Numerical Solutions for Hyperbolic Systems of Conservation Laws:
from Godunov Method to Adaptive Mesh Refinement

Romain Teyssier
CEA Saclay
- Euler equations, MHD, waves, hyperbolic systems of conservation laws, primitive form, conservative form, integral form

- Advection equation, exact solution, characteristic curve, Riemann invariant, finite difference scheme, modified equation, Von Neumann analysis, upwind scheme, Courant condition, Second order scheme

- Finite volume scheme, Godunov method, Riemann problem, approximate Riemann solver, Second order scheme, Slope limiters, Characteristic tracing

- Multidimensional scheme, directional splitting, Godunov, Runge-Kutta, CTU, 3D slope limiting

- AMR, patch-based versus cell-based, octree structure, graded octree, flux correction, EMF correction, restriction and prolongation, divB conserving interpolation

- Parallel computing with the RAMSES code
Patch-based versus tree-based
A few AMR codes in astrophysics

**ENZO**: Greg Bryan, Michael Norman…

**ART**: Andrey Kravtsov, Anatoly Klypin

**RAMSES**: Romain Teyssier

**NIRVANA**: Udo Ziegler

**AMRVAC**: Gabor Thot and Rony Keppens

**FLASH**: The Flash group (PARAMESH lib)

**ORION**: Richard Klein, Chris McKee, Phil Colella

**PLUTO**: Andrea Mignone (CHOMBO lib, Phil Colella)

**CHARM**: Francesco Miniati (CHOMBO lib, Phil Colella)

**ASTROBear**: Adam Frank…

Any other code in the audience?
Fully Threaded Tree (Khokhlov 98). Cartesian mesh refined on a cell by cell basis.

octs: small grid of 8 cells

Pointers (arrays of index)
- 1 parent cell
- 6 neighboring parent cells
- 8 children octs
- 2 linked list indices

Cell-centered variables are updated level by level using linked lists.

Cost = 2 integer per cell.

Optimize mesh adaptation to complex flow geometries, but CPU overhead compared to unigrid can be as large as 50%.

2 type of cell: “leaf” or active cell
“split” or inactive cell
Refinement rules for graded octree

Compute the refinement map: flag = 0 or 1

Step 1: mesh consistency
if a split cell contains at least one split or marked cell, then mark the cell with flag = 1 and mark its 26 neighbors

Step 2: physical criteria
quasi-Lagrangian evolution, Jeans mass
geometrical constraints (zoom)
Truncation errors, density gradients…

Step 3: mesh smoothing
apply a dilatation operator (mathematical morphology) to regions marked for refinement → convex hull
Godunov schemes and AMR

Berger & Oliger (84), Berger & Collela (89)

Prolongation (interpolation) to finer levels
- fill buffer cells (boundary conditions)
- create new cells (refinements)

Restriction (averaging) to coarser levels
- destroy old cells (de-refinements)

Flux correction at level boundary

\[
(F^{n+1/2,\ell}_{i+1/2,j}) = \frac{(F^{n+1/2,\ell+1}_{i+1/2,j-1/4}) + (F^{n+1/2,\ell+1}_{i+1/2,j+1/4})}{2}
\]

Careful choice of interpolation variables (conservative or not ?)

Several interpolation strategies (with $R^T P = I$):
- straight injection
- tri-linear, tri-parabolic reconstruction
Buffer cells provide boundary conditions for the underlying numerical scheme. The number of required buffer cells depends on the kernel of the chosen numerical method. *The kernel is the ensemble of cells on the grid on which the solution depends.*

- **First Order Godunov**: 1 cell in each direction
  \[ u_{i}^{n+1} = u_{i}^{n}(1 - C) + u_{i-1}^{n}C \]

- **Second order MUSCL**: 2 cells in each direction
  \[ u_{i}^{n+1} = u_{i}^{n}(1 - C) + u_{i-1}^{n}C - \frac{C}{2}(1 - C)(\Delta u_{i} - \Delta u_{i-1}) = 0 \]

- **Runge-Kutta or PPM**: 3 cells in each direction

Simple octree AMR requires 2 cells maximum. For higher-order schemes (WENO), we need to have a different data structure (patch-based AMR or augmented octree AMR).
Time integration: single time step or recursive sub-cycling

