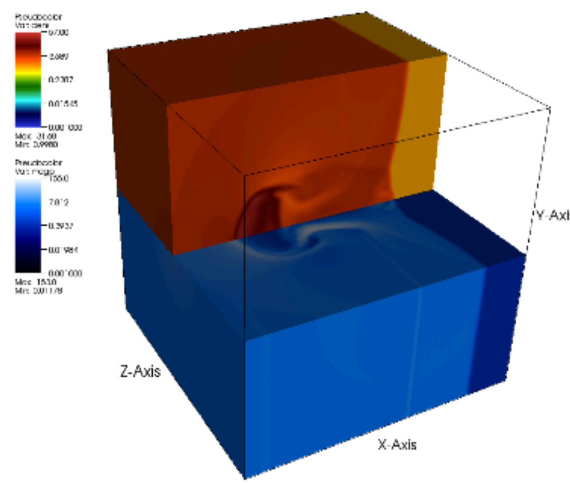
(a) Density and magnetic pressure at $t = 0.0$



(b) Density and magnetic pressure at $t = 0.02$

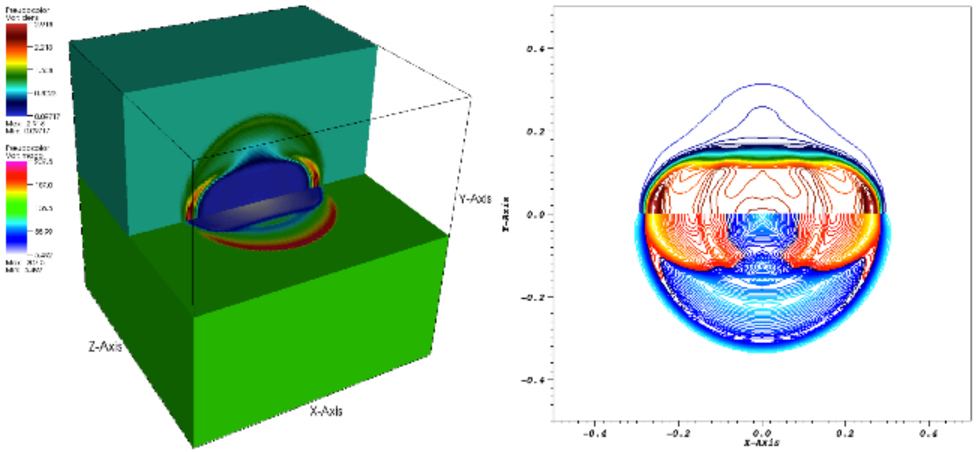


(c) Density and magnetic pressure at $t = 0.04$



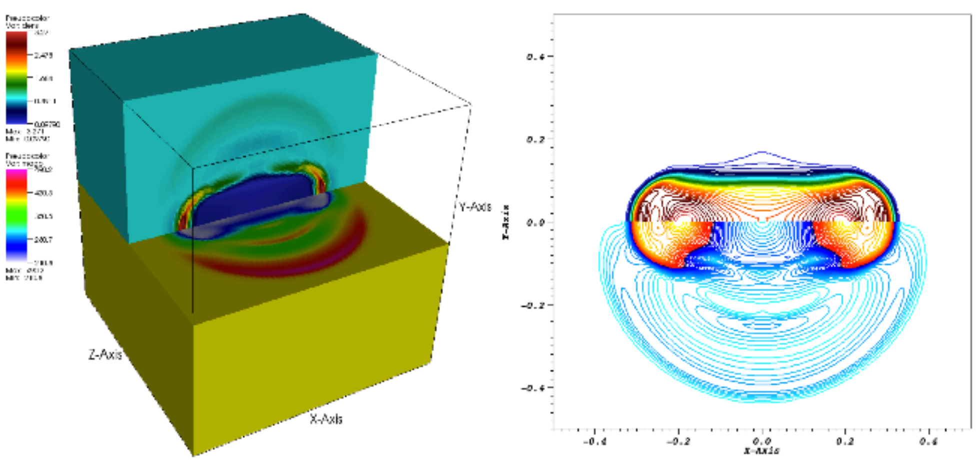
(d) Density and magnetic pressure at $t = 0.06$

Numerical Tests



(a) Density and magnetic pressure at $t = 0.01$ (b) Contours of gas pressure and total velocity at $t = 0.01$

Fig. 17. Results of the blast problem simulation with $B_x = 50/\sqrt{4\pi}$ using a hybrid Riemann solver. In (a), density (denoted as "dens" in the legend) is plotted at the top half. Magnetic pressure (denoted as "magp" in the legend) is plotted at the bottom half. In (b), 40 contour lines are plotted.



(a) Density and magnetic pressure at $t = 0.01$ (b) Contours of gas pressure and total velocity at $t = 0.01$

Runtime Parameters

```
# Runtime Parameters
# (1) Interpolation, reconstruction, slope limiter:
PARAMETER order                INTEGER   2             # Order of scheme: 1st/2nd/3rd/5th order
PARAMETER transOrder           INTEGER   1             # Order of transverse flux: 1st order. 3rd order is experimental.
PARAMETER slopeLimiter         STRING   "vanLeer"      # Slope limiter for Riemann state
PARAMETER charLimiting         BOOLEAN   TRUE          # Turn on/off characteristic/primitive limiting
PARAMETER LimitedSlopeBeta     REAL      1.0           # Any real value specific for the Limited Slope
                                                       # limiter (e.g., 1.0 for minmod, 2.0 for superbee)
PARAMETER use_steepening       BOOLEAN   FALSE         # Turn on/off PPM contact steepening
PARAMETER use_flattening       BOOLEAN   FALSE         # Turn on/off flattening
PARAMETER use_avisc            BOOLEAN   FALSE         # Turn on/off artificial viscosity
PARAMETER cvisc                REAL      0.1           # artificial viscosity constant
PARAMETER use_upwindTVD        BOOLEAN   FALSE         # Turn on/off upwinding TVD slopes

# (2) For 3D CTU
PARAMETER use_3dFullCTU        BOOLEAN   TRUE          # FALSE will give the simpler CTU without corner upwind coupling
                                                       # and will only provide CFL < 1/2
# (3) Riemann solvers
PARAMETER RiemannSolver        STRING   "Roe"          # Approximate Riemann solver:
                                                       # Roe (default), HLL, HLLC, Marquina, MarquinaMod, Hybrid
                                                       # or local Lax-Friedrichs, plus HLLD for MHD
PARAMETER entropy              BOOLEAN   FALSE         # Turn on/off an entropy fix routine
PARAMETER entropyFixMethod     STRING   "HARTENHYMAN"  # Entropy fix method for the Roe Riemann solver:
                                                       # Harten or HartenHyman
PARAMETER shockDetect          BOOLEAN   FALSE         # Turn on/off a shock detecting switch
PARAMETER EOSforRiemann        BOOLEAN   FALSE         # Turn on/off EOS calls for the Riemann states
PARAMETER addThermalFlux       BOOLEAN   TRUE          # Add/don't add thermal fluxes to hydro fluxes

# (4) Gravity updates
PARAMETER use_gravHalfUpdate   BOOLEAN   FALSE         # Include gravitational accelerations to hydro coupling at n+1/2
PARAMETER use_gravConsv        BOOLEAN   FALSE         # Use conservative variables for gravity coupling at n+1/2
PARAMETER use_GravPotUpdate    BOOLEAN   FALSE         # Parameter for half timestep update of gravitational potential
```

