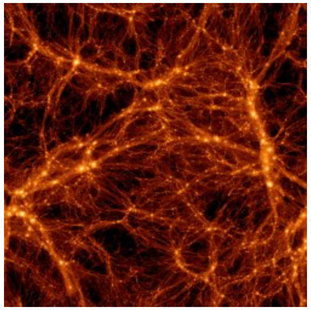
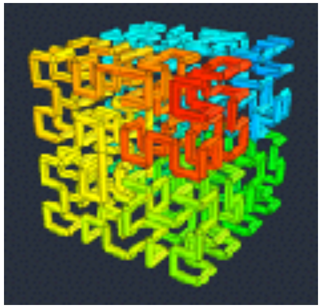
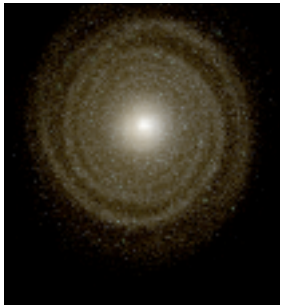
The Astrophysics of Galaxy Formation: Numerical Methods

Romain Teyssier

University of Zurich

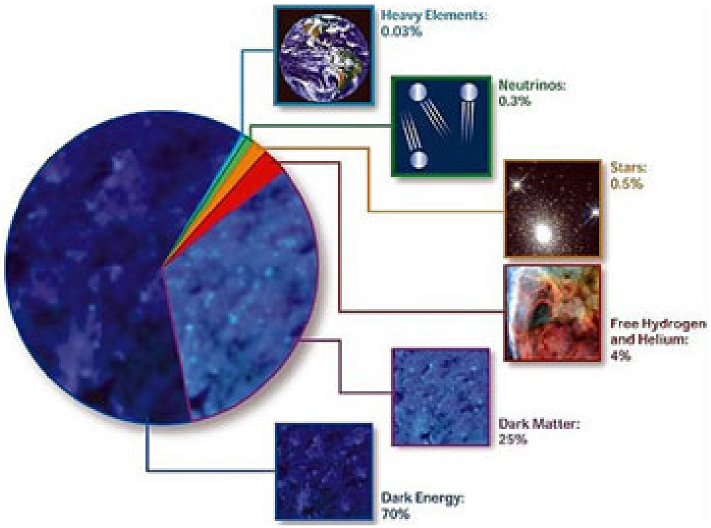




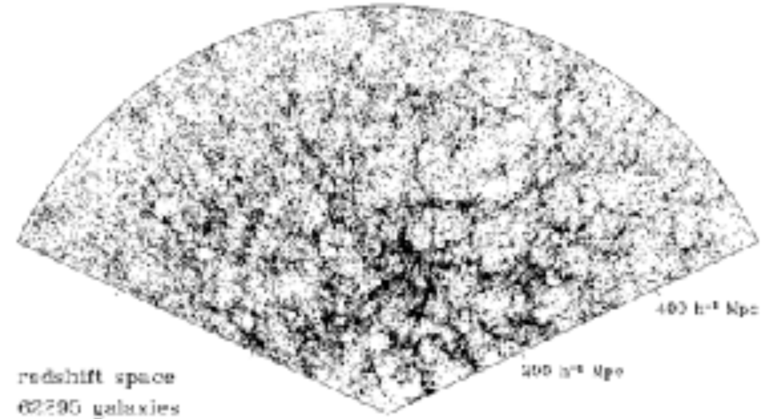
Outline

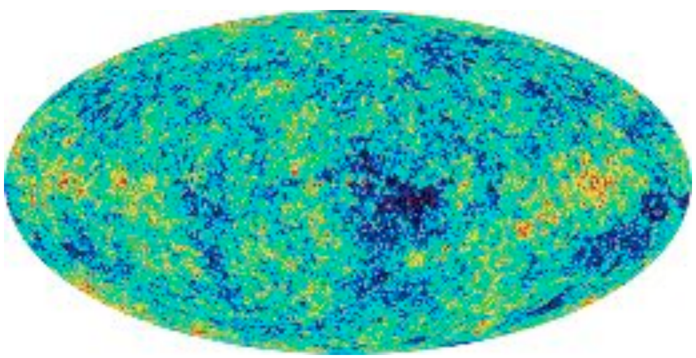
- Cosmological simulations: basic principles
- N body methods and gravity solvers
- High Performance Computing
- Hydrodynamics
- Magneto-Hydrodynamics
- Radiative Transfer
- Subgrid Physics

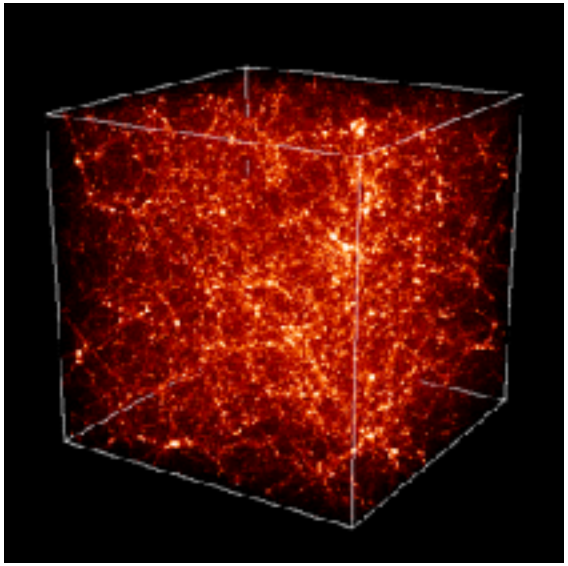
Basic principles of cosmological galaxy formation simulations

Cosmological simulations









From Gaussian random fields to galaxies: non-linear dynamics of gravitational instability with N-body and hydrodynamics codes.

Cosmological initial conditions

Download Gaussian random fields generators from various sources:
- original code from Ed Bertschinger : http://web.mit.edu/edbert/grafic2.101.tar.gz
- MPI version from Simon Prunet : http://www2.iap.fr/users/pichon/mpgrafic.html
- C++ MPI version from Doug Potter: http://sourceforge.net/projects/grafic/
- MUSIC: a new IC generator by Oliver Hahn: http://www.stanford.edu/~ohahn/

Note: grafic1 and mpgrafic generate only periodic unigrid IC.
grafic2, grafic++ and music generate nested-grid IC: zoom simulations.
mpgrafic described in Prunet et al., ApJS, 2008, 178, 179
music described in Hahn & Abel, MNRAS, 2011, 415, 210

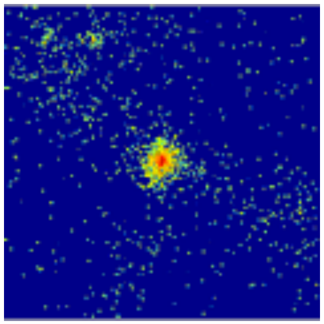
Cosmological inputs are:
- analytical power spectrum from Eisenstein & Hu, ApJ, 1998, 496, 605
- cosmo parameters: omega_m, omega_lambda, omega_b, n_s, sigma_8
- run parameters: box size, grid size, noise random seed or external white noise file

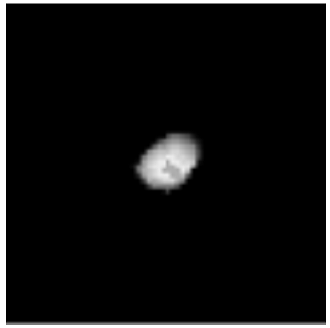
grafic format features 7 binary unformatted fortran files:
ic_velcx, ic_velcy, ic_velcz, ic_deltab, ic_velbx, ic_velby, ic_velbz

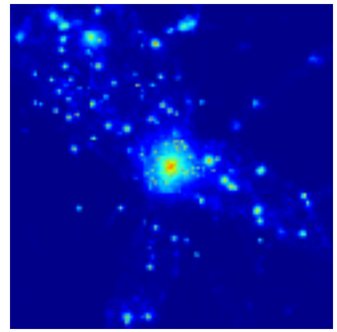
Cosmological zoom initial conditions

1: detect first one halo of interest in a cosmological simulation.

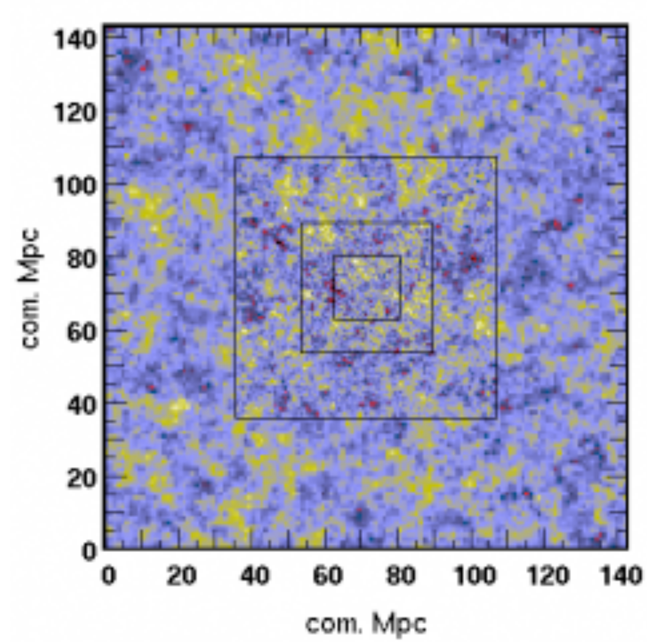
2: compute the Lagrangian volume in the low resolution IC







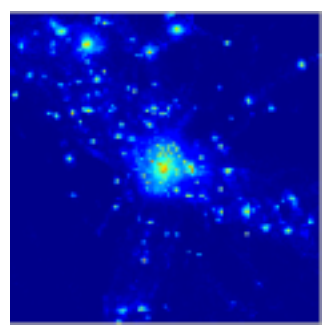
3: generate high-resolution IC by adding high frequency waves to the low resolution initial Gaussian random field



4: use the Lagrangian volume as a map to initialize high resolution particles.

5: do the high resolution simulation and check for contamination

6: eventually, compute a better initial Lagrangian volume and re-do the simulation



Expanding Universe and comoving coordinates

Expansion governed by Friedman-Lemaitre equations: $a(t)$ and $H(t)$

Define comoving coordinates: $\mathrm{x} = \dfrac{\mathrm{r}}{a(t)}$ $\quad \tilde{\rho}(\mathrm{x}, t) = \rho(\mathrm{r}, t) a(t)^3$

Define peculiar velocity: $\mathrm{v} = \mathrm{u} - H(t)\mathrm{r} \quad \tilde{\mathrm{v}} = \mathrm{v} a(t)$

Define supercomoving time (Martel and Shapiro 1998): $\mathrm{d}\tau = \dfrac{\mathrm{d}t}{a(t)^2}$

Then magic happens ! Fluid equations are equal to the one without expansion.

The only difference being Poisson's equation: $\tilde{\phi} = \phi a(t)^2$

$$\tilde{\Delta}\tilde{\phi} = \frac{3}{2} a(t) \Omega_m \frac{\rho - \bar{\rho}}{\bar{\rho}}$$

provided $p = 0$ or $p = (\gamma - 1)e$ with $\gamma = 5/3$ and $\tilde{p} = p a(t)^5$

Dynamics of multiple fluids

Different fluids are modelled using different technics.

1. Dark matter as a collisionless fluid (Vlasov equation)
2. Gas as a compressible ideal gas (Euler equations)
3. Radiation as optically thin, in low density gas, or thick at higher density
4. Magnetic fields as a divergence free vector field
5. Stars as a collisionless fluid (Vlasov equation)
6. Supermassive black holes as individual accreting particles
7. Cosmic rays as an additional energy variables or as a new fluid
8. Metals and dust grains as passive scalars or as new fluids
9. Various chemical species as passive scalars and associated reactions
10. Massive neutrinos as a quasi-relativistic fluid

N body codes and gravity solvers

The Vlasov-Poisson equation

Collisionless limit of the Boltzmann equation:

$$\frac{Df}{Dt} = \frac{\partial}{\partial t} f(\mathbf{x}, \mathbf{v}, t) + \mathbf{v}\frac{\partial}{\partial \mathbf{x}} f + \mathbf{a}\frac{\partial}{\partial \mathbf{v}} f = 0$$

Liouville theorem: number of particles in conserved in phase-space

Gravitational acceleration is given by the Poisson equation

$$\Delta\Phi(\mathbf{x}, t) = 4\pi G m \left(n(\mathbf{x}, t) - \bar{n}\right) \qquad n(\mathbf{x}, t) = \int f(\mathbf{x}, \mathbf{v}, t)\mathrm{d}^3\mathrm{v}$$

3 solution strategies:

- pure fluid on a 6D grid (Yoshikawa et al. 2013) or on a cold 3D manifold (Abel et al. 2012)
- pure N body using direct force computations or fast multipole methods (Barnes & Hut 1986; Bouchet & Hernquist 1988)
- mixture of the 2: the Particle-Mesh method (Hockney & Eastwood 1988)

A history of N body codes

N body code: particle trajectory integrator coupled to your favorite gravity solver

$$\frac{d\mathbf{x}_p}{dt} = \mathbf{v}_p \quad \text{and} \quad \frac{d\mathbf{v}_p}{dt} = -\nabla_x \phi$$

How to compute the gravitational potential ?

Popular techniques are, in chronological order:

1. Direct N body method, scales as N^2
2. PM: Fast Fourier Transform on a grid, O(N log N), low resolution
3. P3M (PP+PM): O(N log N) on large scales, N^2 on small scales, low resolution
4. Tree codes, O(N log N), high resolution. Variant: Tree-PM
5. Adaptive Mesh Refinement (AMR) with Multigrid solver, O(N), high resolution
6. Fast Multipole Method (FMM), O(N), high resolution

Direct N body solvers

Popular code is **`NBODY6`** (Aarseth 1999, Spurzem et al. 2012)

It is used to model collisional systems like globular clusters.

Recent versions are based on GPU acceleration, very efficient for N^2 calculations.

Current performance allows to model one million particles on parallel GPU clusters.

Time integration performed using a fourth-order Hermite integrator.

Time discretisation is distributed over a hierarchy of time steps (factors of 2).

This is the only viable approach for (although see Dehnen 2014):

1. very accurate, long time integrations
2. binary systems
3. tight orbit around SMBH.

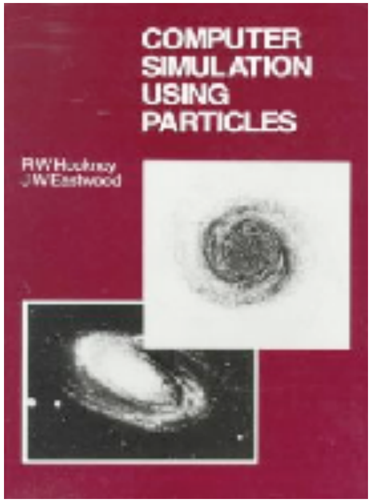
The Particle Mesh method

Introduce a regular Cartesian grid covering the entire domain.