- froze coarse level during fine level solves (one order of accuracy down!)
- average fluxes in time at coarse fine boundaries

\[
(F^{n+1/2, \ell}_{i+1/2,j}) = \frac{1}{\Delta t_{1}^{\ell+1} + \Delta t_{2}^{\ell+1}} \left( \Delta t_{1}^{\ell+1} \frac{(F_{i+1/2,j-1/4}^{n+1/4,\ell+1}) + (F_{i+1/2,j+1/4}^{n+1/4,\ell+1})}{2} + \Delta t_{2}^{\ell+1} \frac{(F_{i+1/2,j-1/4}^{n+3/4,\ell+1}) + (F_{i+1/2,j+1/4}^{n+3/4,\ell+1})}{2} \right)
\]
The AMR catastrophe

First order scheme:
\[ u_{i+1/2}^{n+1/2} = u_i^n + (1 - C) \frac{\Delta x}{2} \left( \frac{\partial u}{\partial x} \right)_i \]

Second order scheme:
\[ u_{i+1/2}^{n+1/2} = u_{i-\alpha}^n + (2\alpha - 1 - C) \frac{\Delta x}{2} \left( \frac{\partial u}{\partial x} \right)_{i-1} \]

Assume a and C>0.

First order scheme: the AMR extension is not consistent at level boundary.

Second order scheme: for \( \alpha = 1.5 \), AMR is unstable at level boundary.

Solutions: 1- refine gradients, 2- enforce first order, 3- add artificial diffusion
Shock wave propagating through level boundary
Shock-tube test with regular grid

100 Eulerian mesh points with PLM
Shock-tube test with AMR

120 AMR mesh points with RAMSES
Maximum numerical dissipation occurs at the 2 fluids interface.

The optimal refinement strategy is based on density gradients.

The number of required cells is directly related to the fractal exponent $n$ of the 2D surface.

$$N_{cell} \propto (\Delta x)^{-n}$$
Particle-Mesh on AMR grids:
Cloud size equal to the local mesh spacing

Poisson solver on the AMR grid
Multigrid or Conjugate Gradient
Interpolation to get Dirichlet boundary conditions (one way interface)

Quasi-Lagrangian mesh evolution:
roughly constant number of particles per cell

\[ n = \frac{\rho_{DM}}{m_{DM}} + \frac{\rho_{gas}}{m_{gas}} + \frac{\rho_\star}{m_\star} \]

Trigger new refinement when \( n > 10-40 \) particles. The fractal dimension is close to 1.5 at large scale (filaments) and is less than 1 at small scales (clumps).
RAMSES: a parallel graded octree AMR

- Tree-based AMR (octree structure): the cartesian mesh is recursively refined on a cell by cell basis.
- Full connectivity: each “oct” have direct access to neighboring parent cells and to children “octs”. (memory overhead: 2 integers per cell).
  → Optimize the mesh adaptivity to complex geometries, but CPU overhead can be as large as 50%.

Code is freely available http://www-dapnia.cea.fr/Projets/COAST

N body module: Particle-Mesh method on AMR grids (similar to the ART code).
  Poisson equation solved using Conjugate Gradient and Multigrid.

Hydro module: Unsplit second order Godunov method: Riemann solver with piecewise linear reconstruction (option: MUSCL or PLMDE).

Time integration: Single time step or W cycle (fine levels subcycling)

Other: Cooling & UV heating, Zoom simulation technology
  → MPI based parallel implementation

5th JETSET School
Romain Teyssier
Domain decomposition for parallel computing

Parallel computing using the MPI library with a domain decomposition based on the Peano-Hilbert curve.

Algorithm inspired by gravity solvers (tree codes).
Use locally essential tree.

Tested and operational up to 6144 core.
Scaling depends on problem size and complexity.


Locally essential trees

Salmon 90, Warren 92, Dubinsky 96

Each processor octree is surrounded by ghost cells (local copy of distant processor octrees) so that the resulting local octree contains all the necessary information.