Runtime Parameters

```
# Runtime Parameters for unsplit USM-MHD solver
PARAMETER killdivb              BOOLEAN   TRUE               # Turn on/off  DivB cleaning
PARAMETER E_modification        BOOLEAN   TRUE               # Turn on/off electric field modification
PARAMETER E_upwind              BOOLEAN   FALSE              # Turn on/off upwind update for induction equations
PARAMETER energyFix             BOOLEAN   FALSE              # Turn on/off an energy correction for CT scheme
PARAMETER ForceHydroLimit       BOOLEAN   FALSE              # Turn on/off a hydro limiting switch
PARAMETER facevar2ndOrder       BOOLEAN   TRUE               # Turn on/off a 2nd order facevar update
PARAMETER use_Biermann          BOOLEAN    FALSE             # Biermann Battery Term
PARAMETER use_Biermann1T        BOOLEAN    FALSE             # 1T Biermann Battery Term
PARAMETER hy_biermannSource     BOOLEAN    FALSE             # enable Battery Source (vs. flux)
PARAMETER hy_bier1TZ            REAL      -1.0               # Zbar value for 1T Biermann Battery Term
PARAMETER hy_bier1TA            REAL      -1.0               # Abar value for 1T Biermann Battery Term
PARAMETER prolMethod            STRING   "INJECTION_PROL"    # Prolongation method: injection_prol/Balsara_prol
PARAMETER hy_biermannCoef       REAL      1.0                # Coefficient of Biermann Battery flux
```

```
# Number of guard cells at each boundary
USESETUPVARS SupportWeno, SupportPpmUpwind
IF SupportWeno
   GUARDCELLS 6 # the Unsplit Hydro/MHD solver requires 6 guard cells to support WENO!
ELSEIF SupportPpmUpwind
   GUARDCELLS 6 # the Unsplit Hydro/MHD solver requires 6 guard cells to support PPM Upwind!
ELSE
   GUARDCELLS 4 # the Unsplit Hydro/MHD solver requires 4 guard cell layers!
ENDIF
```

Intermediate Summary II

- Verification tests for the reduced/full 3D CTU schemes:
  - CFL=0.95 for all 3D simulations using the full CTU scheme
  - CFL=0.475 for the reduced CTU scheme
  - They both converge in 2nd order
  - 20% performance gain in using the full CTU scheme:

$$\frac{CPU_{F-ctu}}{CPU_{R-ctu}} \approx 0.8$$

  - Various choices in runtime parameters

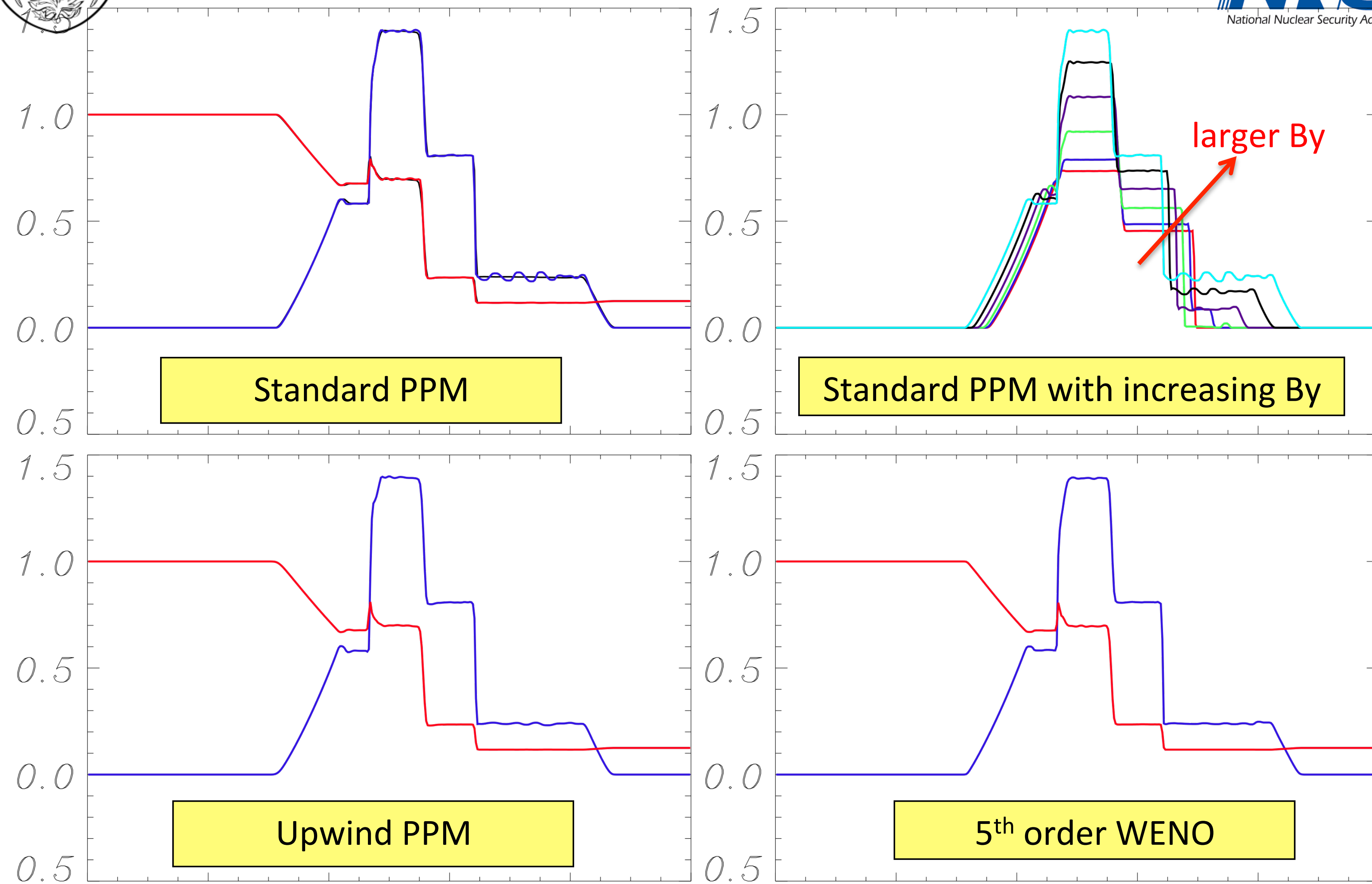
Outline & Conclusion

- **Broad Ranges of Applications**
  - Astrophysics and HEDP
- **Algorithms and mathematics**
  - Governing equations, difference equations of PDEs
  - Dimensionally Split vs. Unsplit formulations
  - Hydro and MHD solvers
    - High-order Reconstructions, Riemann problems, physical units
  - Explicit vs. Implicit schemes
  - Beyond the hyperbolic system (diffusion, source terms, Biermann battery, etc)
- **Examples**: setting up problems, how to use various features and switches
- **Summary**

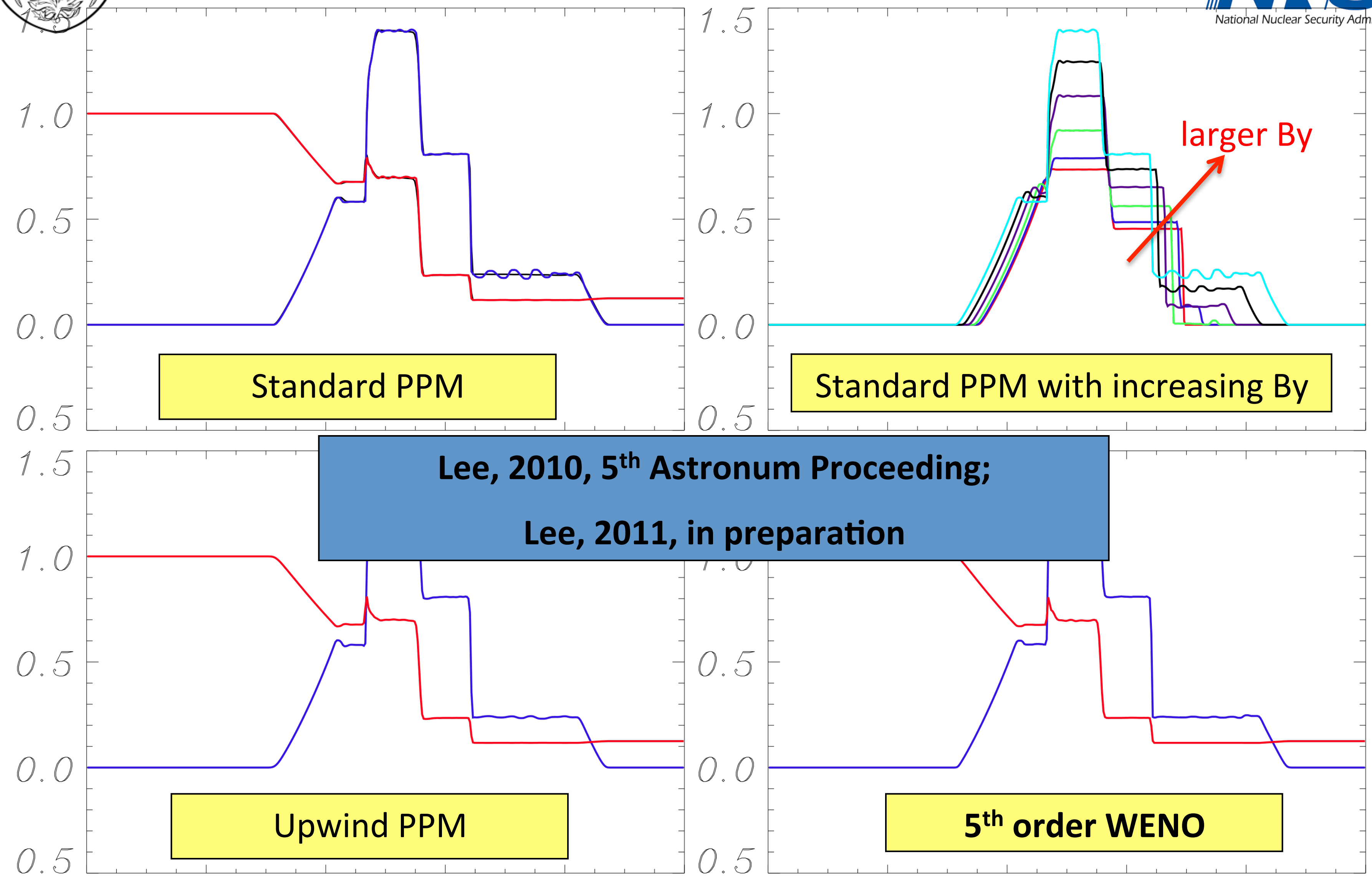
Thank You

Questions?