1- Compute the mass density field on the grid from the particle distribution

2- Solve for the Poisson equation on the grid

3- Interpolate the force back to the particle position



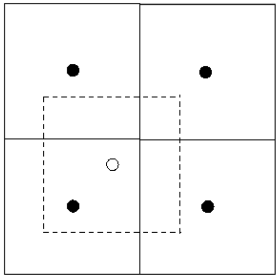
Hockney, R.W., Eastwood, J.W., “Computer Simulation Using Particles”, CRC Press (1985)

Assign to each particle a « shape » with popular choices

1. Nearest Grid Point $S(\mathrm{x}) = \frac{1}{\Delta x}\delta\left(\frac{\mathrm{x}}{\Delta x}\right)$
2. Cloud in Cell $S(\mathrm{x}) = \frac{1}{\Delta x}\Pi\left(\frac{\mathrm{x}}{\Delta x}\right)$
3. Triangular Shape Cloud $S(\mathrm{x}) = \frac{1}{\Delta x}\Delta\left(\frac{\mathrm{x}}{\Delta x}\right)$

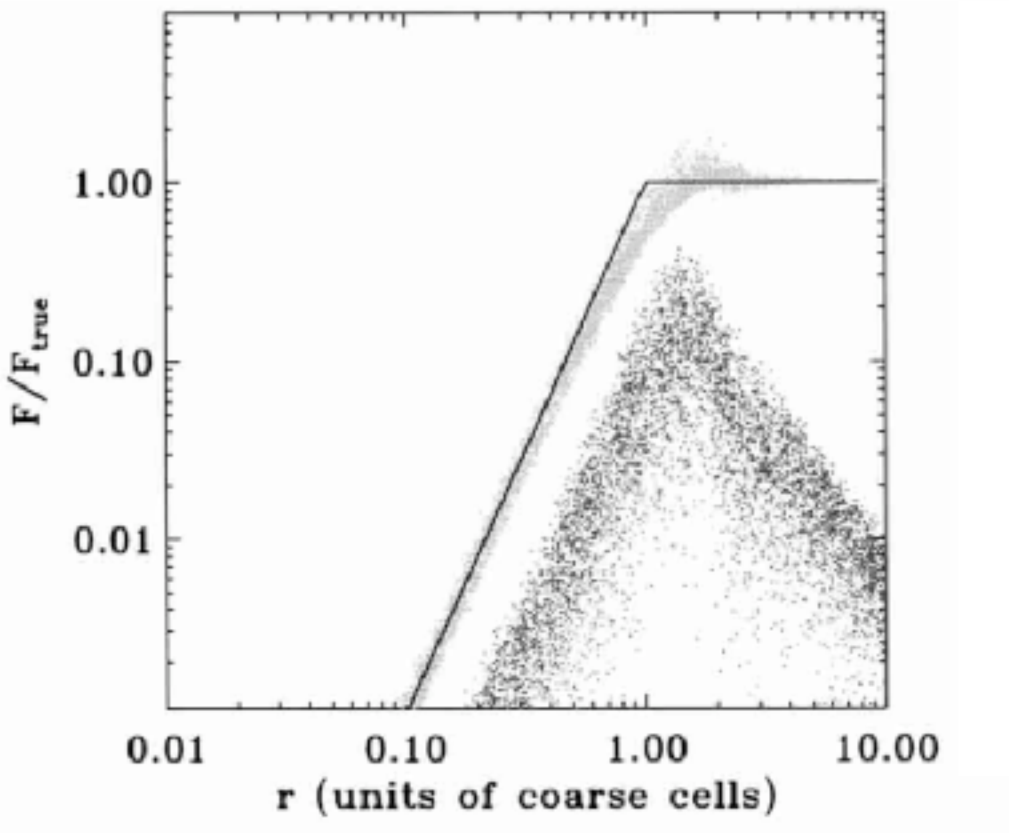
The total mass in each grid cell is then given by

$$\rho_i = \frac{1}{\Delta x^3}\sum_p m_p \int_{V_i} S(\mathbf{x_p} - \mathbf{x_i})\mathrm{d}x^3$$

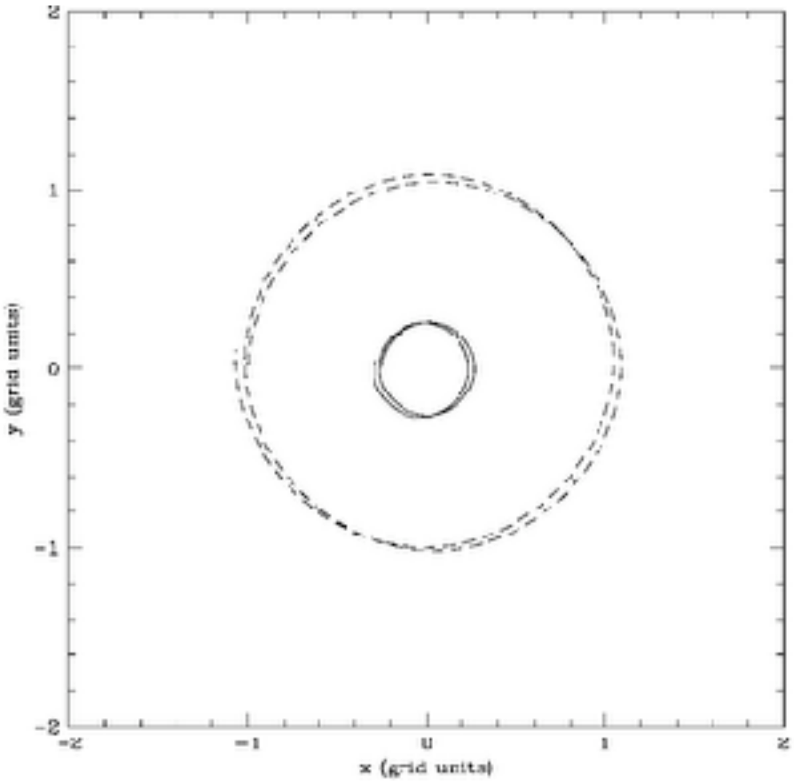


“Cloud-In-Cell” interpolation

Overall PM force accuracy



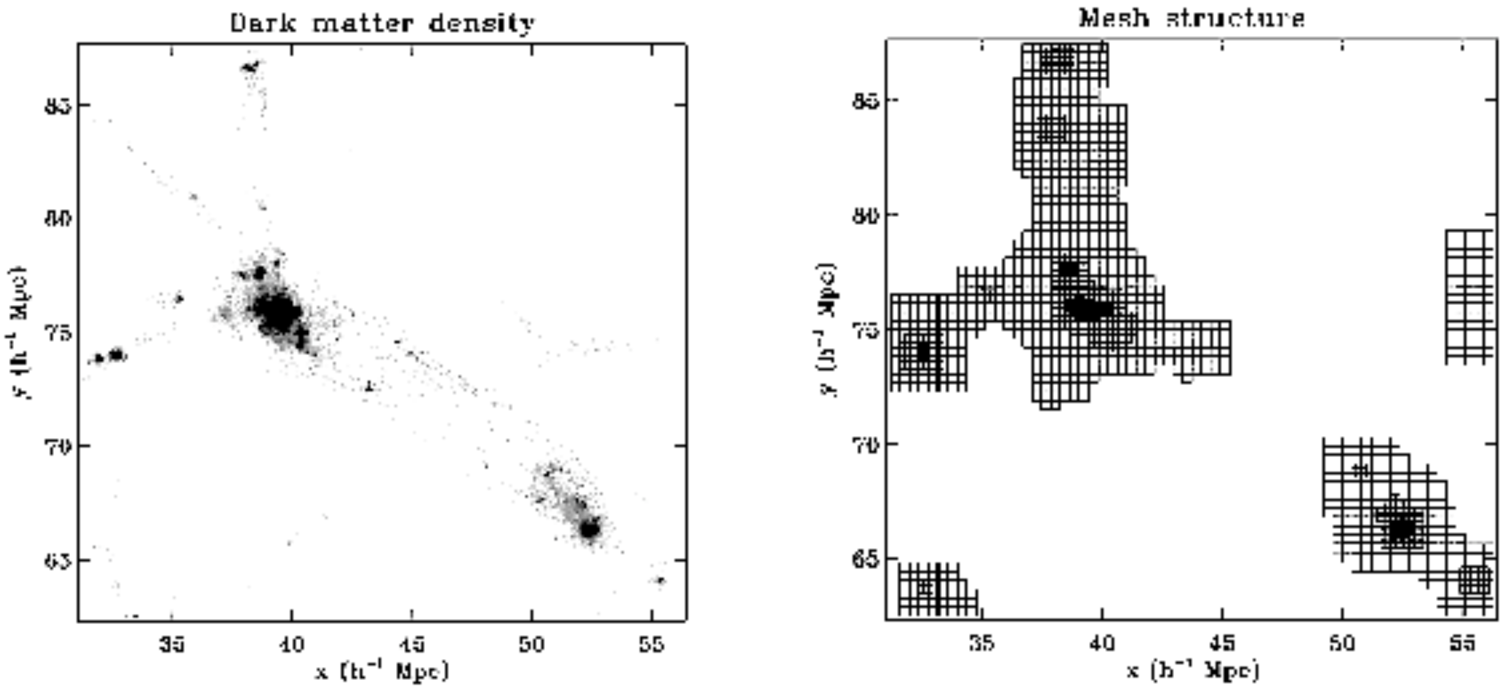
CIC, 7-point Laplacian, 2-point gradient



example of particle trajectory

PM with Adaptive Mesh Refinement

At each grid level, the force softening is equal to the local grid size. For pure dark matter simulations, using a quasi-Lagrangian strategy, the particle shot noise is kept roughly constant.



Popular codes based on this technique are **ART** (Kravtsov et al. 1997), **FLASH** (Fryxell et al. 2000), **RAMSES** (Teyssier et al. 2002), **ENZO** (Bryan et al. 2014). USE VERY DIFFERENT POISSON SOLVERS !

Poisson solvers for Adaptive Particle Mesh

Poisson solvers for adaptive grids come in two main flavors:

1. Relaxation solvers
2. Convolution solvers

Relaxation solvers use Dirichlet boundary conditions for the fine level, interpolated from the coarser level, where a coarse solution has been found before. The discrete Poisson equation is inverted using a relaxation method. Popular choices are:

1. Jacobi and Gauss-Seidel methods with successive over-relaxation
2. Conjugate gradient
3. Multigrid

Convolution methods directly convolve the mass distribution by the Green function.

1. Hierarchical FFT convolution with isolated boundary conditions
2. Tree methods and Fast Multipole Method

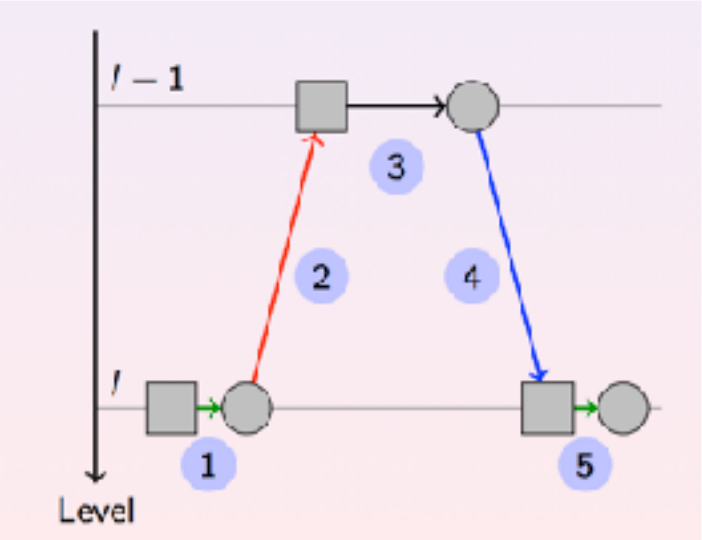
The Multigrid method

Based on the Gauss-Seidel iterative « smoothing operator »:

$$\phi_{i,j}^{n+1} = \frac{1}{4}\left(\phi_{i+1,j}^n + \phi_{i-1,j}^n + \phi_{i,j+1}^n + \phi_{i,j-1}^n\right) - \frac{1}{4}\rho_{i,j}$$

Reduce high-frequency modes in the error, low frequency converge very slowly.

One defines the residual $r_\ell^n = \Delta\phi_\ell^n - \rho_\ell$ and the error $e_\ell^n = \phi_\ell^n - \phi_\ell^{+\infty}$



1- Perform a few GS iterations (smoothing).

2- Restrict the residual to the coarse grid

$$r_\ell^n \rightarrow r_{\ell-1}^0$$

3- Solve for the coarse grid problem

$$\Delta e_{\ell-1}^{+\infty} = r_{\ell-1}^0$$

4- Prolong back the error to the fine grid:

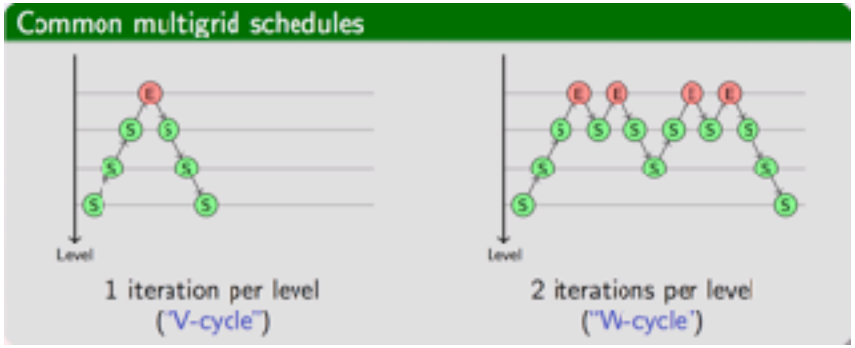
$$e_{\ell-1}^{+\infty} \rightarrow e_\ell^{n+1}$$

5- Correct the fine grid solution $\phi_\ell^{n+1} = \phi_\ell^n + e_\ell^{n+1}$ and perform a few GS iterations.

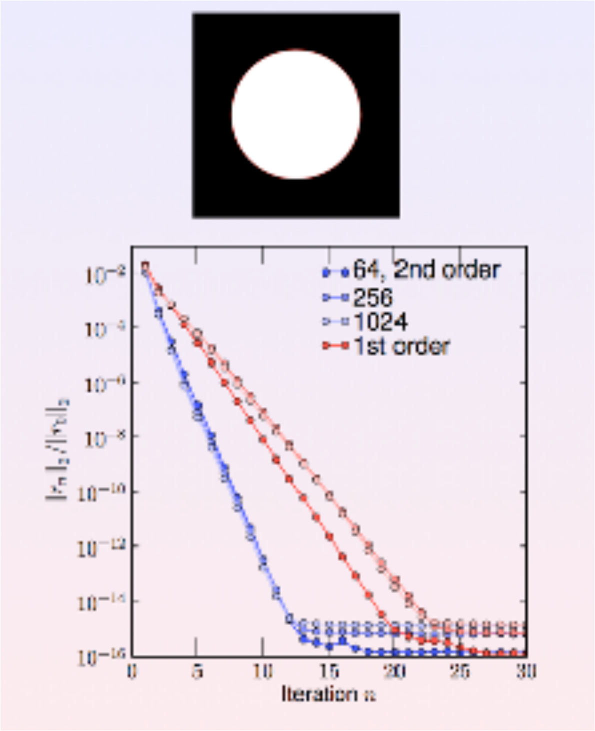
Brandt (1973), Briggs (2000)

Multigrid scheme

Recursively apply the previous 2-grid scheme. Solve for the exact solution only at the coarsest level. Iterate once or twice before going to the finer level.



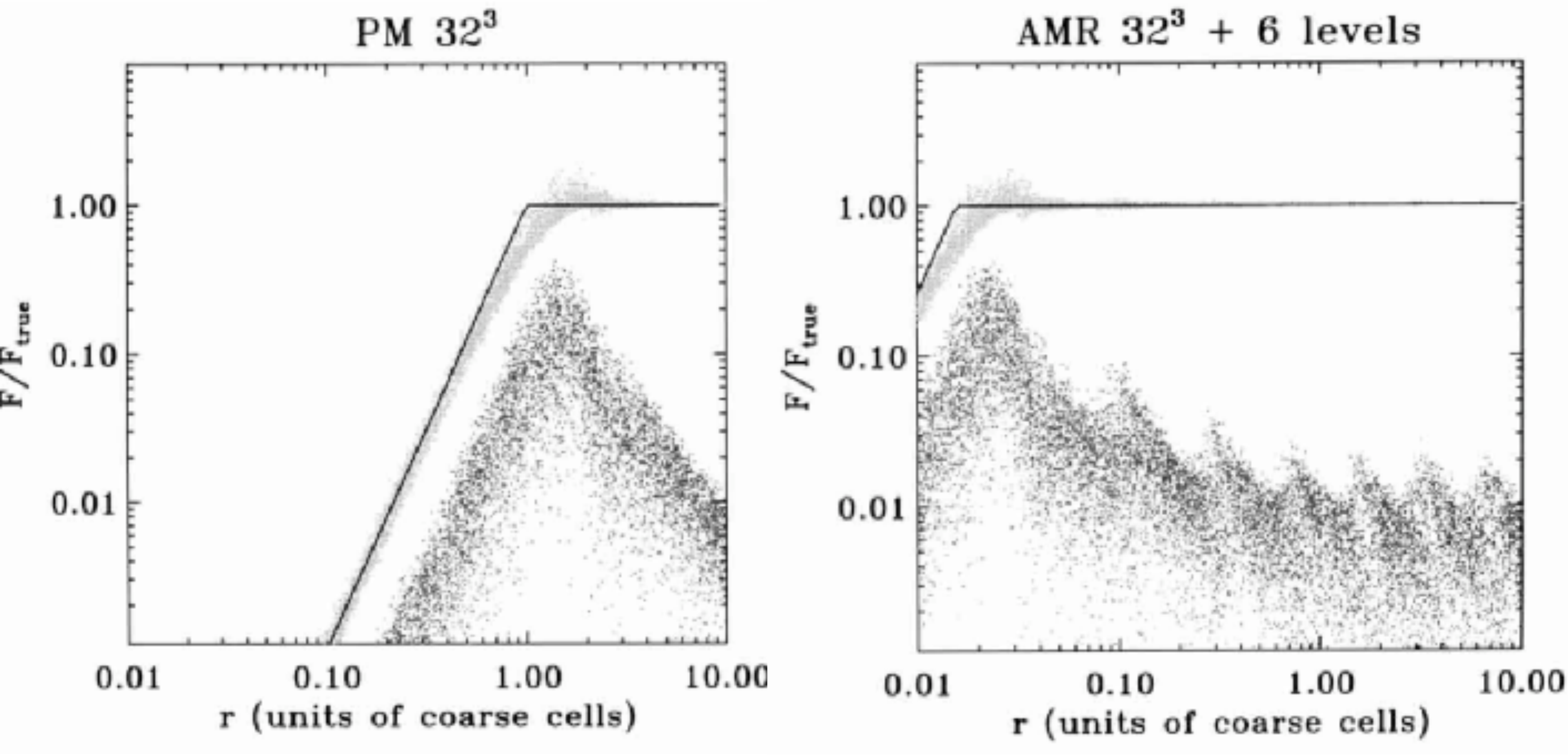
Converge in very few number of iterations, independent on the grid size. Quasi-insensitive to the quality of the initial guess.



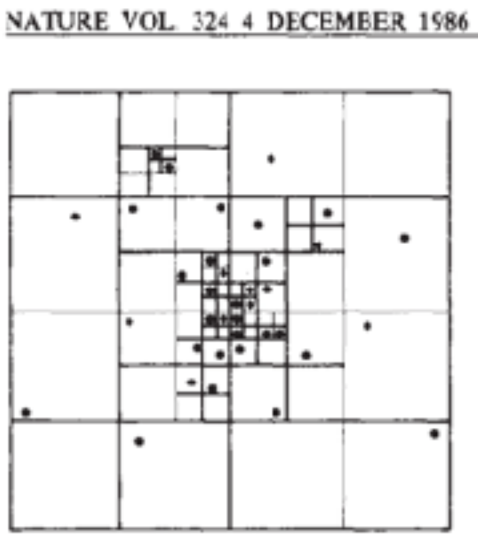
For AMR, additional care must be taken for complex level boundary conditions.