Domain decomposition over 3 processors

Locally essential tree in processor #1

Locally essential tree in processor #2

Locally essential tree in processor #3
Dynamic partitioning in RAMSES

Several cell ordering methods:
1- column, row or plane major
2- Hilbert or Morton
3- User defined (angular, column+Hilbert…)

Dynamic partitioning is performed every N steps by sorting each cell along chosen ordering and redistributing the mesh across processors. Usually, a good choice is N=10 (10% overhead).
Are there any optimal load balancing strategy?
Are there any optimal load balancing strategy?

Recursive bisection
Are there any optimal load balancing strategy?

Hilbert ordering
Are there any optimal load balancing strategy?

Angular ordering
Ideal MHD equations in conservative form

Mass conservation
\[ \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0, \]

Linear momentum conservation
\[ \frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v} \mathbf{v} - \mathbf{B} \mathbf{B}) + \nabla P_{\text{tot}} = 0, \]

Total energy conservation
\[ \frac{\partial E}{\partial t} + \nabla \cdot [(E + P_{\text{tot}}) \mathbf{v} - \mathbf{B} (\mathbf{B} \cdot \mathbf{v})] = 0, \]

Magnetic flux conservation
\[ \frac{\partial \mathbf{B}}{\partial t} + \nabla \cdot (\mathbf{v} \mathbf{B} - \mathbf{B} \mathbf{v}) = 0. \]
\[ \nabla \cdot \mathbf{B} = 0 \]

Total pressure
\[ P_{\text{tot}} = P + \frac{\mathbf{B} \cdot \mathbf{B}}{2}, \]

Total energy
\[ E = \epsilon + \rho \frac{\mathbf{v} \cdot \mathbf{v}}{2} + \frac{B \cdot B}{2}, \]
Euler equations using finite volumes: decades of experience in robust advection and shock-capturing schemes (Godunov, MUSCL, PPM).

Ideal MHD: Euler system augmented by the induction equation

**Finite volume and cell-centered schemes** Crockett et al. 2005
- div B cleaning using Poisson solver
- div B waves (Powell’s 8 waves formulation)
- div B damping

**Constrained Transport and staggered grid** (Yee 66; Evans & Hawley 88)
- 1D Godunov fluxes to compute the EMF Balsara, Spicer 99
- 2D Riemann solver to get the EMF Londrillo, DelZanna 01,05, Ziegler 04,05
- High-order extension of Balsara’s scheme Gardiner & Stone 05

**Constrained Transport for tree-based AMR**
- Teyssier, Fromang & Dormy 2006
- Fromang, Hennebelle & Teyssier 2006
The induction equation in 2D

\[ \frac{\partial \mathbf{B}}{\partial t} = \nabla \times \mathbf{E} + \eta \Delta \mathbf{B}, \quad \mathbf{E} = \mathbf{v} \times \mathbf{B}, \quad \nabla \cdot \mathbf{B} = 0. \]

Finite-surface approximation (Constrained Transport)

\[ \langle B_x \rangle_{i+\frac{1}{2},j}^n = \frac{1}{\Delta y} \int_{y_i-\frac{1}{2}}^{y_i+\frac{1}{2}} B_x(x_{i+\frac{1}{2}}, y, t^n) \, dy, \quad \langle B_y \rangle_{i,j+\frac{1}{2}}^n = \frac{1}{\Delta x} \int_{x_i-\frac{1}{2}}^{x_{i+\frac{1}{2}}} B_y(x, y_{j+\frac{1}{2}}, t^n) \, dx. \]

Integral form using Stoke’s theorem

\[ \langle B_x \rangle_{i+\frac{1}{2},j}^{n+1} = \langle B_x \rangle_{i+\frac{1}{2},j}^n + \frac{\Delta t}{\Delta y} \left( \langle E_z \rangle_{i+\frac{1}{2},j+\frac{1}{2}}^{n+\frac{1}{2}} - \langle E_z \rangle_{i+\frac{1}{2},j-\frac{1}{2}}^{n+\frac{1}{2}} \right), \]
\[ \langle B_y \rangle_{i,j+\frac{1}{2}}^{n+1} = \langle B_y \rangle_{i,j+\frac{1}{2}}^n - \frac{\Delta t}{\Delta x} \left( \langle E_z \rangle_{i+\frac{1}{2},j+\frac{1}{2}}^{n+\frac{1}{2}} - \langle E_z \rangle_{i-\frac{1}{2},j+\frac{1}{2}}^{n+\frac{1}{2}} \right). \]