New Upwind PPM for Slowly Moving Shock

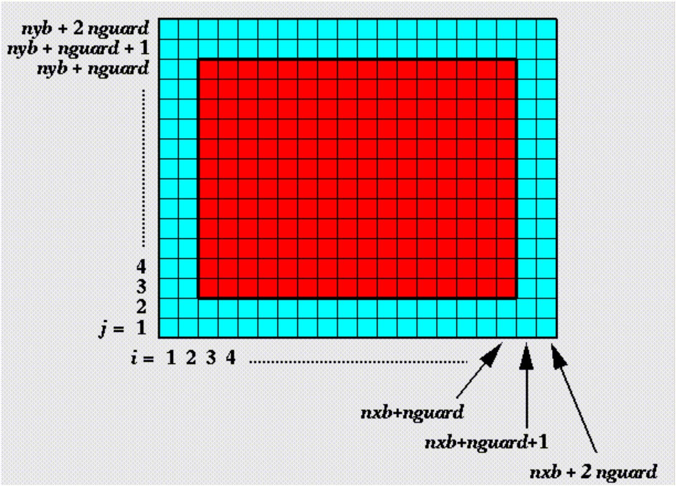


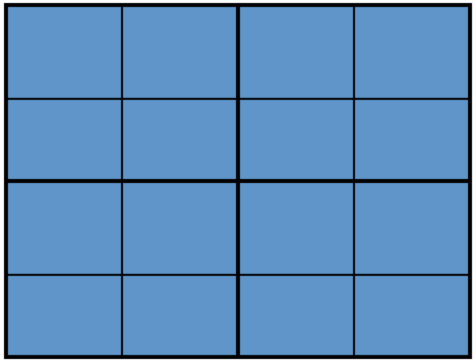
New Upwind PPM for Slowly Moving Shock



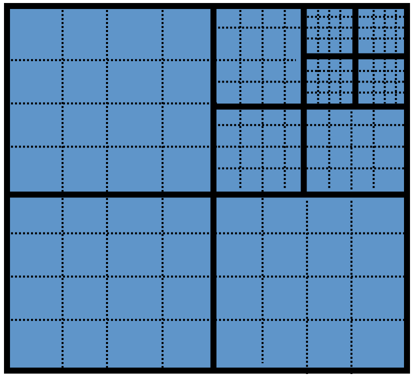
Block and Mesh Packages

- Mesh package can be selected at configuration time
- The basic abstraction is a block of interior cells surrounded by guard cells
- Grid unit makes sure that blocks are self contained before being given to the solvers

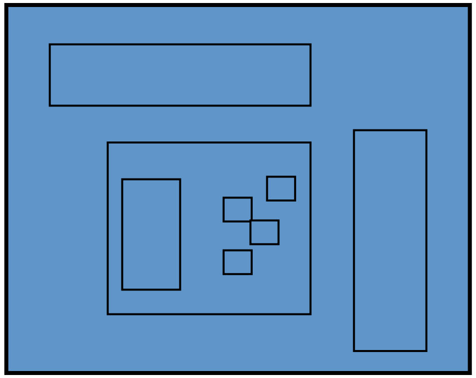




Uniform Grid



Oct tree based AMR - PARAMESH



AMR with variable patch size - CHOMBO

Unsplit Formulations

Take a deep breath!

Linearized System

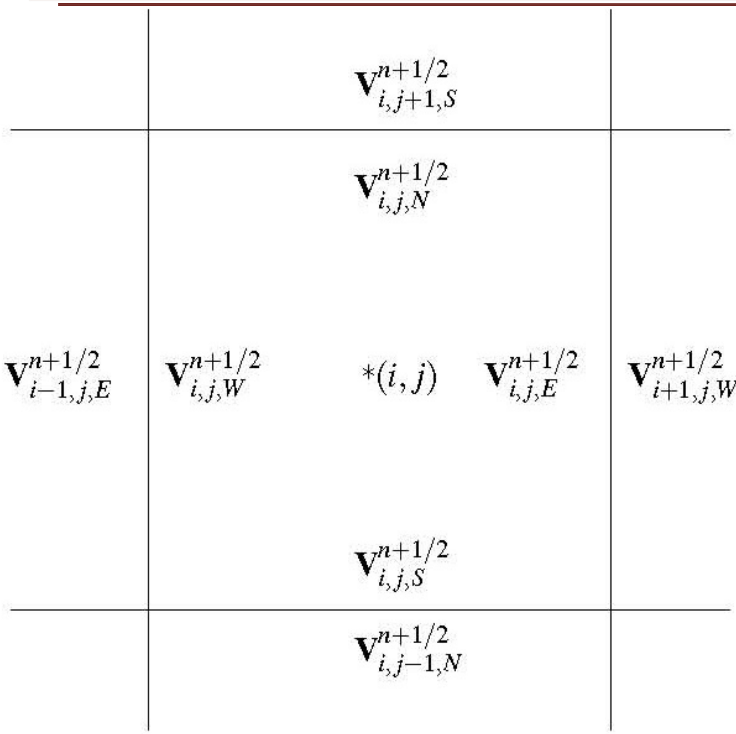
- A primitive form:

$$\mathbf{V} = (\rho, u, v, w, B_x, B_y, B_z, p)^T, \quad \frac{\partial \mathbf{V}}{\partial t} + \mathbf{A}_x \frac{\partial \mathbf{V}}{\partial x} + \mathbf{A}_y \frac{\partial \mathbf{V}}{\partial y} + \mathbf{A}_z \frac{\partial \mathbf{V}}{\partial z} = 0.$$

where the coefficient matrix is

$$\mathbf{A}_x = \begin{pmatrix} u & \rho & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & u & 0 & 0 & -\frac{B_x}{\rho} & \frac{B_y}{\rho} & \frac{B_z}{\rho} & \frac{1}{\rho} \\ 0 & 0 & u & 0 & -\frac{B_y}{\rho} & -\frac{B_x}{\rho} & 0 & 0 \\ 0 & 0 & 0 & u & -\frac{B_z}{\rho} & 0 & -\frac{B_x}{\rho} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & B_y & -B_x & 0 & -v & u & 0 & 0 \\ 0 & B_z & 0 & -B_x & -w & 0 & u & 0 \\ 0 & \gamma p & 0 & 0 & -k\mathbf{u} \cdot \mathbf{B} & 0 & 0 & u \end{pmatrix},$$

Corner Transport Upwind (CTU)