Guillet & Teyssier (2011)

Overall Adaptive PM force accuracy



Tree code and Fast Multipole Method



Barnes & Hut (1986): the Tree code

Particles are decomposed in small chunks (10 to 100) in a hierarchical octree structure of cells.

In each cell at each level, one computes the center of mass and higher order moments of the mass distribution.

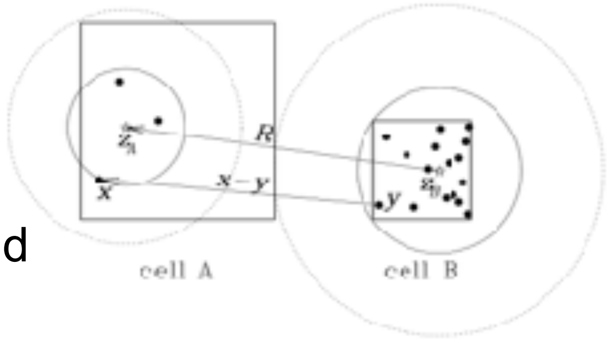
Distant particles are grouped into bigger and bigger cells, giving rise to O(N log N) scaling for gravity,.

Dehnen (2000): the Fast Multipole Method

Approximate the potential in coarse cells of the tree using a multiple expansion of the gravitational interaction.

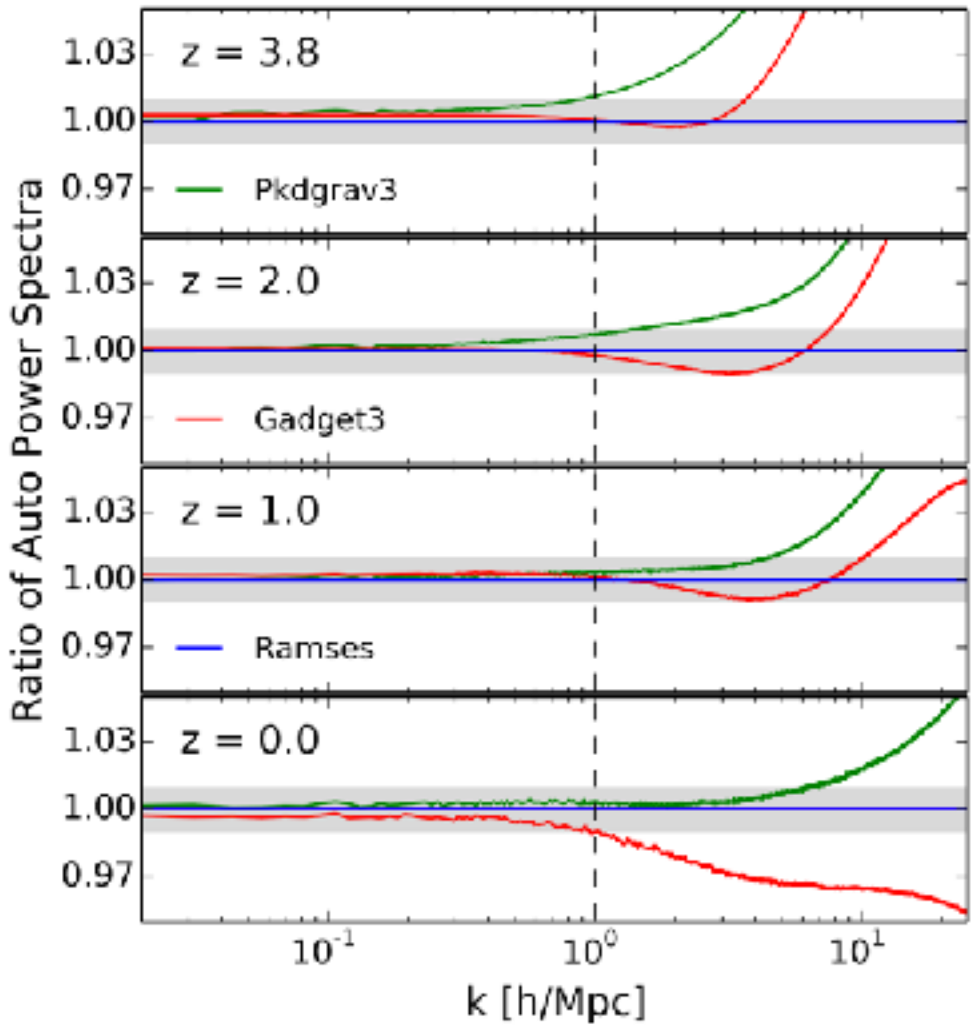
For distant cells, use directly cell-cell interactions and interpolate back to individual particles.

Exactly momentum conserving and scales as O(N).



Accuracy of large scale structure simulations

Schneider et al. 2016

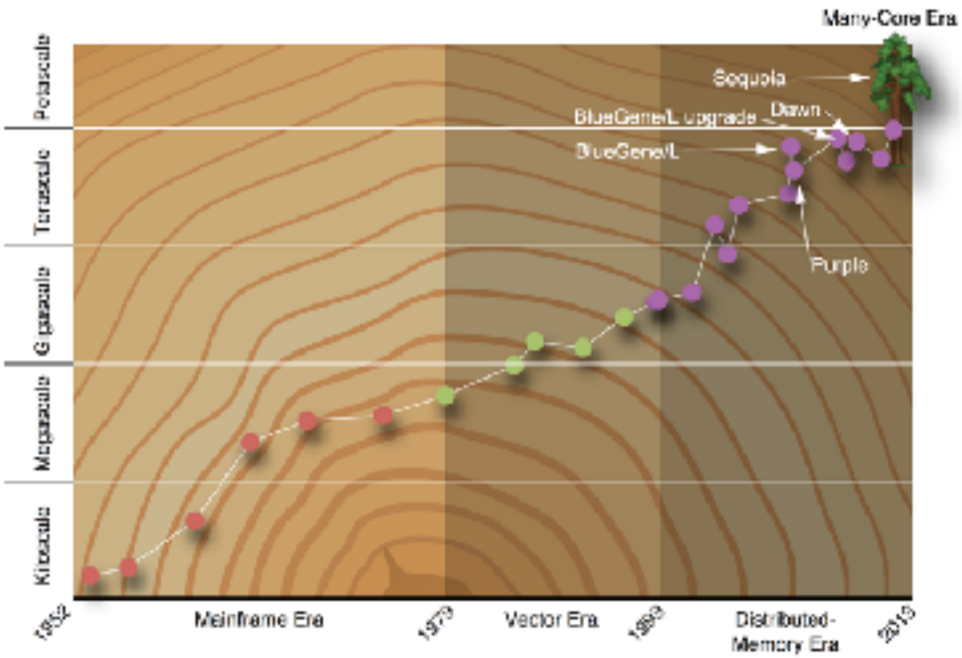


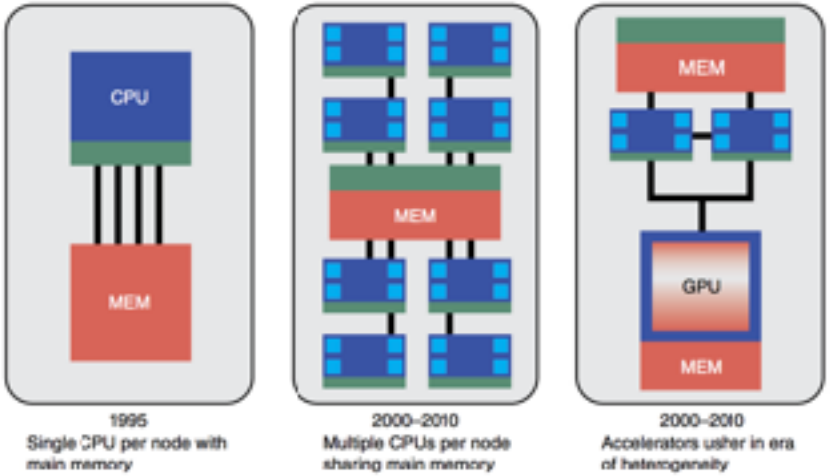
High Performance Computing

A History of High Performance Computing

Technical trends:

- Computing power is doubling every year
- Massively parallel and many-cores architectures are dominant
- Since 2010, Graphical Processing Units became a key player
- Increasing hardware complexity (hybrid system, multiple cache layers)
- Memory per core is stagnating or decreasing
- Performance per core is stagnating
- Disk Input/Output bandwidth is increasing very slowly





High Performance Computing

Evolution of programming methods:

- MPI is still the dominant programming technique
- Hybrid OpenMP/MPI approach most effective on supercomputers
- GPU programming develops quickly (CUDA and OpenACC)
- Message Passing directly within the GPU
- New specific parallel programming languages are developed: Co-array Fortran, PGAS, X10, Chapel…
- New runtime systems to handle task-based parallelism: Charm++, HPX

Consequences for users

- It is necessary to exploit a large number of (relatively slow) cores
- The memory per core is constant or even decreasing (memory limited)
- Higher level of parallelism
- I/O bottlenecks

Consequences for developers

- Raw performance of individual core not increasing anymore
- More complex architecture (cache levels, accelerators, latency…)
- Multi-disciplinary approach is required, leading to the concept of ”co-design”

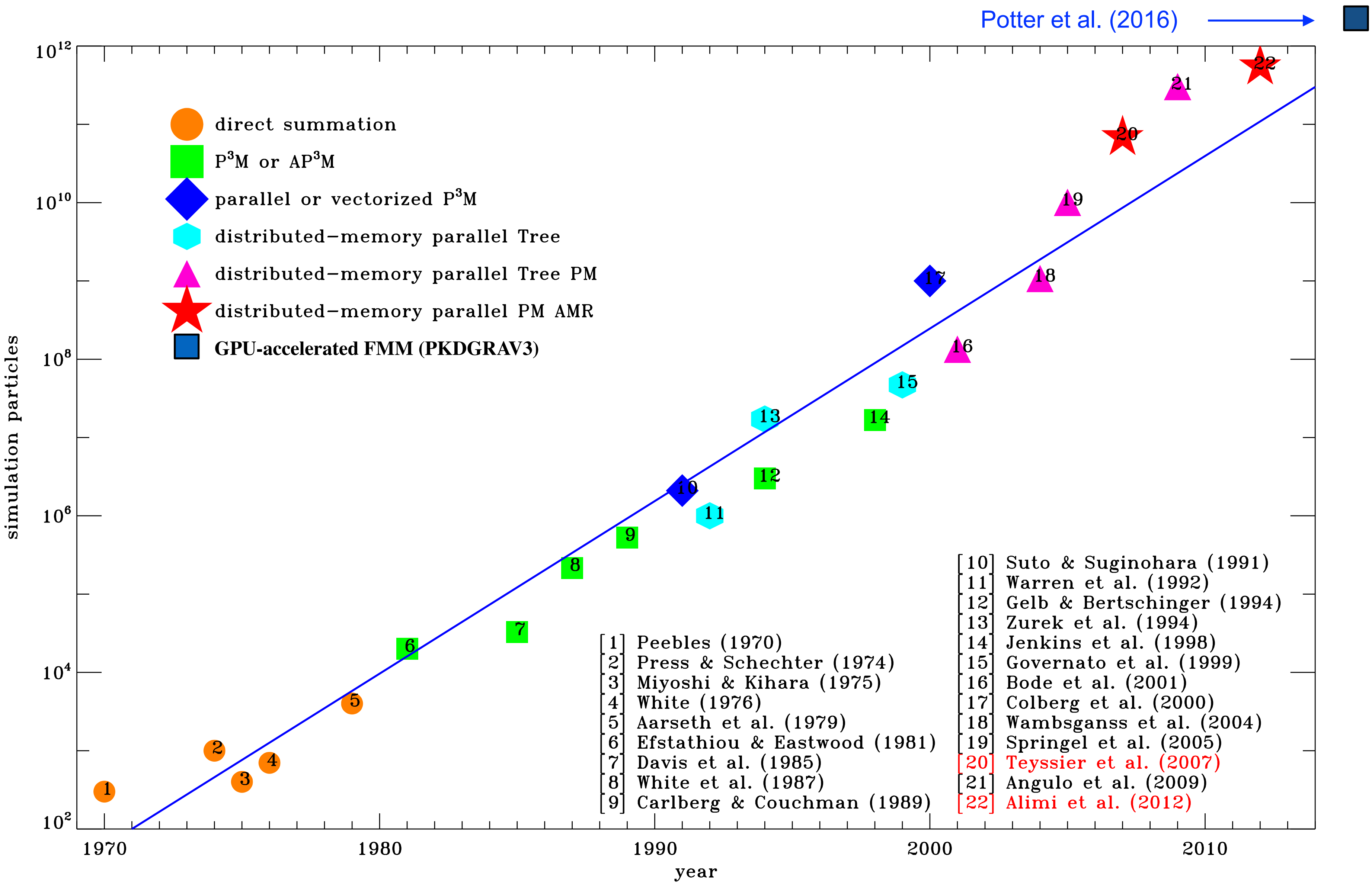
Cray XC50 with GPU

High Performance Computing

Rank	Site	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	DOE/SC/Oak Ridge National Laboratory United States	**Summit** - IBM Power System AC922, IBM POWER9 22C 3.07GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband IBM	2,282,544	122,300.0	187,659.3	8,806
2	National Supercomputing Center in Wuxi China	**Sunway TaihuLight** - Sunway MPP, Sunway SW26010 260C 1.45GHz, Sunway NRCPC	10,649,600	93,014.6	125,435.9	15,371
3	DOE/NNSA/LLNL United States	**Sierra** - IBM Power System S922LC, IBM POWER9 22C 3.1GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband IBM	1,572,480	71,610.0	119,193.6	
4	National Super Computer Center in Guangzhou China	**Tianhe-2A** - TH-IVB-FEP Cluster, Intel Xeon E5-2692v2 12C 2.2GHz, TH Express-2, Matrix-2000 NUDT	4,981,760	61,444.5	100,678.7	18,482
5	National Institute of Advanced Industrial Science and Technology (AIST) Japan	**AI Bridging Cloud Infrastructure (ABCI)** - PRIMERGY CX2550 M4, Xeon Gold 6148 20C 2.4GHz, NVIDIA Tesla V100 SXM2, Infiniband EDR Fujitsu	391,680	19,880.0	32,576.6	1,649
6	Swiss National Supercomputing Centre (CSCS) Switzerland	**Piz Daint** - Cray XC50, Xeon E5-2690v3 12C 2.6GHz, Aries interconnect , NVIDIA Tesla P100 Cray Inc.	361,760	19,590.0	25,326.3	2,272
7	DOE/SC/Oak Ridge National Laboratory United States	**Titan** - Cray XK7, Opteron 6274 16C 2.200GHz, Cray Gemini interconnect, NVIDIA K20x Cray Inc.	560,640	17,590.0	27,112.5	8,209

top500.org as of June 2018

Performance of N body codes



Hydrodynamics

The Euler equations in conservative form

Gas is a highly collisional system with a Maxwell distribution function.