For piecewise constant initial data, the flux function is self-similar at corner points

\[ \langle E_z \rangle_{i+\frac{1}{2},j+\frac{1}{2}}^{n+1} = E_z(U^n_{i+\frac{1}{2},j+\frac{1}{2}}(0,0)). \]

For pure induction, the 2D Riemann problem has the following exact (upwind) solution:

\[ \langle E_z \rangle_{i\frac{1}{2},j\frac{1}{2}}^{n+\frac{1}{2}} = \frac{\langle B_y \rangle_{i+1,j+\frac{1}{2}}^n + \langle B_y \rangle_{i,j+\frac{1}{2}}^n - \langle B_x \rangle_{i+\frac{1}{2},j+1}^n - \langle B_x \rangle_{i+\frac{1}{2},j-\frac{1}{2}}^n}{2} - |a| \frac{\langle B_y \rangle_{i+1,j+\frac{1}{2}}^n - \langle B_y \rangle_{i,j+\frac{1}{2}}^n}{2} + |v| \frac{\langle B_x \rangle_{i+\frac{1}{2},j+1}^n - \langle B_x \rangle_{i+\frac{1}{2},j-\frac{1}{2}}^n}{2}. \]

Numerical diffusivity \( \eta_x = |u| \Delta x \) and \( \eta_y = |v| \Delta y \).
« Divergence-free preserving » restriction and prolongation operators


Flux conserving interpolation and averaging within cell faces using TVD slopes in 2 dimensions

EMF correction for conservative update at coarse-fine boundaries

\[
\langle E_z \rangle_k^\ell \Delta z_\ell \Delta t^{2n} = \langle E_z \rangle_{2k}^{\ell+1} \Delta z_{\ell+1} \Delta t^{2n+1} + \langle E_z \rangle_{2k+1}^{\ell+1} \Delta z_{\ell+1} \Delta t^{2n+1}
\]
\( \alpha = \pi \): 2 solutions: 2 shocks or 1 c.w.
\( \alpha \neq \pi \): 2 shocks only

Dissipation properties are crucial.
Only AMR can resolve scales small enough within reasonable CPU time.

\[
W = (\rho, v_x, v_y, v_z, P, B_x, B_y, B_z)^T
\]
\[
W^L = (1, 0, 0, 1, 1, 0)^T
\]
\[
W^R = (0.2, 0, 0, 0.2, 1, \cos \alpha, \sin \alpha)^T
\]
Field loop advection (Gardiner & Stone 2005)
Current sheet and magnetic reconnection
We report on a 70 billions particles N-body simulation for a 2 Gpc/h periodic box in a LCDM universe.

We use a new French supercomputer BULL Novascale 3045 recently commissioned at CCRT (Centre de Calcul Recherche et Technologie, CEA).

We ran RAMSES in pure N-body mode with 6144 processors for 2 months.

We have simulated half of the observable universe, with “resolved” dark matter halos as small as the Milky Way.

Using our light cone, we have computed a full sky convergence map for simulating future weak-lensing surveys like DUNE or LSST.

The Horizon Simulation

direct summation
P³M or AP³M
parallel or vectorized P³M
distributed-memory parallel Tree
distributed-memory parallel Tree PM
distributed-memory parallel PM AMR

Courtesey of V. Springel

5th JETSET School
Romain Teyssier
Initial conditions on a $4096^3$ have been generated using mpgrafic and FFTW.

MPI communication routines in RAMSES have been optimized (reduce and broadcast) with BULL R&D teams.

140 billions AMR cells to compute the gravitational acceleration.

Starting with a base grid of $4096^3$ cells, we used 6 additional level of refinements for a formal resolution of $262144^3$ (7 kpc/h).

$10^4$ time steps for 70 billion particle trajectories.

737 main steps (1 hour of elapsed time) for which 3 Tb of data were dumped (checkpoint restart) in 5 minutes using the LUSTRE parallel file system.

We have generated data for 5 snapshots and 2 light cones (50 Tb).

- 1 light cone up to redshift 1 for 4Π.
- 1 light cone up to redshift 7 for 500 square degree.