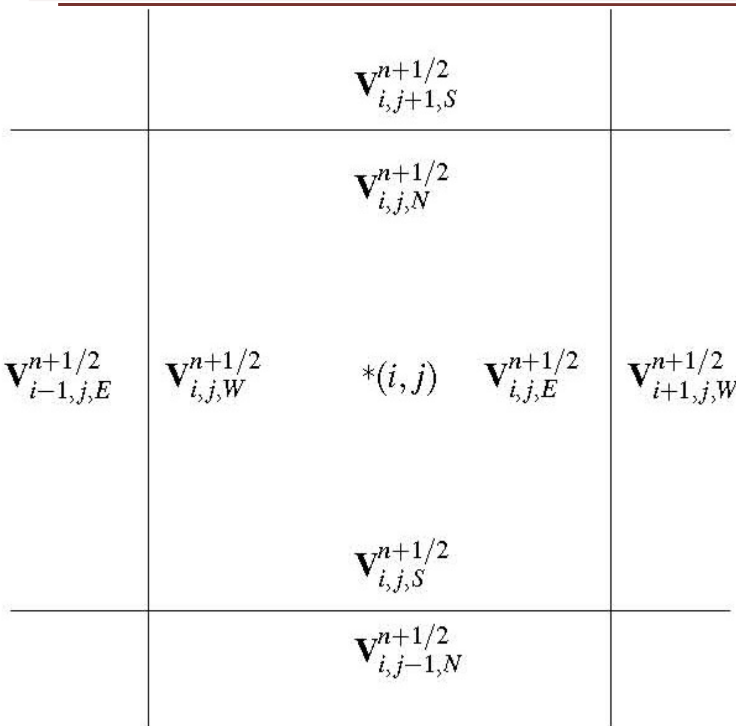
Linear system in 3D

$$\mathbf{V}^{n+1/2}_{i,j,k,E,W} = \mathbf{V}^n_{i,j,k} + \frac{1}{2}[\pm\mathbf{I} - \frac{\Delta t}{\Delta x}\mathbf{A}_x(\mathbf{V}^n_{i,j,k})]\Delta^n_i - \frac{\Delta t}{2\Delta y}\mathbf{A}_y(\mathbf{V}^n_{i,j,k})\Delta^n_j - \frac{\Delta t}{2\Delta z}\mathbf{A}_z(\mathbf{V}^n_{i,j,k})\Delta^n_k,$$

$$\mathbf{V}^{n+1/2}_{i,j,k,N,S} = \mathbf{V}^n_{i,j,k} - \frac{\Delta t}{2\Delta x}\mathbf{A}_x(\mathbf{V}^n_{i,j,k})\Delta^n_i + \frac{1}{2}[\pm\mathbf{I} - \frac{\Delta t}{\Delta y}\mathbf{A}_y(\mathbf{V}^n_{i,j,k})]\Delta^n_j - \frac{\Delta t}{2\Delta z}\mathbf{A}_z(\mathbf{V}^n_{i,j,k})\Delta^n_k,$$

$$\mathbf{V}^{n+1/2}_{i,j,k,T,B} = \mathbf{V}^n_{i,j,k} - \frac{\Delta t}{2\Delta x}\mathbf{A}_x(\mathbf{V}^n_{i,j,k})\Delta^n_i - \frac{\Delta t}{2\Delta y}\mathbf{A}_y(\mathbf{V}^n_{i,j,k})\Delta^n_j + \frac{1}{2}[\pm\mathbf{I} - \frac{\Delta t}{\Delta z}\mathbf{A}_z(\mathbf{V}^n_{i,j,k})]\Delta^n_k,$$

Corner Transport Upwind (CTU)



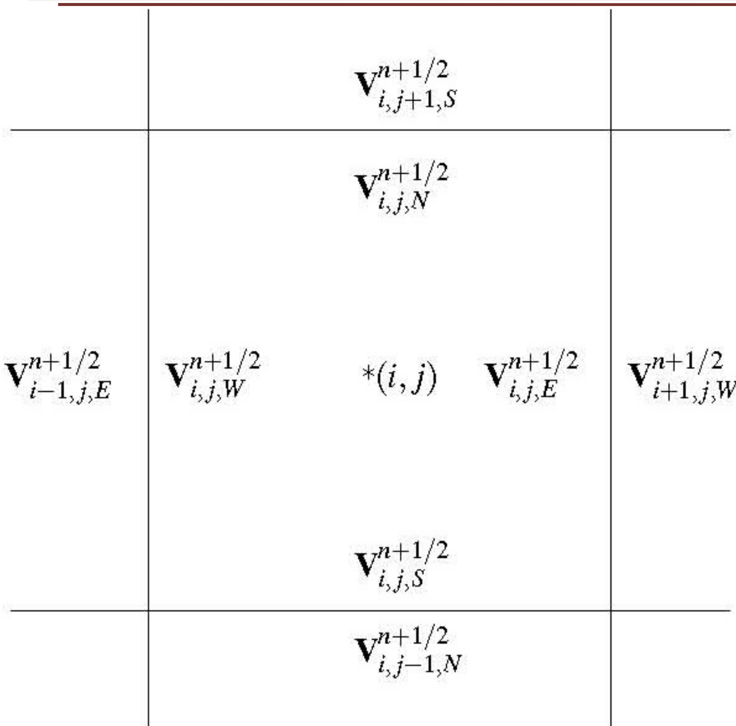
Linear system in 3D

$$\mathbf{V}^{n+1/2}_{i,j,k,E,W} = \overbrace{\boxed{\mathbf{V}^n_{i,j,k} + \frac{1}{2}\left[\pm\mathbf{I} - \frac{\Delta t}{\Delta x}\mathbf{A}_x(\mathbf{V}^n_{i,j,k})\right]\Delta^n_i}}^{\text{Normal predictor}} \overbrace{\boxed{- \frac{\Delta t}{2\Delta y}\mathbf{A}_y(\mathbf{V}^n_{i,j,k})\Delta^n_j - \frac{\Delta t}{2\Delta z}\mathbf{A}_z(\mathbf{V}^n_{i,j,k})\Delta^n_k}}^{\text{Transverse corrector}},$$

$$\mathbf{V}^{n+1/2}_{i,j,k,N,S} = \mathbf{V}^n_{i,j,k} - \frac{\Delta t}{2\Delta x}\mathbf{A}_x(\mathbf{V}^n_{i,j,k})\Delta^n_i + \frac{1}{2}\left[\pm\mathbf{I} - \frac{\Delta t}{\Delta y}\mathbf{A}_y(\mathbf{V}^n_{i,j,k})\right]\Delta^n_j - \frac{\Delta t}{2\Delta z}\mathbf{A}_z(\mathbf{V}^n_{i,j,k})\Delta^n_k,$$

$$\mathbf{V}^{n+1/2}_{i,j,k,T,B} = \mathbf{V}^n_{i,j,k} - \frac{\Delta t}{2\Delta x}\mathbf{A}_x(\mathbf{V}^n_{i,j,k})\Delta^n_i - \frac{\Delta t}{2\Delta y}\mathbf{A}_y(\mathbf{V}^n_{i,j,k})\Delta^n_j + \frac{1}{2}\left[\pm\mathbf{I} - \frac{\Delta t}{\Delta z}\mathbf{A}_z(\mathbf{V}^n_{i,j,k})\right]\Delta^n_k,$$

Corner Transport Upwind (CTU)