A system of 3 conservation laws (mass, momentum and energy) + the EoS

$$\partial_t \rho + \nabla \cdot \mathbf{m} = 0$$

$$\partial_t \mathbf{m} + \nabla \cdot (\rho \mathbf{u} \times \mathbf{u}) + \partial_x P = 0$$

$$\partial_t E + \nabla \cdot \mathbf{u}(E + P) = 0$$

In cosmology, one need to add source terms to these conservation laws:

- gravity
- radiative processes
- star formation and feedback

Euler equations as conservation laws

General system of conservation laws with ***F*** flux vector.

$$\partial_t \mathbf{U} + \partial_x \mathbf{F}(\mathbf{U}) = 0$$

Examples:

1- Isothermal Euler equations

$$\mathbf{U} = (\rho, m)$$

$$\mathbf{F} = \left(u\rho, um + \rho a^2\right)$$

2- Euler equation

$$\mathbf{U} = (\rho, m, E)$$

$$\mathbf{F} = (u\rho, um + P, u(E + P))$$

3- Ideal MHD equations

$$\mathbf{U} = \left(\rho, m_x, m_y, m_z, E, B_x, B_y, B_z\right)$$

$$\mathbf{F} = \left(v_x\rho, v_x m_x + P_{tot} - B_x^2, v_x m_y - B_x B_y, v_x m_z - B_x B_z, \right.$$
$$\left. 0, v_x B_y - v_y B_x, v_x B_z - v_z B_x\right)$$

The 1D isothermal Euler equations

Conservative form with conservative variables $\mathbf{U} = (\rho, m)$

$$\partial_t \rho + \partial_x m = 0$$

$$\partial_t m + \partial_x \left(\rho u^2 + \rho a^2 \right) = 0$$

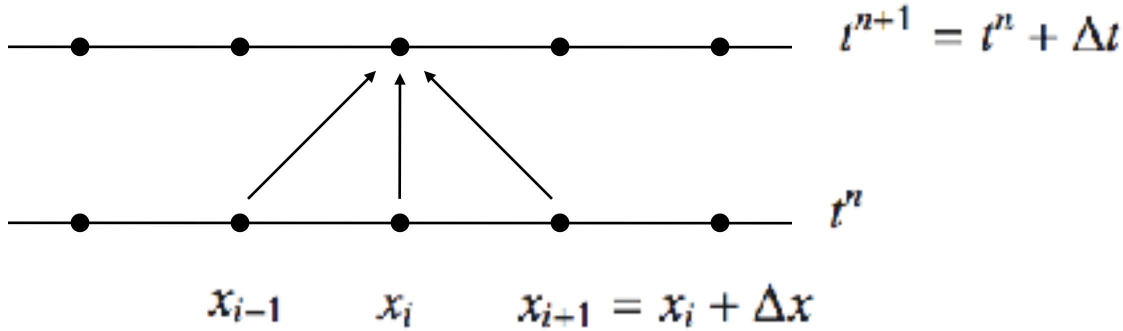
Primitive form with primitive variables $\mathbf{W} = (\rho, u)$

$$\partial_t \rho + u \partial_x \rho + \rho \partial_x u = 0$$

$$\partial_t u + u \partial_x u + \frac{a^2}{\rho} \partial_x \rho = 0$$

a is the isothermal sound speed

Finite difference scheme for the advection equation



$$u_i^n = u(x_i, t^n) \qquad \partial_x u \simeq \frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x} \qquad \partial_t u \simeq \frac{u_i^{n+1} - u_i^n}{\Delta t}$$

Finite difference approximation of the advection equation

$$\partial_t u + a\partial_x u = 0 \quad \Longrightarrow \quad \frac{u_i^{n+1} - u_i^n}{\Delta t} + a\frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x} = 0$$

The modified equation

$$\frac{u_i^{n+1}-u_i^n}{\Delta t}+a\frac{u_{i+1}^n-u_{i-1}^n}{2\Delta x}=0$$

Taylor expansion in time up to second order

$$u_i^{n+1}=u_i^n+\Delta t\left(\frac{\partial u}{\partial t}\right)+\frac{(\Delta t)^2}{2}\left(\frac{\partial^2 u}{\partial t^2}\right)$$

Taylor expansion in space up to second order

$$u_{i+1}^n=u_i^n+\Delta x\left(\frac{\partial u}{\partial x}\right)+\frac{(\Delta x)^2}{2}\left(\frac{\partial^2 u}{\partial x^2}\right)$$

$$u_{i-1}^n=u_i^n-\Delta x\left(\frac{\partial u}{\partial x}\right)+\frac{(\Delta x)^2}{2}\left(\frac{\partial^2 u}{\partial x^2}\right)$$

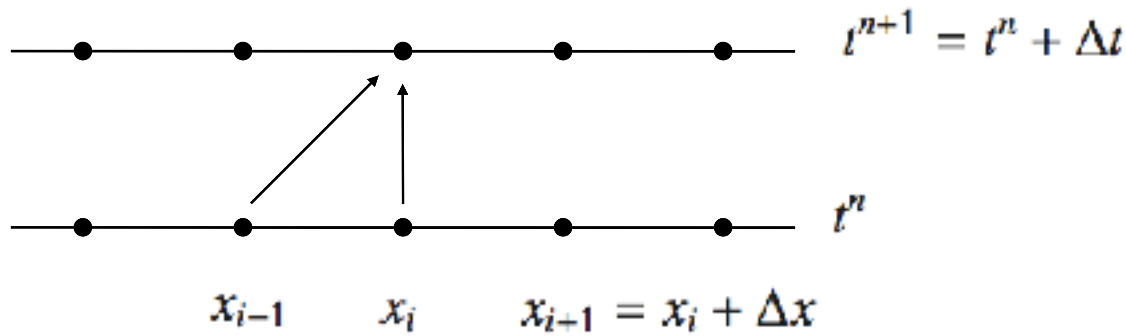
The advection equation becomes the advection-diffusion equation

$$\left(\frac{\partial u}{\partial t}\right)+a\left(\frac{\partial u}{\partial x}\right)=-\frac{\Delta t}{2}\left(\frac{\partial^2 u}{\partial t^2}\right)+O(\Delta t^2,\Delta x^2)$$

$$\left(\frac{\partial u}{\partial t}\right)+a\left(\frac{\partial u}{\partial x}\right)=-a^2\frac{\Delta t}{2}\left(\frac{\partial^2 u}{\partial x^2}\right)+O(\Delta t^2,\Delta x^2)$$

Negative diffusion coefficient: the scheme is *unconditionally unstable*

Upwind scheme for the advection equation



a>0: use only upwind values, discard downwind variables

$$\partial_x u \simeq \frac{u_i^n - u_{i-1}^n}{\Delta x} \quad \Longrightarrow \quad \frac{u_i^{n+1} - u_i^n}{\Delta t} + a\frac{u_i^n - u_{i-1}^n}{\Delta x} = 0$$

Taylor expansion up to second order:

$$\left(\frac{\partial u}{\partial t}\right) + a\left(\frac{\partial u}{\partial x}\right) = -\frac{\Delta t}{2}\left(\frac{\partial^2 u}{\partial t^2}\right) + a\frac{\Delta x}{2}\left(\frac{\partial^2 u}{\partial x^2}\right) + O(\Delta t^2, \Delta x^2)$$

Upwind scheme is stable if C<1, with $C = a\frac{\Delta t}{\Delta x}$

$$\left(\frac{\partial u}{\partial t}\right) + a\left(\frac{\partial u}{\partial x}\right) = a\frac{\Delta x}{2}(1 - C)\left(\frac{\partial^2 u}{\partial x^2}\right) + O(\Delta t^2, \Delta x^2)$$

Advection-diffusion type modified equation

Finite difference approximation of the advection equation:

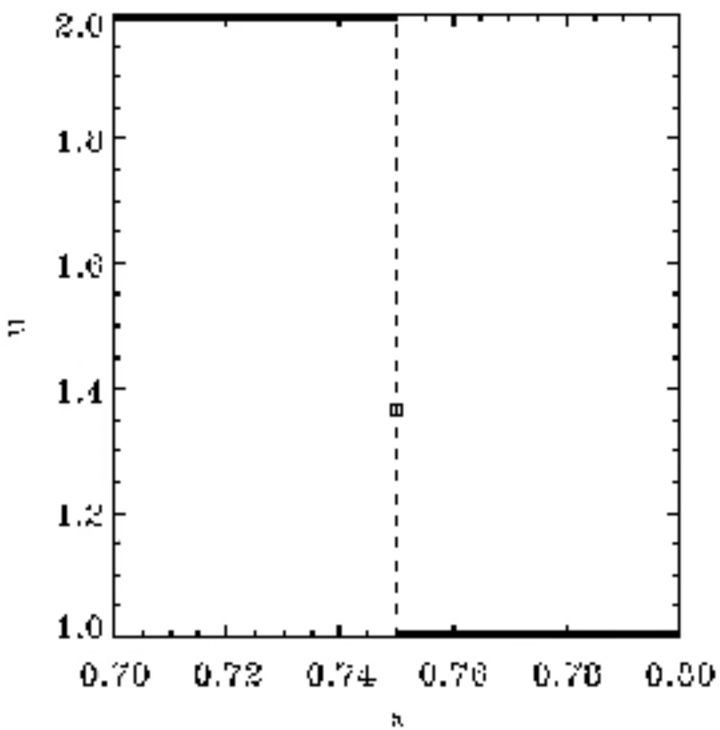
$$\left(\frac{\partial u}{\partial t}\right) + a\left(\frac{\partial u}{\partial x}\right) = \eta\left(\frac{\partial^2 u}{\partial x^2}\right)$$

Central differencing unstable: $\eta < 0$

Upwind differencing is stable: $\eta > 0$ $\quad \eta = a\frac{\Delta x}{2}(1 - C)$

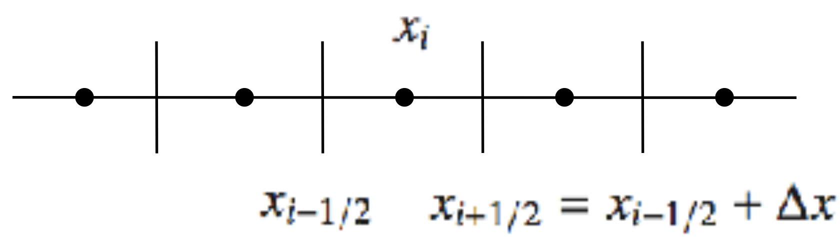
Smearing of initial discontinuity:

“numerical diffusion”



Thickness increases as $\sqrt{\eta t}$

Finite volume scheme for the advection equation



Finite volume approximation of the advection equation:

$$u_i^n = \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} u(x, t^n) \mathrm{d}x$$

Use integral form of the conservation law:

$$\int_{x_{i-1/2}}^{x_{i+1/2}} \int_{t^n}^{t^{n+1}} \mathrm{d}x \mathrm{d}t \, (\partial_t u + a \partial_x u) = 0$$

Exact evolution of volume averaged quantities:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} + a \frac{u_{i+1/2}^{n+1/2} - u_{i-1/2}^{n+1/2}}{\Delta x} = 0$$

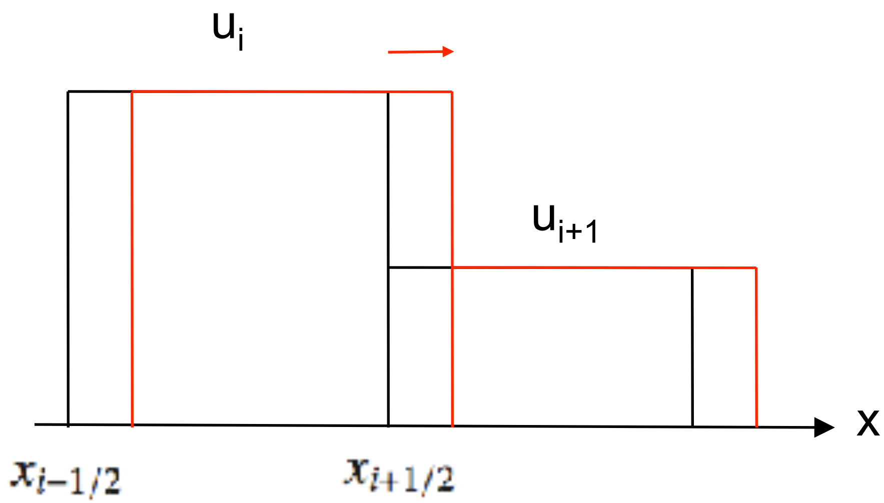
Time averaged flux function: $$u_{i+1/2}^{n+1/2} = \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} u(x_{i+1/2}, t) \mathrm{d}t$$

Godunov scheme for the advection equation

The time averaged flux function: $$u_{i+1/2}^{n+1/2} = \frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} u(x_{i+1/2}, t)\mathrm{dt}$$

is computed using the exact solution of the problem defined

at cell interfaces with piecewise constant initial data: the Riemann problem



For all t>0: $$u(x_{i+1/2}, t) = u_i^n \quad \text{if} \quad a > 0$$
$$u(x_{i+1/2}, t) = u_{i+1}^n \quad \text{if} \quad a < 0$$

The Godunov scheme for the advection equation is identical to the upwind finite difference scheme.

The isothermal wave equation

We linearise the isothermal Euler equation around some equilibrium state.

$$\mathbf{W} = \mathbf{W}_0 + \Delta\mathbf{W}$$

Using the system in primitive form, we get the ***linear*** system:

$$\partial_t \Delta\mathbf{W} + \mathbf{A}_0 \partial_x \Delta\mathbf{W} = 0$$

where the constant matrix has 2 real eigenvalues and 2 eigenvectors

$$\mathbf{A}_0 = \left\{ \begin{array}{cc} u & \rho \\ \frac{a^2}{\rho} & u \end{array} \right\} \qquad \lambda^+ = u + a \qquad \Delta\alpha^+ = \frac{1}{2}\left(\Delta\rho + \rho\frac{\Delta u}{a}\right)$$

$$\lambda^- = u - a \qquad \Delta\alpha^- = \frac{1}{2}\left(\Delta\rho - \rho\frac{\Delta u}{a}\right)$$

The previous system is equivalent to 2 independent *scalar linear* PDEs.

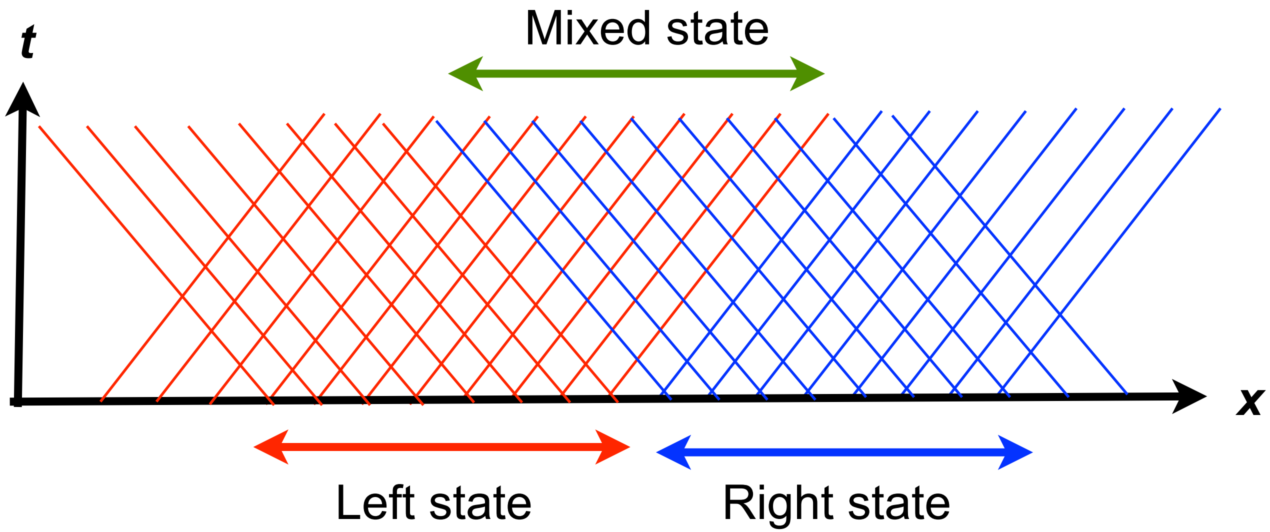
$$\partial_t \Delta\alpha^+ + (u+a)\partial_x \Delta\alpha^+ = 0$$

$$\partial_t \Delta\alpha^- + (u-a)\partial_x \Delta\alpha^- = 0$$

$\Delta\alpha^+$ ($\Delta\alpha^-$) is a Riemann invariant along characteristic curves moving with velocity $u + a$ ($u - a$)

Riemann problem for isothermal waves

Initial conditions are defined by 2 semi-infinite regions with piecewise constant initial states $(\Delta\rho_R, \Delta u_R)$ and $(\Delta\rho_L, \Delta u_L)$



"Star" state is obtained using the 2 Riemann invariants.

$$u - a < \frac{x}{t} < u + a$$

$$\Delta\rho^* = \Delta\alpha_L^+ + \Delta\alpha_R^-$$

$$\Delta u^* = \frac{a}{\rho}(\Delta\alpha_L^+ - \Delta\alpha_R^-)$$

Godunov scheme for the isothermal wave equation

We now explain the Godunov scheme for the density only.

$$\frac{\Delta\rho_i^{n+1} - \Delta\rho_i^n}{\Delta t} + \frac{F_{i+1/2}^{n+1/2} - F_{i-1/2}^{n+1/2}}{\Delta x} = 0$$

$$F_{i+1/2}^{n+1/2} = u\ \Delta\rho_{i+1/2}^{n+1/2} + \rho\ \Delta u_{i+1/2}^{n+1/2}$$

Using the Riemann problem defined by $(L, R) = (i, i+1)$ we get

$$\Delta\rho_{i+1/2}^{n+1/2} = \Delta\alpha_i^+ + \Delta\alpha_{i+1}^- = \frac{\Delta\rho_i + \Delta\rho_{i+1}}{2} - \frac{\rho}{2a}(\Delta u_{i+1} - \Delta u_i)$$

$$\Delta u_{i+1/2}^{n+1/2} = \frac{a}{\rho}\left(\Delta\alpha_i^+ - \Delta\alpha_{i+1}^-\right) = \frac{\Delta u_i + \Delta u_{i+1}}{2} - \frac{a}{2\rho}(\Delta\rho_{i+1} - \Delta\rho_i)$$

The final flux function is given by the explicit linear function:

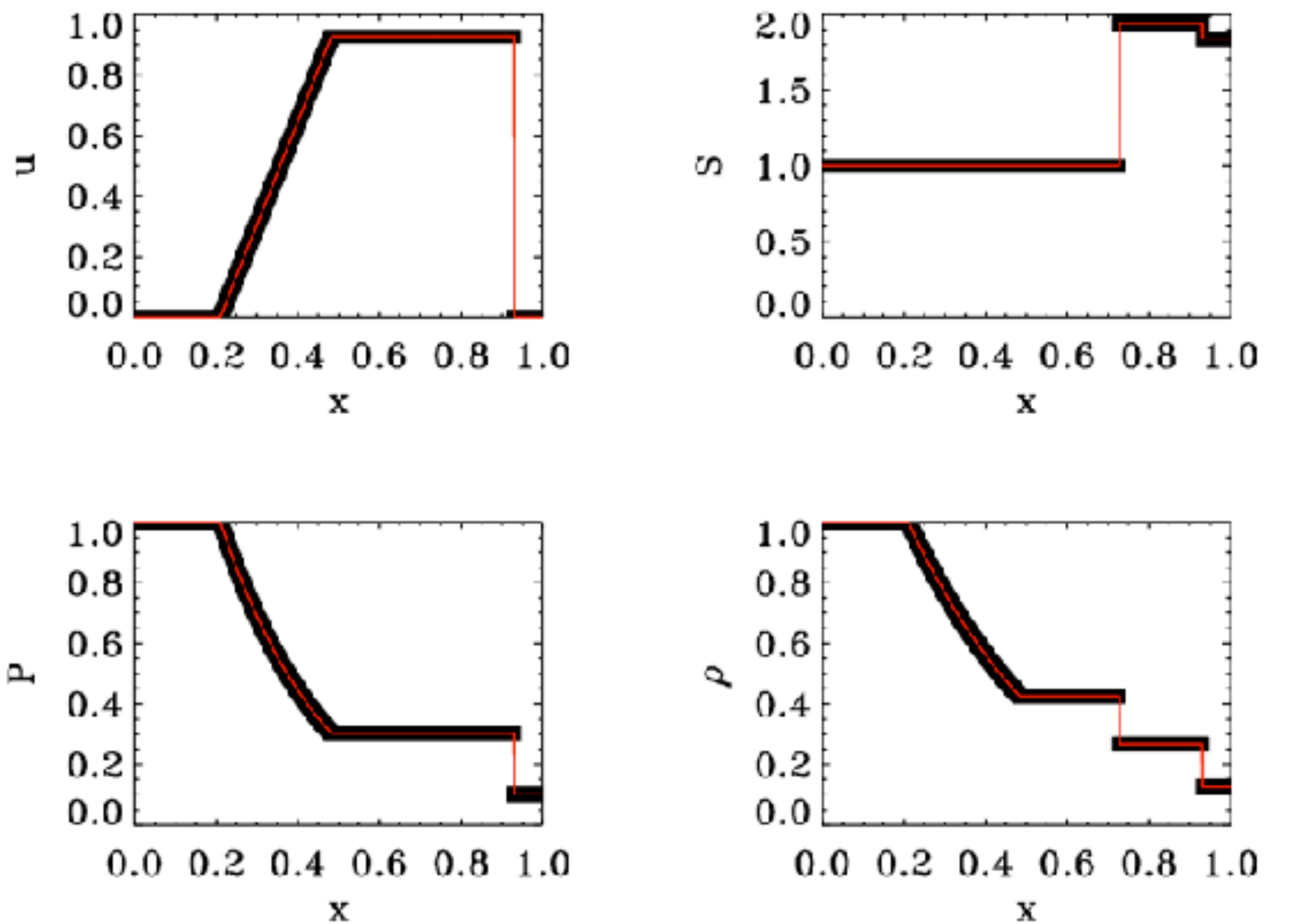
$$F_{i+1/2}^{n+1/2} = \underbrace{u\ \frac{\Delta\rho_i + \Delta\rho_{i+1}}{2} + \rho\ \frac{\Delta u_i + \Delta u_{i+1}}{2}}_{\text{Unstable centered FD scheme}} - \underbrace{\frac{a\Delta x}{2}\frac{\partial\rho}{\partial x} - \frac{\rho}{a}\frac{u\Delta x}{2}\frac{\partial u}{\partial x}}_{\text{Diffusive term}}$$

Formally, we have $F = F_{\rm true} - \nu\nabla\rho$ which results in the modified equation

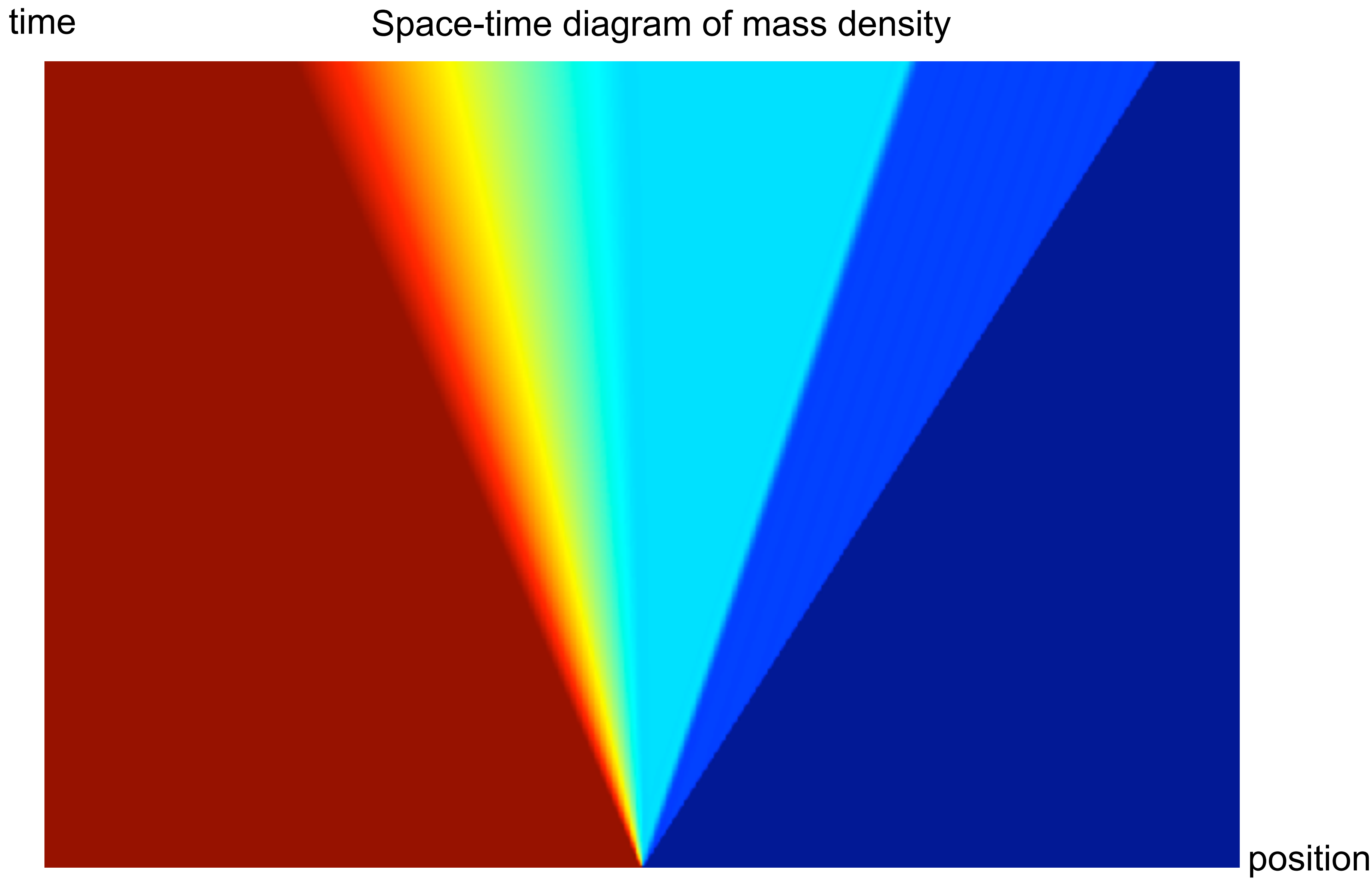
$$\frac{\partial\rho}{\partial t} + \nabla\cdot F_{\rm true} = \nu\Delta\rho \quad \text{with} \quad \nu = \frac{a\Delta x}{2}$$

Riemann problem for the Euler equations

The Sod shock tube test at t=0.245



Riemann problem for the Euler equations



Godunov scheme for hyperbolic systems

The system of conservation laws

$$\partial_t \mathbf{U} + \partial_x \mathbf{F} = 0$$

is discretised using the following exact integral form:

$$\frac{\mathbf{U}_i^{n+1} - \mathbf{U}_i^n}{\Delta t} + \frac{\mathbf{F}_{i+1/2}^{n+1/2} - \mathbf{F}_{i-1/2}^{n+1/2}}{\Delta x} = 0$$

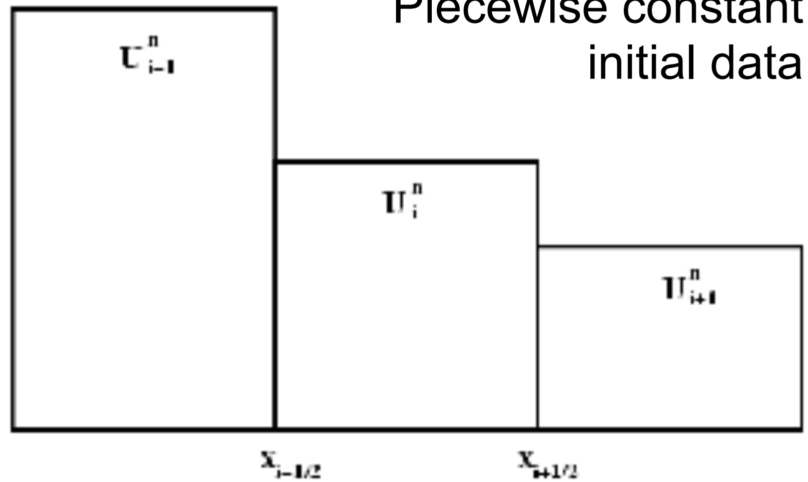
The time average flux function is computed using the self-similar solution of the inter-cell Riemann problem:

$$\mathbf{U}_{i+1/2}^*(x/t) = \mathcal{RP}\left[\mathbf{U}_i^n, \mathbf{U}_{i+1}^n\right]$$

$$\mathbf{F}_{i+1/2}^{n+1/2} = \mathbf{F}(\mathbf{U}_{i+1/2}^*(0))$$

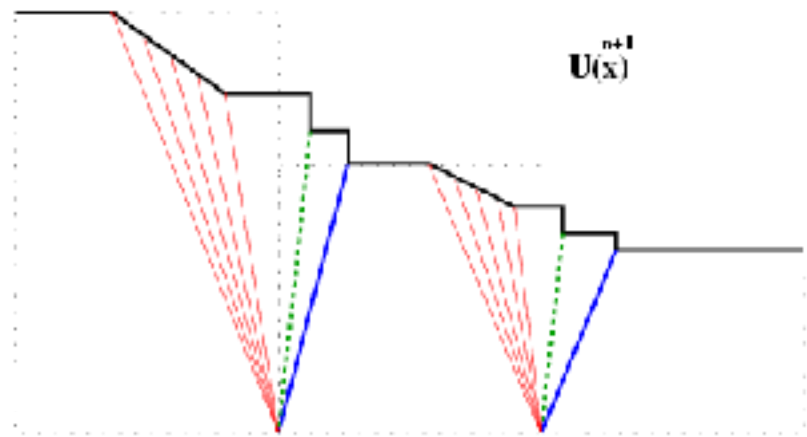
This defines the Godunov flux:

$$\mathbf{F}_{i+1/2}^{n+1/2} = \mathbf{F}^*(\mathbf{U}_i^n, \mathbf{U}_{i+1}^n)$$



Piecewise constant initial data

- Godunov, S. K. (1959), A Difference Scheme for Numerical Solution of Discontinuos Solution of Hydrodynamic Equations, *Math. Sbornik, 47, 271-306*, translated US Joint Publ. Res. Service, JPRS 7226, 1969.

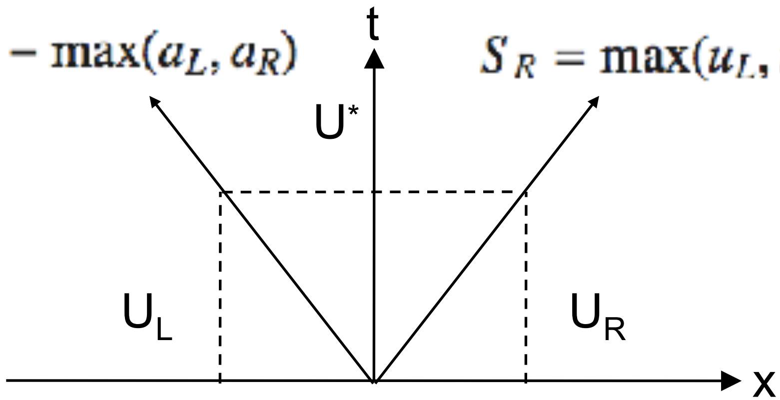


Advection: 1 wave, Euler: 3 waves, MHD: 7 waves

The HLL Riemann solver

Approximate the true Riemann fan by 2 waves and 1 intermediate state:

$$S_L = \min(u_L, u_R) - \max(a_L, a_R) \qquad S_R = \max(u_L, u_R) + \max(a_L, a_R)$$



Compute U* using the integral form between $S_L t$ and $S_R t$

$$\mathbf{U}^*(\mathbf{U}_L, \mathbf{U}_R) = \frac{S_R \mathbf{U}_R - S_L \mathbf{U}_L - (\mathbf{F}_R - \mathbf{F}_L)}{S_R - S_L}$$

Compute F* using the integral form between $S_L t$ and 0.

$$S_L > 0 \quad \mathbf{F}^*(U_L, U_R) = \mathbf{F}_L$$

$$S_R < 0 \quad \mathbf{F}^*(U_L, U_R) = \mathbf{F}_R$$

$$S_L < 0 \ \text{ and } \ S_R > 0 \quad \mathbf{F}^*(U_L, U_R) = \frac{S_R \mathbf{F}_L - S_L \mathbf{F}_R + S_L S_R (\mathbf{U}_R - \mathbf{U}_L)}{S_R - S_L}$$

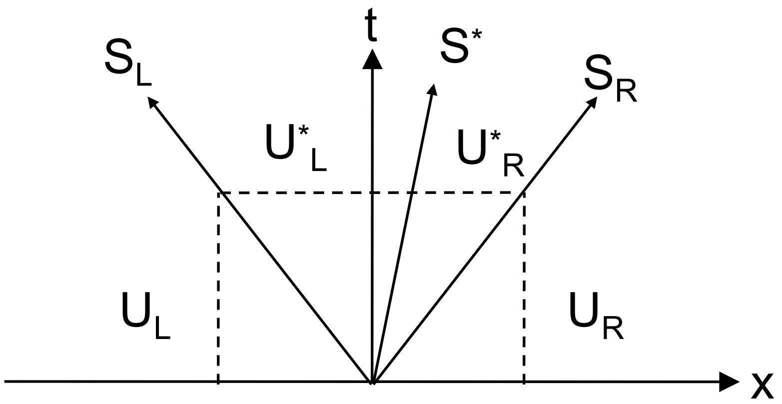
Other HLL-type Riemann solvers

Lax-Friedrich Riemann solver: $S_* = S_R = -S_L = \max(|u_L| + a_L, |u_R| + a_R)$

$$\mathbf{F}^*(U_L, U_R) = \frac{\mathbf{F}_L + \mathbf{F}_R}{2} - S_* \frac{\mathbf{U}_R - \mathbf{U}_L}{2}$$

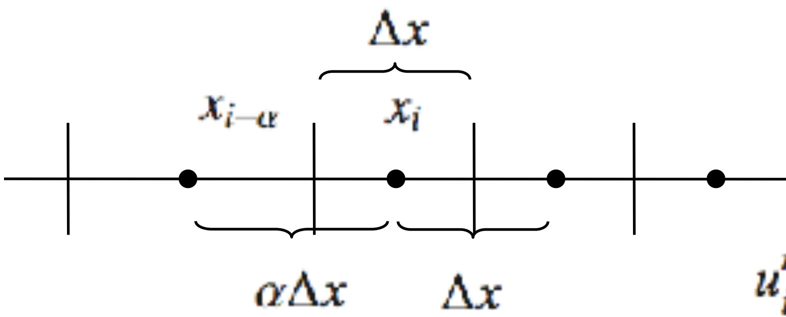
HLLC Riemann solver: add a third wave for the contact (entropy) wave.

$$S_* = \frac{(\rho u)^*_{HLL}}{\rho^*_{HLL}}$$



See Toro (1997) for details.

Numerical diffusion with non-uniform meshes



Assume a and C>0.

$$u_{i+1/2}^{n+1/2} = u_i^n + (1-C)\frac{\Delta x}{2}\left(\frac{\partial u}{\partial x}\right)_i$$

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} + a\frac{u_{i+1/2}^{n+1/2} - u_{i-1/2}^{n+1/2}}{\Delta x} = 0$$

$$u_{i-1/2}^{n+1/2} = u_{i-\alpha}^n + (2\alpha - 1 - C)\frac{\Delta x}{2}\left(\frac{\partial u}{\partial x}\right)_{i-1}$$

First order scheme: $$\left(\frac{\partial u}{\partial t}\right) + \alpha a\left(\frac{\partial u}{\partial x}\right) = a\frac{\Delta x}{2}(\alpha^2 - C)\left(\frac{\partial^2 u}{\partial x^2}\right) + O(\Delta t^2, \Delta x^2)$$

Second order scheme: $$\left(\frac{\partial u}{\partial t}\right) + a\left(\frac{\partial u}{\partial x}\right) = a\frac{\Delta x}{2}(\alpha - C)(1-\alpha)\left(\frac{\partial^2 u}{\partial x^2}\right) + O(\Delta t^2, \Delta x^2)$$

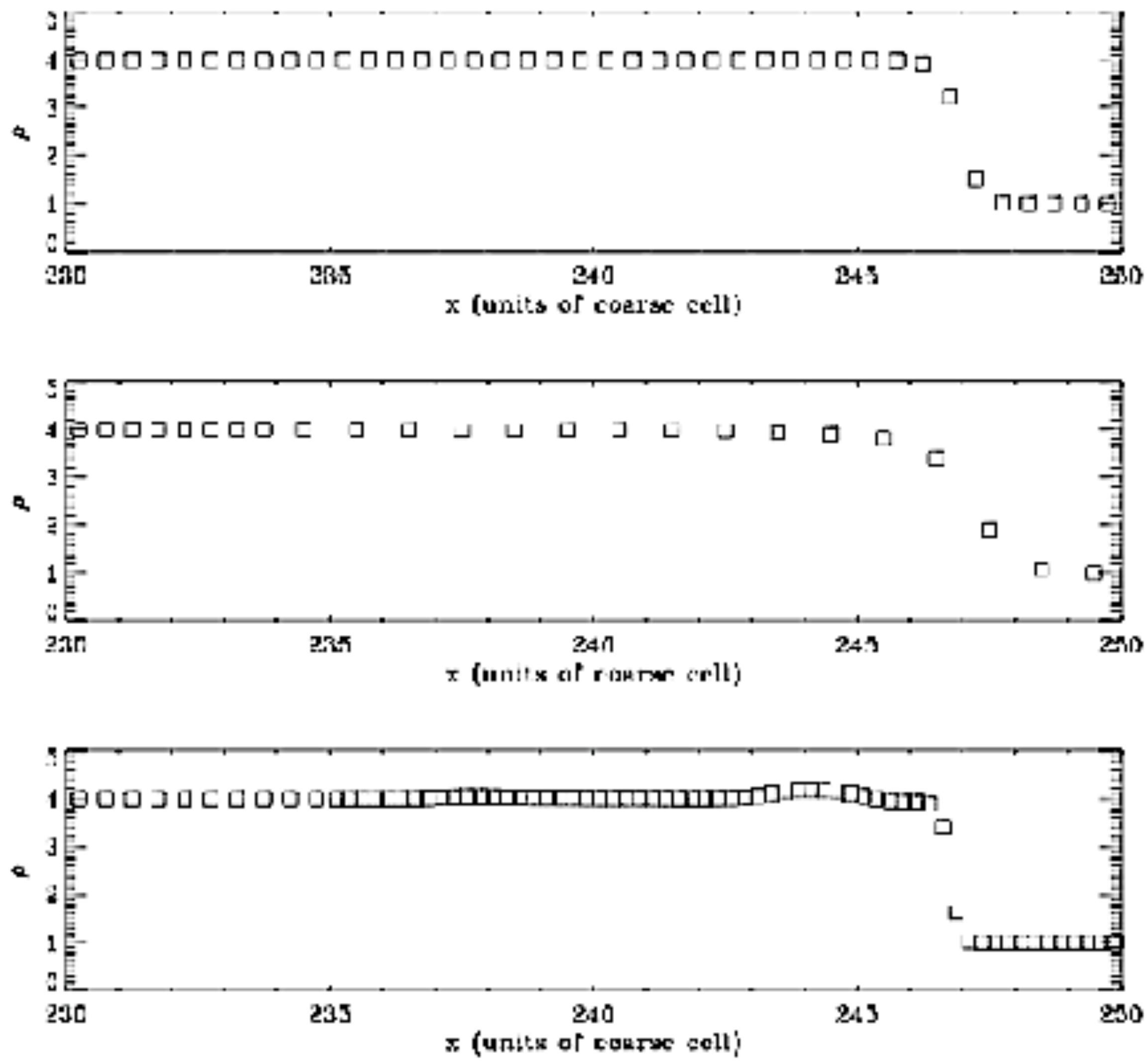
In conclusion, we loose one order of accuracy in the modified equation.

The first order scheme is *not consistent* at level boundary (wrong wave speed).

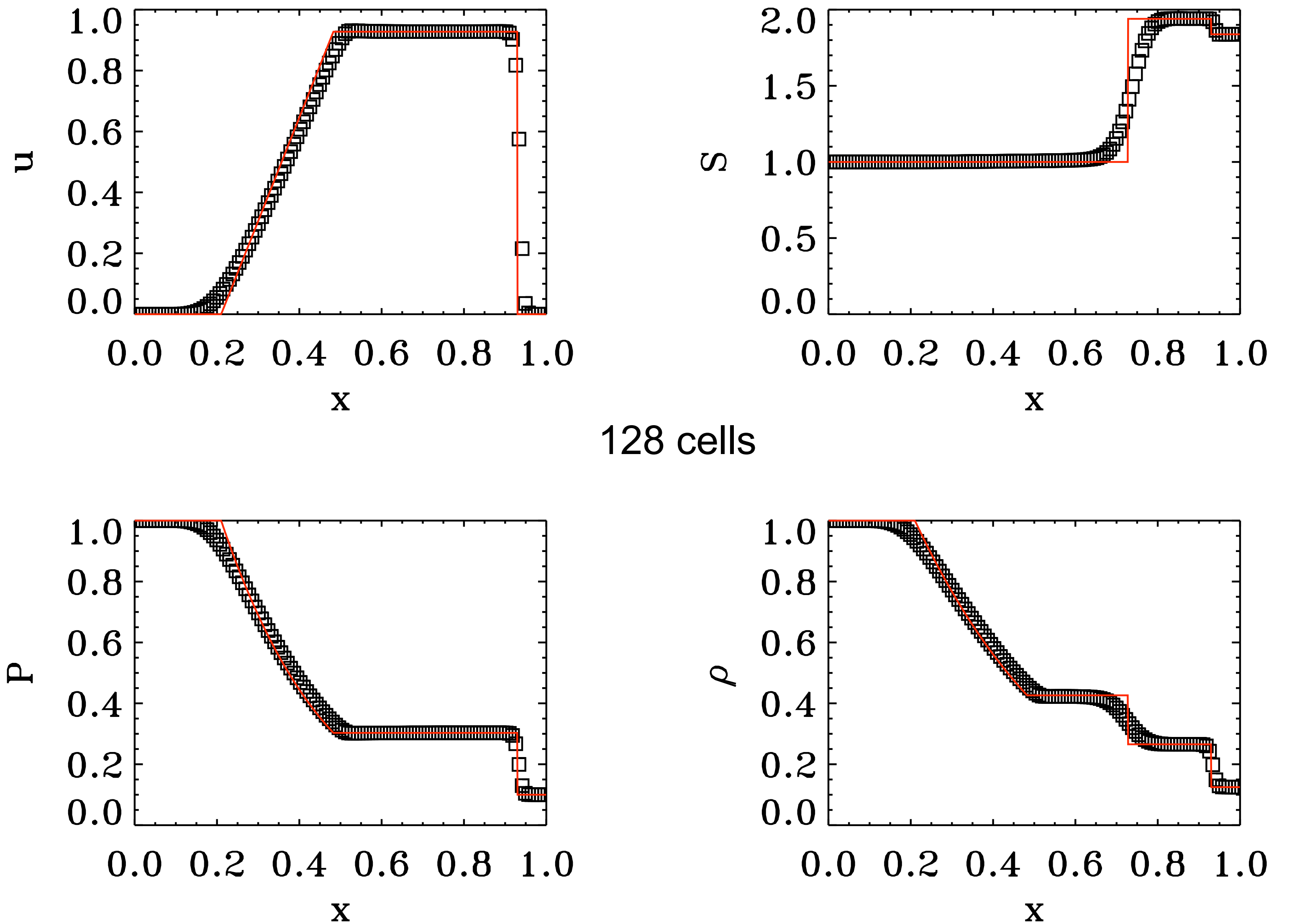
The second order scheme (for α=1.5) is *unstable* at level boundary.

Solutions: 1- refine gradients, 2- add artificial diffusion, 3- go to higher order

Shock wave propagating through level boundary

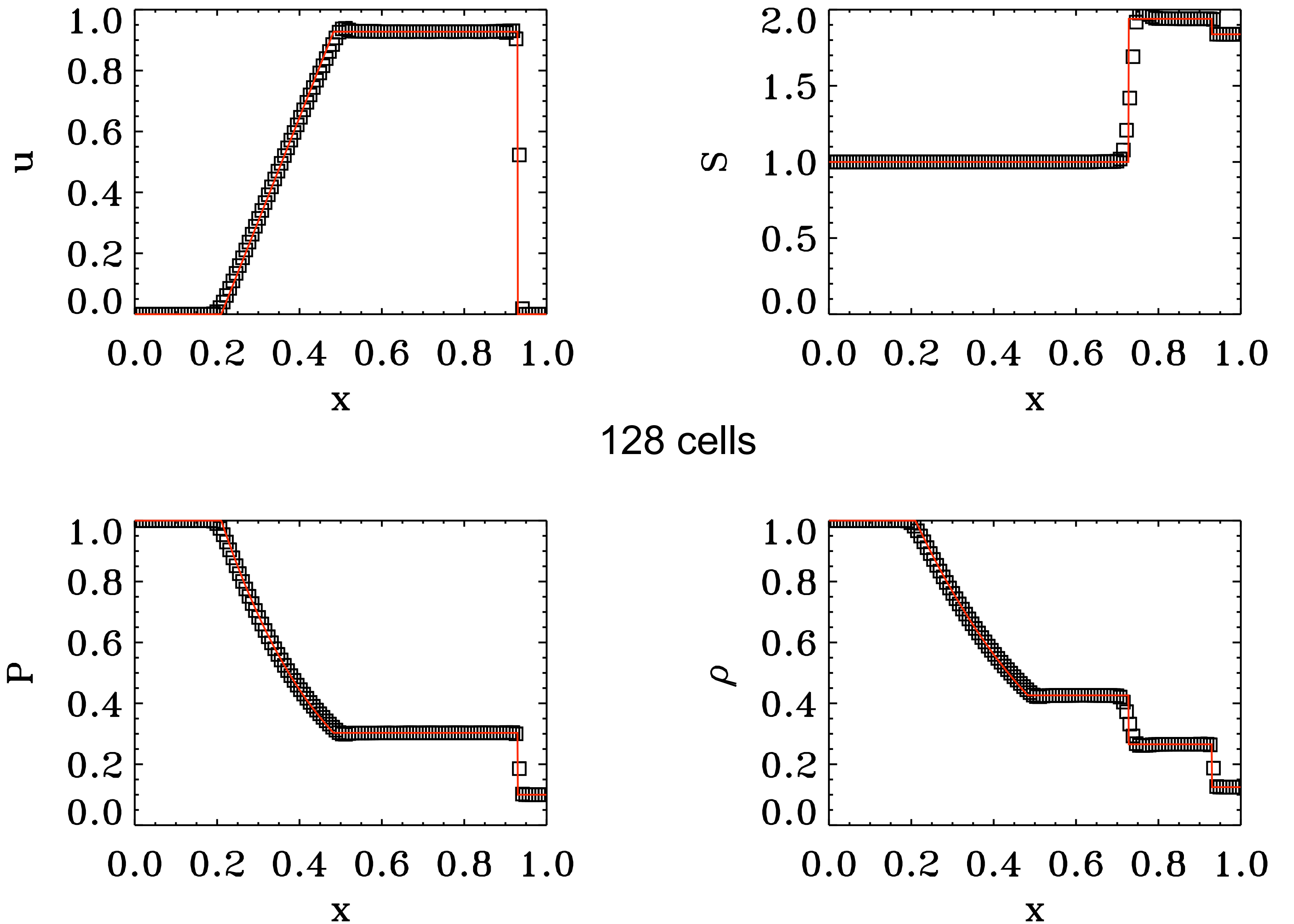


Sod test with first order Godunov scheme

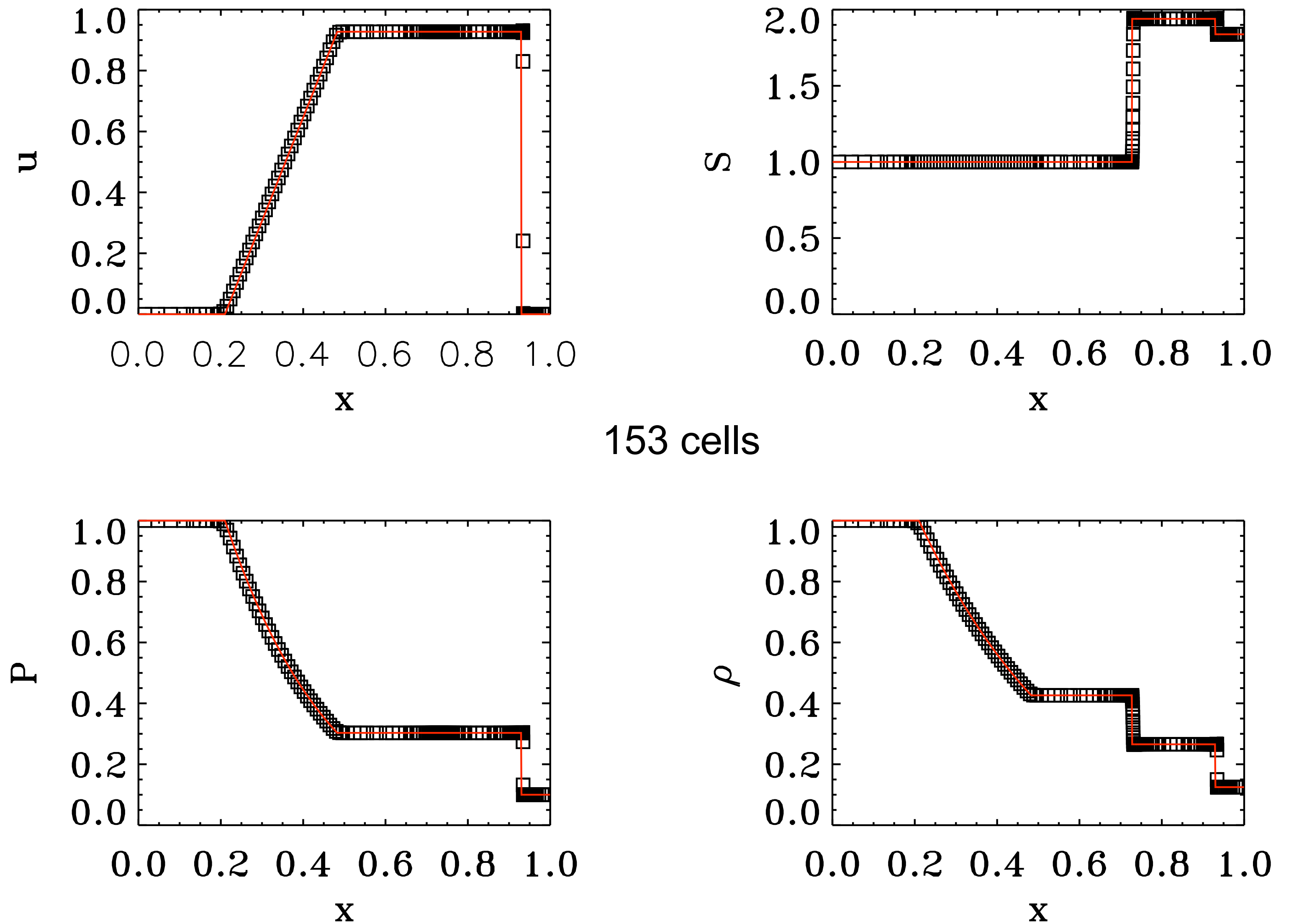


128 cells

Sod test with second order Godunov scheme



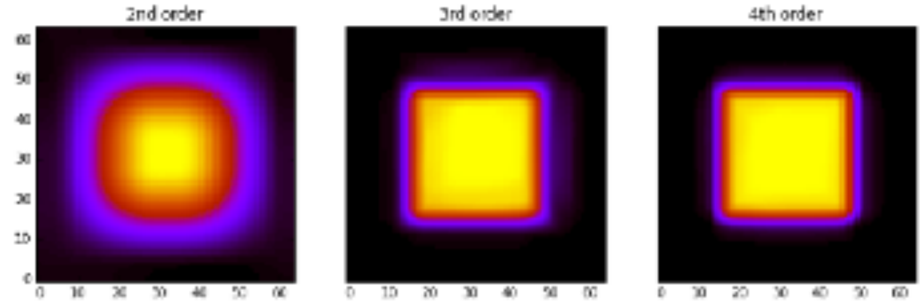
Sod test with 2nd order Godunov + AMR



Towards very high-order schemes

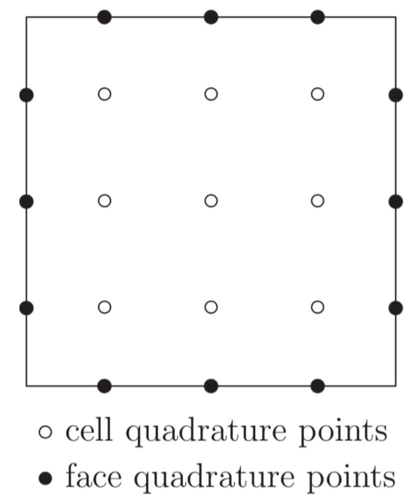
Reduce advection errors using higher-order schemes (Colella's 4th order finite-volume scheme or discontinuous Galerkin method)