- ❑ Traditional approach (Colella 1990; Saltzman 1994)
  - ❑ Characteristic tracing for the normal predictor
  - ❑ Subsequent calls to Riemann solvers for transverse corrector

Normal predictor | Transverse corrector

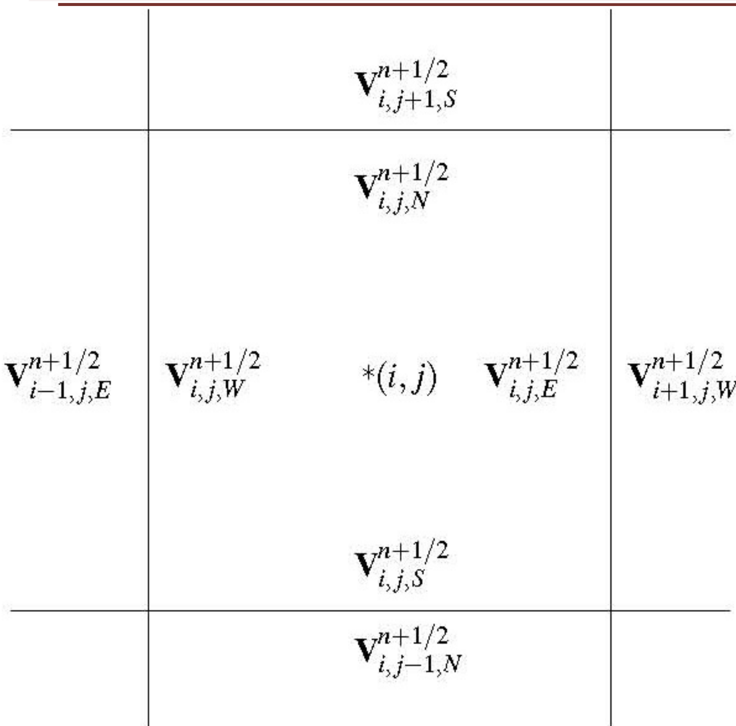
Linear system in 3D

$$\mathbf{V}^{n+1/2}_{i,j,k,E,W} = \boxed{\mathbf{V}^n_{i,j,k} + \frac{1}{2}[\pm\mathbf{I} - \frac{\Delta t}{\Delta x}\mathbf{A}_x(\mathbf{V}^n_{i,j,k})]\Delta^n_i} \boxed{- \frac{\Delta t}{2\Delta y}\mathbf{A}_y(\mathbf{V}^n_{i,j,k})\Delta^n_j - \frac{\Delta t}{2\Delta z}\mathbf{A}_z(\mathbf{V}^n_{i,j,k})\Delta^n_k}$$

$$\mathbf{V}^{n+1/2}_{i,j,k,N,S} = \mathbf{V}^n_{i,j,k} - \frac{\Delta t}{2\Delta x}\mathbf{A}_x(\mathbf{V}^n_{i,j,k})\Delta^n_i + \frac{1}{2}[\pm\mathbf{I} - \frac{\Delta t}{\Delta y}\mathbf{A}_y(\mathbf{V}^n_{i,j,k})]\Delta^n_j - \frac{\Delta t}{2\Delta z}\mathbf{A}_z(\mathbf{V}^n_{i,j,k})\Delta^n_k,$$

$$\mathbf{V}^{n+1/2}_{i,j,k,T,B} = \mathbf{V}^n_{i,j,k} - \frac{\Delta t}{2\Delta x}\mathbf{A}_x(\mathbf{V}^n_{i,j,k})\Delta^n_i - \frac{\Delta t}{2\Delta y}\mathbf{A}_y(\mathbf{V}^n_{i,j,k})\Delta^n_j + \frac{1}{2}[\pm\mathbf{I} - \frac{\Delta t}{\Delta z}\mathbf{A}_z(\mathbf{V}^n_{i,j,k})]\Delta^n_k,$$

Corner Transport Upwind (CTU)



- Traditional approach (Colella 1990; Saltzman 1994)
  - Characteristic tracing for the normal predictor
  - Subsequent calls to Riemann solvers for transverse corrector
- New approach (Lee and Deane 2009):
  - Characteristic tracing for BOTH normal predictor and transverse corrector!

Linear system in 3D

Normal predictor

Transverse corrector

$$\mathbf{V}^{n+1/2}_{i,j,k,E,W} = \boxed{\mathbf{V}^n_{i,j,k} + \frac{1}{2}\left[\pm\mathbf{I} - \frac{\Delta t}{\Delta x}\mathbf{A}_x(\mathbf{V}^n_{i,j,k})\right]\Delta^n_i} \boxed{- \frac{\Delta t}{2\Delta y}\mathbf{A}_y(\mathbf{V}^n_{i,j,k})\Delta^n_j - \frac{\Delta t}{2\Delta z}\mathbf{A}_z(\mathbf{V}^n_{i,j,k})\Delta^n_k}$$

$$\mathbf{V}^{n+1/2}_{i,j,k,N,S} = \mathbf{V}^n_{i,j,k} - \frac{\Delta t}{2\Delta x}\mathbf{A}_x(\mathbf{V}^n_{i,j,k})\Delta^n_i + \frac{1}{2}\left[\pm\mathbf{I} - \frac{\Delta t}{\Delta y}\mathbf{A}_y(\mathbf{V}^n_{i,j,k})\right]\Delta^n_j - \frac{\Delta t}{2\Delta z}\mathbf{A}_z(\mathbf{V}^n_{i,j,k})\Delta^n_k,$$

$$\mathbf{V}^{n+1/2}_{i,j,k,T,B} = \mathbf{V}^n_{i,j,k} - \frac{\Delta t}{2\Delta x}\mathbf{A}_x(\mathbf{V}^n_{i,j,k})\Delta^n_i - \frac{\Delta t}{2\Delta y}\mathbf{A}_y(\mathbf{V}^n_{i,j,k})\Delta^n_j + \frac{1}{2}\left[\pm\mathbf{I} - \frac{\Delta t}{\Delta z}\mathbf{A}_z(\mathbf{V}^n_{i,j,k})\right]\Delta^n_k,$$

Linearized System, cont'd

- A primitive form:

$$\mathbf{V} = (\rho, u, v, w, B_x, B_y, B_z, p)^T, \quad \boxed{\frac{\partial \mathbf{V}}{\partial t} + \mathbf{A}_x \frac{\partial \mathbf{V}}{\partial x}} + \mathbf{A}_y \frac{\partial \mathbf{V}}{\partial y} + \mathbf{A}_z \frac{\partial \mathbf{V}}{\partial z} = 0.$$

where the coefficient matrix is

$$\mathbf{A}_x = \begin{pmatrix} u & \rho & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & u & 0 & 0 & -\frac{B_x}{\rho} & \frac{B_y}{\rho} & \frac{B_z}{\rho} & \frac{1}{\rho} \\ 0 & 0 & u & 0 & -\frac{B_y}{\rho} & -\frac{B_x}{\rho} & 0 & 0 \\ 0 & 0 & 0 & u & -\frac{B_z}{\rho} & 0 & -\frac{B_x}{\rho} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & B_y & -B_x & 0 & -v & u & 0 & 0 \\ 0 & B_z & 0 & -B_x & -w & 0 & u & 0 \\ 0 & \gamma p & 0 & 0 & -k\mathbf{u}\cdot\mathbf{B} & 0 & 0 & u \end{pmatrix},$$

- First consider the evolution in the x-normal direction and treat the normal magnetic field separately from the other variables:

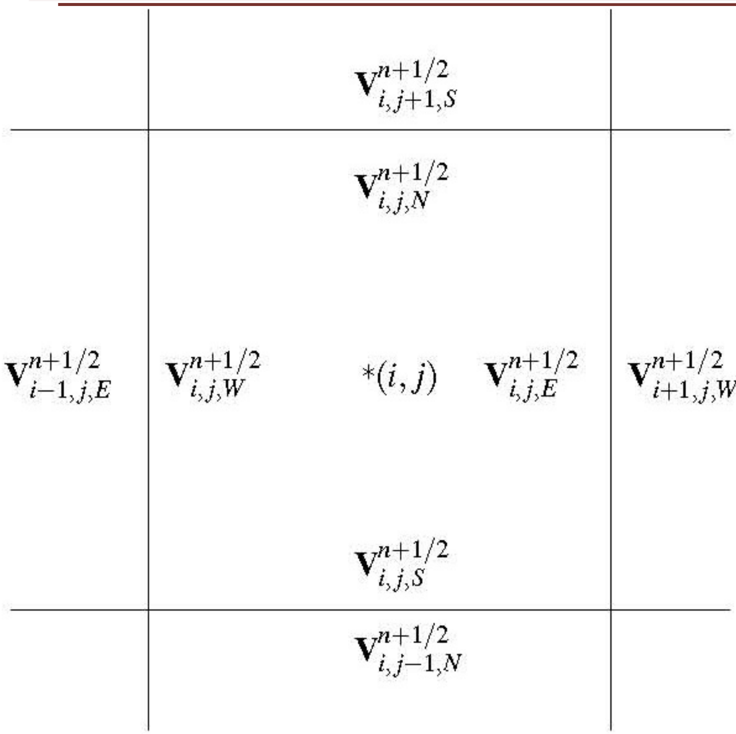
$$\boxed{\begin{bmatrix} \hat{\mathbf{V}}_x \\ B_x \end{bmatrix}_{i,j,k,E,W}^{n+1/2,\|} = \begin{bmatrix} \hat{\mathbf{V}}_x \\ B_x \end{bmatrix}_{i,j,k}^{n} + \frac{1}{2}\left(\pm \begin{bmatrix} \hat{\mathbf{I}} & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} - \frac{\Delta t}{\Delta x} \begin{bmatrix} \hat{\mathbf{A}}_x & \mathbf{A}_{B_x} \\ \mathbf{0} & 0 \end{bmatrix}_{i,j,k}^{n} \right) \bar{\Delta}_i^n,}$$

← Normal predictor

$$\bar{\mathbf{V}}_x = \begin{bmatrix} \hat{\mathbf{V}}_x \\ B_x \end{bmatrix} \text{ and } \bar{\mathbf{A}}_x = \begin{bmatrix} \hat{\mathbf{A}}_x & \mathbf{A}_{B_x} \\ \mathbf{0} & 0 \end{bmatrix}. \qquad \mathbf{A}_{B_x} = \left[0, -\frac{B_x}{\rho}, -\frac{B_y}{\rho}, -\frac{B_z}{\rho}, -v, -w, -k\mathbf{u}\cdot\mathbf{B}\right]^T$$

← MHD source term

Single-step data Reconstruction-evolution in USM



Normal Predictor

$$\begin{cases} \hat{\mathbf{V}}^{n+1/2,\|}_{x,i,j,k,E,W} = \hat{\mathbf{V}}^{n}_{x,i,j,k} + \frac{1}{2}\left(\pm\hat{\mathbf{I}} - \frac{\Delta t}{\Delta x}\hat{\mathbf{A}}_x\right)^{n}_{i,j,k}\hat{\Delta}^{n}_{i} - \frac{\Delta t}{2\Delta x}(\mathbf{A}_{B_x})^{n}_{i,j,k}\Delta B^{n}_{x,i}, \\ (B_x)^{n+1/2,\|}_{i,j,k,E,W} = B^{n}_{x,i,j,k} \pm \frac{1}{2}\Delta B^{n}_{x,i}, \end{cases}$$

Characteristic Tracing

$$\hat{\mathbf{V}}^{n+1/2,\|}_{x,i,j,k,E} = \hat{\mathbf{V}}^{n}_{x,i,j,k} + \frac{1}{2}\sum_{k;\lambda^{k}_{i,j,k}>0}\left(1 - \frac{\Delta t}{\Delta x}\lambda^{k}_{i,j}\right)\mathbf{r}^{k}_{x,i,j,k}\hat{\Delta}\alpha^{n}_{i} - \frac{\Delta t}{2\Delta x}(\mathbf{A}_{B_x})^{n}_{i,j}\Delta B^{n}_{x,i},$$

Characteristic tracing for Transverse corrector

- A jump relationship:

$$\hat{\mathbf{A}}_y \hat{\mathbf{V}}_l + \sum_{m=1}^{m_0-1} \lambda^m \mathbf{r}^m \tilde{\Delta}\alpha = \hat{\mathbf{A}}_y \hat{\mathbf{V}}_r - \sum_{m=m_0}^{7} \lambda^m \mathbf{r}^m \tilde{\Delta}\alpha. \tag{42}$$

The property of conservation across discontinuities of the Roe matrix $\hat{\mathbf{A}}_y$ (see (11) and (16)) [36, 56] gives

$$\hat{\mathbf{A}}_y \hat{\Delta} = \hat{\mathbf{A}}_y (\hat{\mathbf{V}}_r - \hat{\mathbf{V}}_l) = G(\hat{\mathbf{V}}_r) - G(\hat{\mathbf{V}}_l) = \mathbf{G}_{i,j+1/2} - \mathbf{G}_{i,j-1/2}. \tag{43}$$