Cockburn & Sho (2001), Mc Corquodale & Colella (2011)



$$\boldsymbol{u}^K(\boldsymbol{x}, t) = \sum_{l=1}^{N(k)} \boldsymbol{w}_l^K(t)\phi_l^K(\boldsymbol{x}). \qquad \boldsymbol{w}_j^K = \frac{1}{|K|}\int_K \boldsymbol{u}^K \phi_j^K \, \mathrm{d}V$$

$$\int_K \left[\frac{\partial \boldsymbol{u}^K}{\partial t} + \sum_{\alpha=1}^{3} \frac{\partial \boldsymbol{f}_\alpha}{\partial x_\alpha} \right] \phi_j^K \, \mathrm{d}V = 0.$$



$$\boxed{\frac{\mathrm{d}}{\mathrm{d}t}\int_K \boldsymbol{u}^K \phi_j^K \, \mathrm{d}V = |K| \frac{\mathrm{d}\boldsymbol{w}_j^K}{\mathrm{d}t}.} - \sum_{\alpha=1}^{3} \int_K \boldsymbol{f}_\alpha \frac{\partial \phi_j^K}{\partial x_\alpha} \, \mathrm{d}V + \boxed{\sum_{\alpha=1}^{3} \int_{\partial K} \boldsymbol{f}_\alpha n_\alpha \phi_j^K \, \mathrm{d}S} = 0,$$

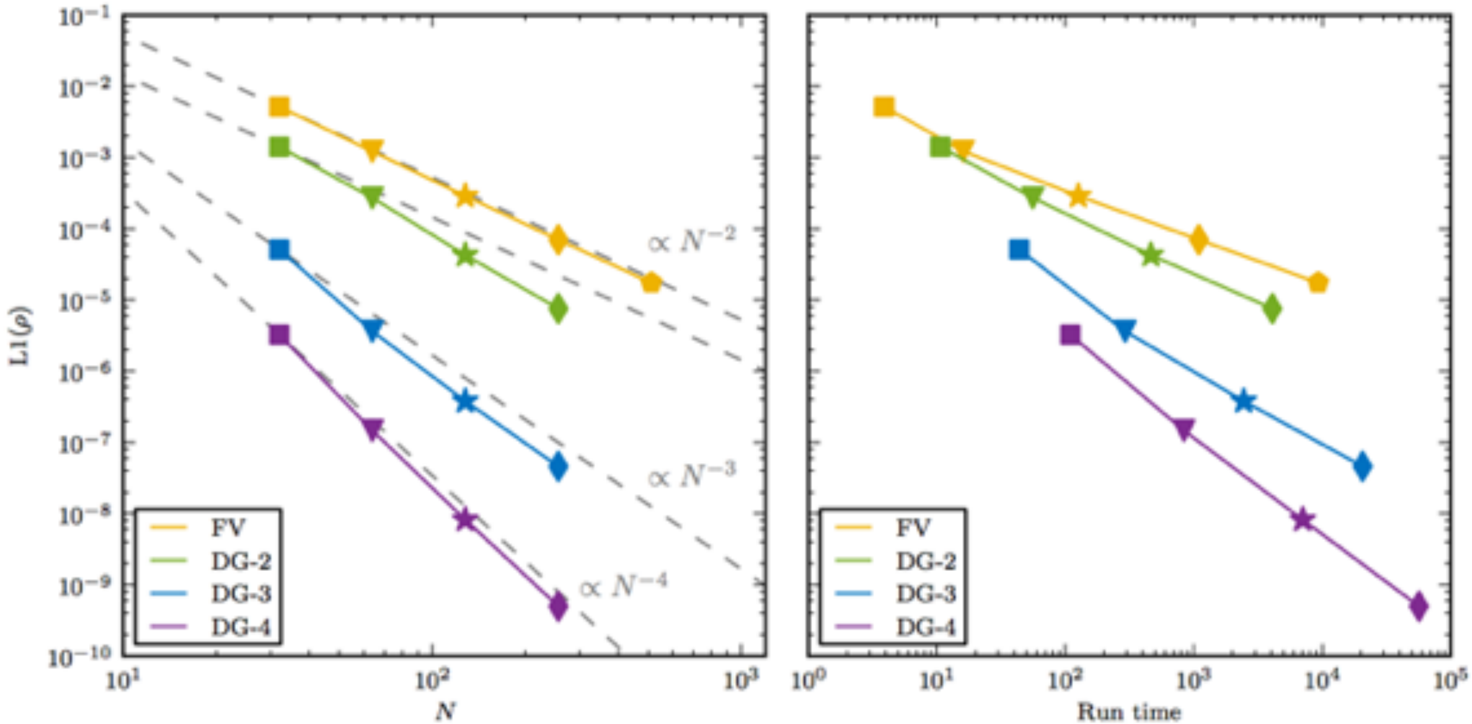
$$\boxed{= \frac{(\Delta x^K)^2}{4} \int_{\partial[-1,1]^3} \bar{\boldsymbol{f}}\left(\boldsymbol{u}^{K-}(\boldsymbol{\xi}, t), \boldsymbol{u}^{K+}(\boldsymbol{\xi}, t), \hat{n}(\boldsymbol{\xi})\right) \phi_j(\boldsymbol{\xi}) \, \mathrm{d}S_{\boldsymbol{\xi}}}$$

Towards very high-order schemes

Going to high-order is more efficient for smooth solutions.

What about discontinuous solutions ? Towards h-p adaptivity.

Schaal et al. 2015



Magneto-hydrodynamics

The ideal MHD equations in conservative forms

Mass conservation	$\partial_t \rho + \nabla \cdot \left(\rho \mathbf{v}\right) = 0$
Momentum conservation	$\partial_t \left(\rho \mathbf{v}\right) + \nabla \cdot \left(\rho \mathbf{v} \otimes \mathbf{v} - \mathbf{B} \otimes \mathbf{B}\right) + \nabla P_{\mathrm{tot}} = 0$
Total energy conservation	$\partial_t E + \nabla \cdot \left((E + P_{\mathrm{tot}})\mathbf{v} - (\mathbf{B} \cdot \mathbf{v})\mathbf{B}\right) = 0$
Magnetic flux conservation	$\partial_t (\mathbf{B}) + \nabla \times (\mathbf{B} \times \mathbf{v}) = 0$
Total energy	$E = \rho \epsilon + \frac{1}{2}\rho v^2 + \frac{1}{2}B^2$
Total pressure	$P_{\mathrm{tot}} = P + \frac{1}{2}B^2$
No magnetic monopoles	$\nabla \cdot \mathbf{B} = 0$

Cell-centered Godunov method for MHD

Natural extension of finite-volume Godunov schemes to MHD equations.

Define a volume-average magnetic field **B** in a cell *V* as:

$$\mathbf{B}_{i,j,k} = \frac{1}{V}\int_V \mathbf{B}(x,y,z)\mathrm{d}^3x$$

with $V = \left[x_{i-1/2}, x_{i+1/2}\right] \times \left[y_{i-1/2}, y_{i+1/2}\right] \times \left[z_{i-1/2}, z_{i+1/2}\right]$

Divergence cleaning methods

- Powell’s 8-wave scheme (Powell 1999)

- Projection scheme (Brackbil & Barnes 1980)

- Dedner’s diffuson scheme (Dedner et al. 2002)

A little bit of everything (Crockett et al. 2005)

div B cleaning schemes

Powell (1999) explicitly introduces magnetic monopole and magnetic current

Add source terms to the momentum equation and to the induction equation

$$\partial_t \left(\rho \mathbf{v}\right) + \nabla \cdot \left(\rho \mathbf{v} \otimes \mathbf{v} - \mathbf{B} \otimes \mathbf{B}\right) + \nabla P_{\rm tot} = -\left(\nabla \cdot \mathbf{B}\right) \mathbf{B}$$

$$\partial_t (\mathbf{B}) + \nabla \times (\mathbf{B} \times \mathbf{v}) = -\left(\nabla \cdot B\right) \mathbf{v}$$

magnetic current

Pros: magnetic monopoles are advected away. Powell's system is still hyperbolic.

Cons: the resulting scheme is not conservative. Jump relations are incorrect. Spurious Lorentz force collinear with the field. Add additional spurious dynamo channels to the induction equation.

In 1D, B_x is not constant anymore (it is advected at the flow velocity).

We now have 8 conservative variables with 8 waves (the "div B" wave).

Modify all Riemann solvers to account for this additional degree of freedom.

div B cleaning by the projection method

The previous step gives a normal magnetic flux with non-zero divergence.

Brackbill & Barnes (1980) proposed to remove explicitly magnetic monopoles using the *Projection Method* (also used in incompressible fluids)

Compute the monopole (magnetic charge) for each cell

$$(\nabla \cdot \mathbf{B})_{i,j} = \left(B^{n+1/2}_{x,i+1/2,j} - B^{n+1/2}_{x,i-1/2,j}\right)/\Delta x + \left(B^{n+1/2}_{y,i,j+1/2} - B^{n+1/2}_{y,i,j-1/2}\right)/\Delta y$$

Solve for the potential with the Poisson equation $\Delta\Phi = \nabla \cdot \mathbf{B}^{n+1/2}$

Correct the normal magnetic field with $\mathbf{B}^{n+1} = \mathbf{B}^{n+1/2} - \nabla\Phi$

It can be shown that this corrected field is the zero-divergence field closest (using the L2 norm) to the original one.

Problems: Poisson equation is non-local (elliptic) and time consuming.

Corrections in the magnetic field can result in large truncation errors in the gas pressure.

Dedner et al. (2002) develops a variant of the scheme, with an hyperbolic div B cleaning step that works also close to stagnation points.

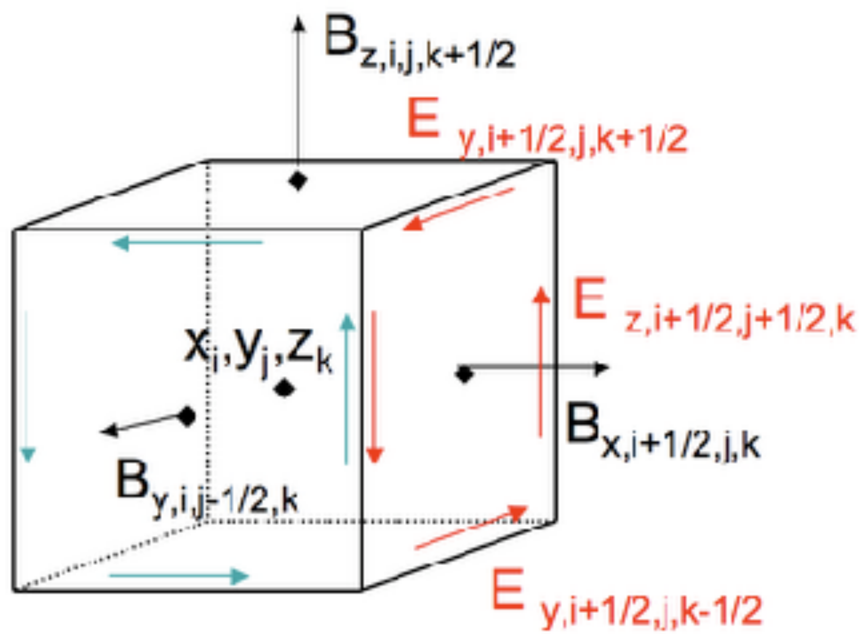
Godunov method with Constrained Transport

The induction equation in integral form suggests a surface-average form:

$$\partial_t \mathbf{B} + \nabla \times (\mathbf{B} \times \mathbf{u}) = 0 \qquad \text{(Stokes theorem)} \qquad \partial_t \int_S \mathbf{B} \cdot \mathrm{d}\mathbf{s} + \int_L (\mathbf{B} \times \mathbf{u}) \cdot \mathrm{d}\mathbf{l} = 0$$

The magnetic field is face-centred while Euler-type variables are cell-centred (staggered mesh approach).

$$(B_x)_{i+1/2,jk} = \frac{1}{S} \int_S B_x(y,z)\mathrm{d}y\mathrm{d}z \qquad S = [y_{i-1/2}, y_{i+1/2}] \times [z_{i-1/2}, z_{i+1/2}]$$



Similar to potential vector methods (Yee 1966; Dorfi 1986; Evans & Hawley 1988).

CT: exact div B preserving scheme

Surface-averaged magnetic fields are updated conservatively:

$$B^{n+1}_{z,i,j,k-1/2} = B^{n}_{x,i,j,k-1/2} + \frac{\Delta t}{\Delta x}\left(E^{n+1/2}_{y,i+1/2,j,k-1/2} - E^{n+1/2}_{y,i-1/2,j,k-1/2}\right) - \frac{\Delta t}{\Delta y}\left(E^{n+1/2}_{x,i,j+1.2,k-1/2} - E^{n+1/2}_{x,i,j-1/2,k-1/2}\right)$$

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

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

using time-averaged electric fields defined at cell edge center:

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

$$E^{n+1/2}_{y,i-1/2,j,k-1/2} = \frac{1}{\Delta t \Delta y}\int_{t^n}^{t^{n+1}}\int_{y_{j-1/2}}^{y_{j+1/2}} E_y(x_{i-1/2}, y, z_{k-1/2})\mathrm{d}t\mathrm{d}y$$

$$E^{n+1/2}_{z,i-1/2,j-1/2,k} = \frac{1}{\Delta t \Delta z}\int_{t^n}^{t^{n+1}}\int_{z_{k-1/2}}^{z_{k+1/2}} E_z(x_{i-1/2}, y_{j-1/2}, z)\mathrm{d}t\mathrm{d}z$$

The total flux (div B) across each cell bounding surface vanishes exactly !

But how do we compute the electric field on cell edges ?

The induction equation in 2D

We write Faraday's law $\partial_t \mathbf{B} = \nabla \times \mathbf{E}$ using now the EMF vector $\mathbf{E} = \mathbf{u} \times \mathbf{B}$

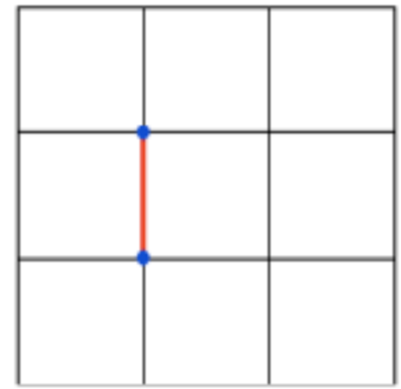
We use a finite-surface approximation for the magnetic field

$$B^n_{x,i+1/2,j} = \frac{1}{\Delta y}\int_{y_{j-1/2}}^{y_{j+1/2}} B_x(x_{i+1/2}, y)\mathrm{d}y \qquad B^n_{y,i,j+1/2} = \frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_{i+1/2}} B_y(x, y_{j+1/2})\mathrm{d}x$$

Integral form of the induction equation using Stoke's theorem

$$B^{n+1}_{x,i+1/2,j} = B^n_{x,i+1/2,j} + \frac{\Delta t}{\Delta y}\left(E^{n+1/2}_{z,i+1/2,j+1/2} - E^{n+1/2}_{z,i-1/2,j+1/2}\right)$$

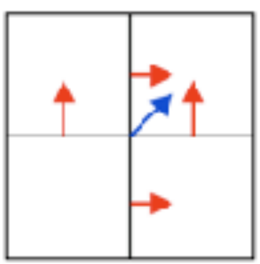
$$B^{n+1}_{y,i,j+1/2} = B^n_{y,i,j+1/2} + \frac{\Delta t}{\Delta x}\left(E^{n+1/2}_{z,i+1/2,j+1/2} - E^{n+1/2}_{z,i+1/2,j-1/2}\right)$$



By construction, div B vanishes exactly:

$$\frac{B^n_{x,i+1/2,j} - B^n_{x,i-1/2,j}}{\Delta x} + \frac{B^n_{x,i,j+1/2} - B^n_{x,i,j-1/2}}{\Delta y} = 0$$

For piece-wise initial constant data, the flux function is self-similar at corner points.

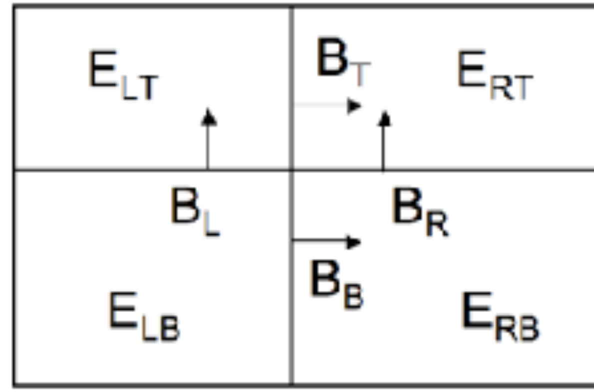


Induction Riemann problem

For pure induction, the exact Riemann solution is:

$$E^{n+1/2}_{z,i+1/2,j+1/2} = u\frac{B^n_{y,i+1,j+1/2} + B^n_{y,i,j+1/2}}{2} - v\frac{B^n_{x,i+1/2,j+1} + B^n_{y,i+1/2,j}}{2}$$
$$- |u|\frac{B^n_{y,i+1,j+1/2} - B^n_{y,i,j+1/2}}{2} + |v|\frac{B^n_{x,i+1/2,j+1} - B^n_{y,i+1/2,j}}{2}$$

2D Riemann solvers for MHD



Londrillo & Del Zana 2004, Gardiner & Stone 2005
Teyssier et al. 2006; Fromang et al. 2006
Balsara 2010, 2012

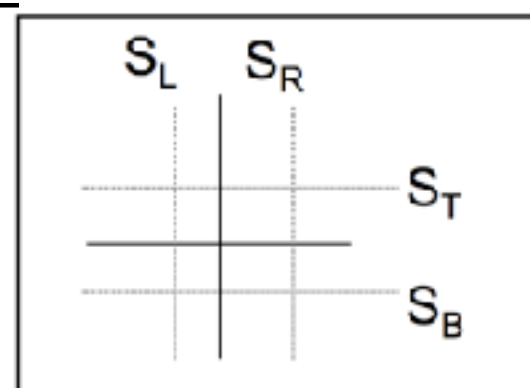
1- The Lax-Friedrich Riemann solver in 2D:

$$E^* = \frac{E_{LB}+E_{RT}+E_{RB}+E_{LT}}{4} + \frac{S_{\max}}{2}\left(B_R - B_L\right) - \frac{S_{\max}}{2}\left(B_T - B_B\right)$$

2- The HLL solver in 2D:

$$E^* = \frac{S_R S_T E_{LB} + S_L S_B E_{RT} - S_L S_T E_{RB} - S_R S_B E_{LT}}{(S_R - S_L)(S_T - S_B)}$$

$$-\frac{S_B S_L}{S_T - S_B}\left(B_R - B_L\right) + \frac{S_L S_R}{S_R - S_L}\left(B_T - B_B\right)$$



3- 2D version of HLLD in RAMSES.