From (42) and (43), the upwind flux gradient can be replaced by

$$\mathbf{G}_{i,j+1/2} - \mathbf{G}_{i,j-1/2} = \sum_{m=1}^{7} \lambda^m \mathbf{r}^m \tilde{\Delta}\alpha, \tag{44}$$

where the first-order upwind slope limiter $\tilde{\Delta}$ is applied to each characteristic variable α by

$$\tilde{\Delta}\alpha = \begin{cases} \mathbf{l}^m \cdot \hat{\Delta}^n_+ & \text{if} \quad \lambda^m < 0 \\ \mathbf{l}^m \cdot \hat{\Delta}^n_- & \text{if} \quad \lambda^m > 0 \end{cases}. \tag{45}$$

Reduced 3D CTU in USM

Characteristic Tracing for Normal Predictor

$$\hat{\mathbf{V}}^{n+1/2,\|}_{x,i,j,k,E} = \hat{\mathbf{V}}^{n}_{x,i,j,k} + \frac{1}{2}\sum_{k;\lambda^k_{i,j,k}>0}\left(1-\frac{\Delta t}{\Delta x}\lambda^k_{i,j}\right)\mathbf{r}^k_{x,i,j,k}\hat{\Delta}\alpha^n_i - \frac{\Delta t}{2\Delta x}(\mathbf{A}_{B_x})^n_{i,j}\Delta B^n_{x,i},$$

$$\hat{\Delta}\alpha^n_i = \mathtt{TVD\_Limiter}\left[\mathbf{l}^k_{x,i,j,k}\cdot\hat{\Delta}^n_{i,+},\mathbf{l}^k_{x,i,j,k}\cdot\hat{\Delta}^n_{i,-}\right].$$

Transverse Corrector

$$\mathbf{V}^{n+1/2,y}_{i,j,k,E,W} = \mathbf{V}^{n+1/2,\|}_{i,j,k,E,W} - \frac{\Delta t}{2\Delta y}\mathbf{A}_y(\mathbf{V}^n_{i,j,k})\Delta^n_j,$$

$$\mathbf{V}^{n+1/2}_{i,j,k,E,W} = \mathbf{V}^{n+1/2,y}_{i,j,k,E,W} - \frac{\Delta t}{2\Delta z}\mathbf{A}_z(\mathbf{V}^n_{i,j,k})\Delta^n_j.$$

$$\hat{\mathbf{V}}^{n+1/2}_{y,i,j,k,E,W} = \hat{\mathbf{V}}^{n+1/2,\|}_{y,i,j,k,E,W} - \frac{\Delta t}{2\Delta y}\sum_{k=1}^{7}\lambda^k_{y,i,j,k}\mathbf{r}^k_{y,i,j,k}\tilde{\Delta}\alpha^n_j - \frac{\Delta t}{2\Delta y}(\mathbf{A}_{B_y})^n_{i,j,k}\Delta B^n_{y,j},$$

$$\tilde{\Delta}\alpha^n_j = \mathtt{Upwinding}\left[\mathbf{l}^k_{y,i,j,k}\cdot\hat{\Delta}^n_{j,+},\mathbf{l}^k_{y,i,j}\cdot\hat{\Delta}^n_{j,-}\right]$$

Full 3D CTU in USM

Characteristic Tracing for Normal Predictor

$$\hat{\mathbf{V}}^{n+1/2,\|}_{x,i,j,k,E} = \hat{\mathbf{V}}^{n}_{x,i,j,k} + \frac{1}{2}\sum_{k;\lambda^k_{i,j,k}>0}\left(1-\frac{\Delta t}{\Delta x}\lambda^k_{i,j}\right)\mathbf{r}^k_{x,i,j,k}\hat{\Delta}\alpha^n_i - \frac{\Delta t}{2\Delta x}(\mathbf{A}_{B_x})^n_{i,j}\Delta B^n_{x,i},$$

$$\hat{\Delta}\alpha^n_i = \mathtt{TVD\_Limiter}\left[\mathbf{l}^k_{x,i,j,k}\cdot\hat{\Delta}^n_{i,+}, \mathbf{l}^k_{x,i,j,k}\cdot\hat{\Delta}^n_{i,-}\right].$$

Transverse Corrector

$$\mathbf{V}^{n+1/2,y}_{i,j,k,E,W} = \mathbf{V}^{n+1/2,\|}_{i,j,k,E,W} - \frac{\Delta t}{2\Delta y}\mathbf{A}_y\left(\boxed{\mathbf{V}^n_{i,j,k}}\right)\Delta^n_j,$$

$$\mathbf{V}^{n+1/2}_{i,j,k,E,W} = \mathbf{V}^{n+1/2,y}_{i,j,k,E,W} - \frac{\Delta t}{2\Delta z}\mathbf{A}_z\left(\boxed{\mathbf{V}^n_{i,j,k}}\right)\Delta^n_j.$$

$$\hat{\mathbf{V}}^{n+1/2}_{y,i,j,k,E,W} = \hat{\mathbf{V}}^{n+1/2,\|}_{y,i,j,k,E,W} - \frac{\Delta t}{2\Delta y}\sum_{k=1}^{7}\lambda^k_{y,i,j,k}\mathbf{r}^k_{y,i,j,k}\tilde{\Delta}\alpha^n_j - \frac{\Delta t}{2\Delta y}(\mathbf{A}_{B_y})^n_{i,j,k}\Delta B^n_{y,j},$$

$$\tilde{\Delta}\alpha^n_j = \mathtt{Upwinding}\left[\mathbf{l}^k_{y,i,j,k}\cdot\hat{\Delta}^n_{j,+}, \mathbf{l}^k_{y,i,j}\cdot\hat{\Delta}^n_{j,-}\right]$$

Full CTU diagonal coupling

$$\mathbf{V}^{n+1/3,z}_{i,j,k} = \mathbf{V}^n_{i,j,k} - \frac{\Delta t}{3\Delta z}(\mathbf{A}_z)^n_{i,j,k}\Delta^n_k, \quad \mathbf{V}^{n+1/3,y}_{i,j,k} = \mathbf{V}^n_{i,j,k} - \frac{\Delta t}{3\Delta y}(\mathbf{A}_y)^n_{i,j,k}\Delta^n_j.$$