Conclusion for MHD

- Ideal MHD equations in 1D can be modeled using cell-centered Godunov schemes

- We have designed several MHD Riemann solvers (they are all present in RAMSES)

- In 2D and 3D, MHD equations are more problematic: numerical build-up of magnetic monopoles: instabilities and spurious forces

- Cell-centered schemes are easier to develop but they require div B cleaning (time consuming, not robust, not conservative)

- Face-centered schemes are more natural (exact magnetic flux conservation and vanishing divergence are easy to obtain)

- Constrained Transport approach requires proper 2D upwinding of MHD waves to compute the electric field: we need 2D Riemann solvers

- 2D versions of LLF, Roe, HLL and HLLD are now available

Radiative processes

The Radiative Transfer Equation

Conservation of photons in phase-space

$$\frac{1}{c}\frac{\partial I_\nu}{\partial t} + \mathbf{n}\cdot\nabla I_\nu = -\kappa_\nu I_\nu + \eta_\nu$$

$I_\nu(\mathbf{x},\mathbf{n},t)$ radiation specific intensity

$\kappa_\nu(\mathbf{x},\mathbf{n},t)$ absorption coefficient

$\eta_\nu(\mathbf{x},\mathbf{n},t)$ source function

Source terms: microscopic collisions leading to absorption and emission.

Moments of the radiation transfer equation

Radiation energy:

$$E_\nu(\mathbf{x},t) = \int I_\nu(\mathbf{x},\mathbf{n},t)\frac{\mathrm{d}\Omega}{c}$$

Radiation flux:

$$\mathbf{F}_\nu(\mathbf{x},t) = \int I_\nu(\mathbf{x},\mathbf{n},t)\mathbf{n}\frac{\mathrm{d}\Omega}{c}$$

Pressure tensor:

$$\mathbb{P}_\nu(\mathbf{x},t) = \int I_\nu(\mathbf{x},\mathbf{n},t)\mathbf{n}\otimes\mathbf{n}\frac{\mathrm{d}\Omega}{c}$$

Energy equation:

$$\frac{\partial E_\nu}{\partial t} + \nabla\cdot\mathbf{F}_\nu = -\kappa_\nu c E_\nu + S_\nu$$

Flux equation:

$$\frac{\partial \mathbf{F}_\nu}{\partial t} + c^2\nabla\cdot\mathbb{P}_\nu = -\kappa_\nu c\mathbf{F}_\nu$$

Gnedin & Abel 2001, Aubert & Teyssier 2007, Davis et al. 2012

Radiation hydrodynamics

Fluid energy equation writes: $$\frac{\partial E}{\partial t} + \nabla \cdot \mathbf{u}(E+P) = \Gamma - \Lambda$$

Heating and cooling functions: $$\Gamma = \int \kappa_\nu c E_\nu \mathrm{d}\nu \qquad \Lambda = \int S_\nu \mathrm{d}\nu$$

Momentum equation writes: $$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u} + P\mathbb{I}) = -\rho \nabla \phi + \mathbf{F}_{\rm rad}$$

Radiation force: $$\mathbf{F}_{\rm rad} = \int \frac{\kappa_\nu}{c} \mathbf{F}_\nu \mathrm{d}\nu$$

Self-gravitating hydrodynamics coupled to radiation transport and non-equilibrium chemistry. Relevant physics for galaxy formation: photoionization of atomic species, photodissociation of molecular species, heating of dust grains

Numerical challenges:
Use operator split to perform a radiation + chemistry step after the hydro and gravity step, the main time step being controlled by standard Courant conditions.
The radiation solver is used for radiation transport, but also for chemical evolution (HI, HII, H2 and metals) and gas cooling and heating: stiff source terms.

Dichotomy in the numerical methods: SPH versus mesh for hydro
Ray-tracing versus moment-based methods for radiation

Cooling function for astrophysical plasmas

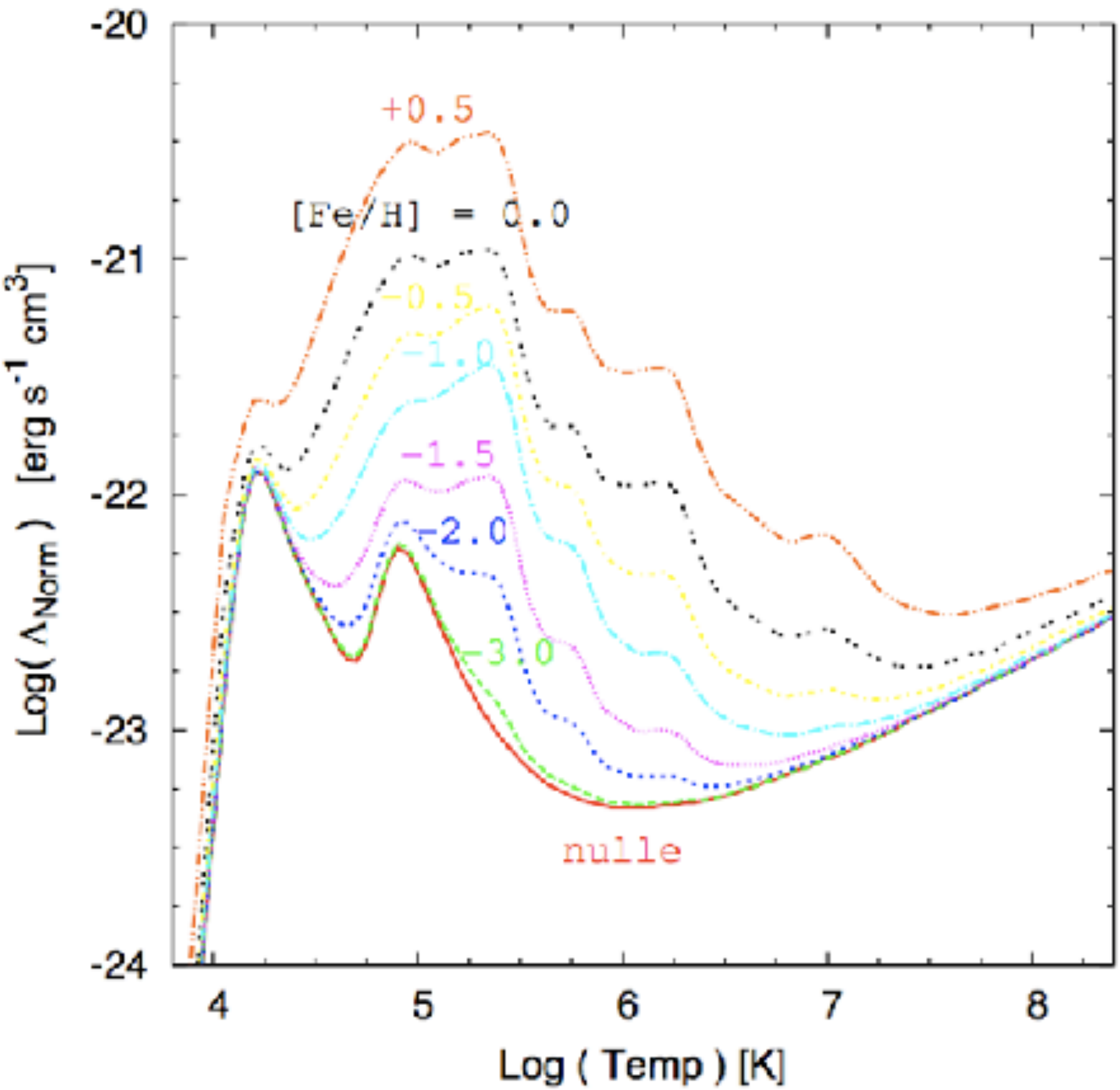
Radiation is emitted or absorbed when electrons make transitions between different states:

Bound-bound: electrons moves between 2 bound states in an atom or an ion. A photon is emitted or absorbed.

Bound-free: electrons move to the continuum (ionization) or a absorbed from the continuum to a bound state (recombination)

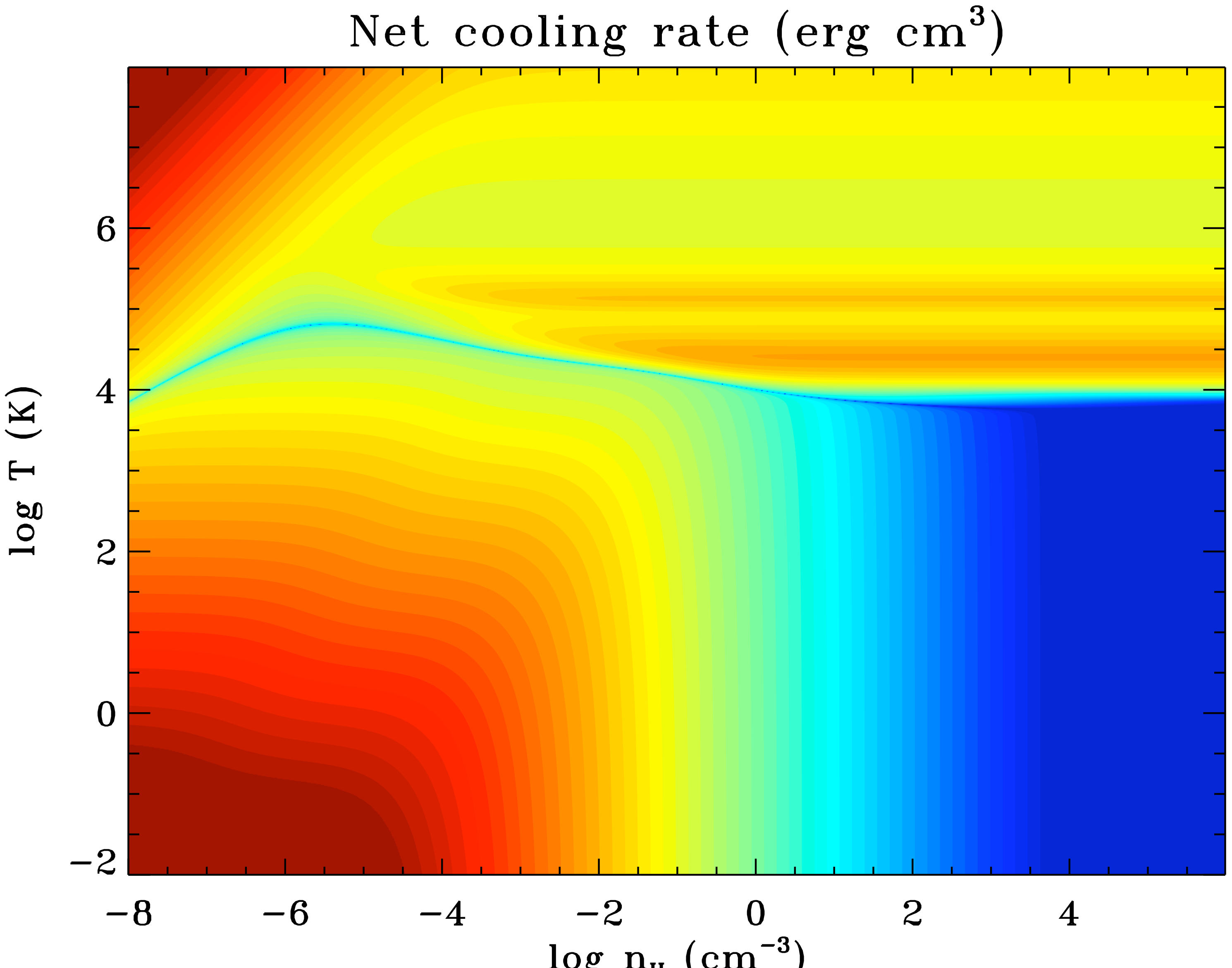
Free-free: electrons in the continuum gain or loose energy (a photon) when orbiting around ions (Bremsstrahlung).

Collisional Ionization Equilibrium: depends only on T



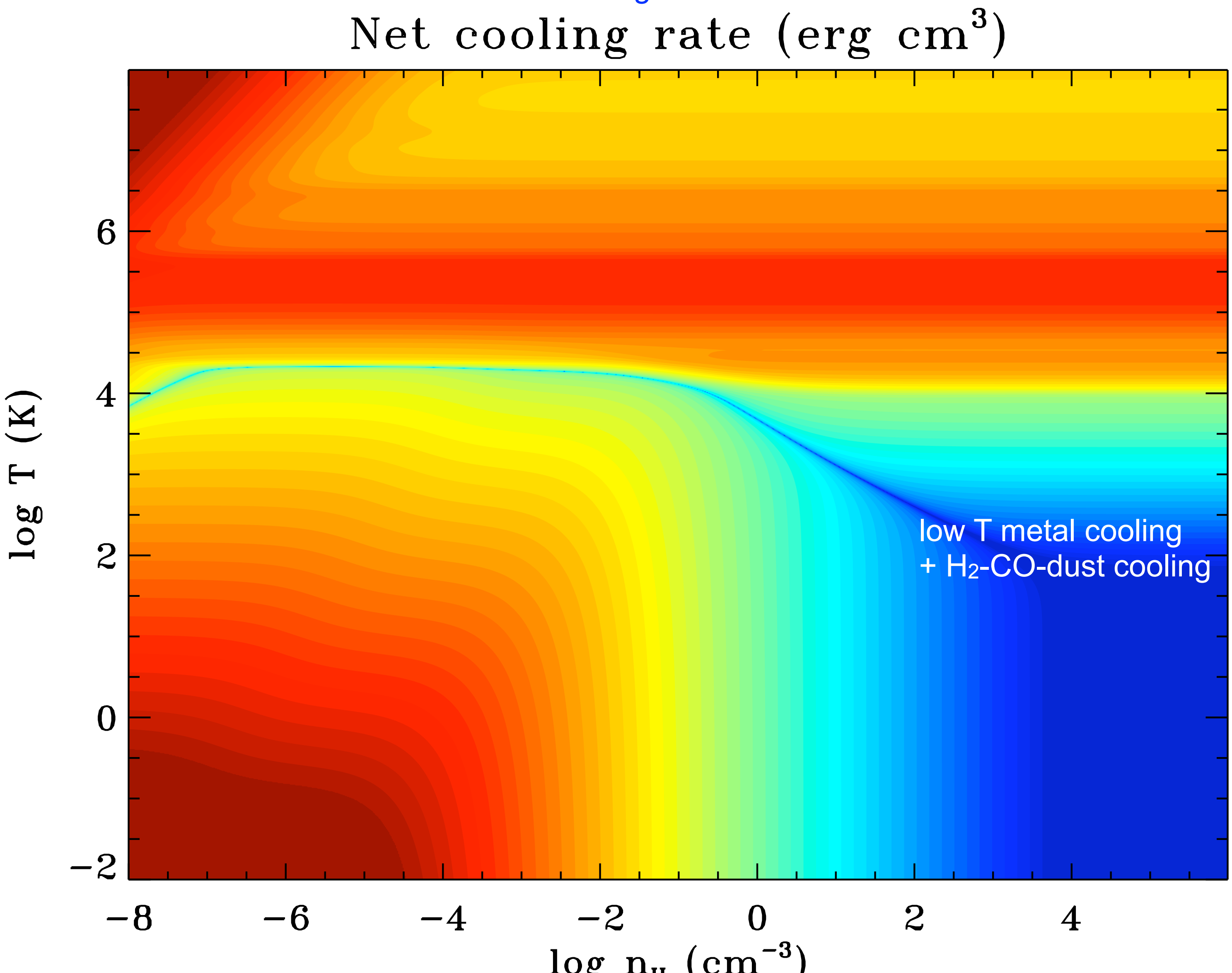
Cooling function for astrophysical plasmas

Photo-Ionization Equilibrium: depends on T and n



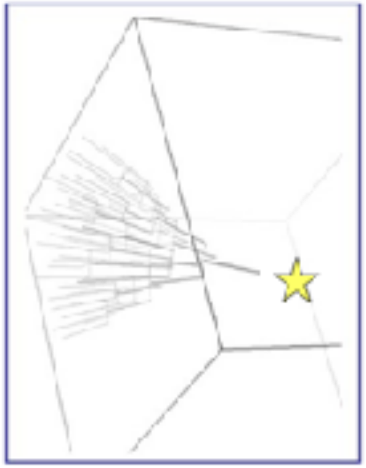
Cooling function for astrophysical plasmas

PIE including metals at solar level



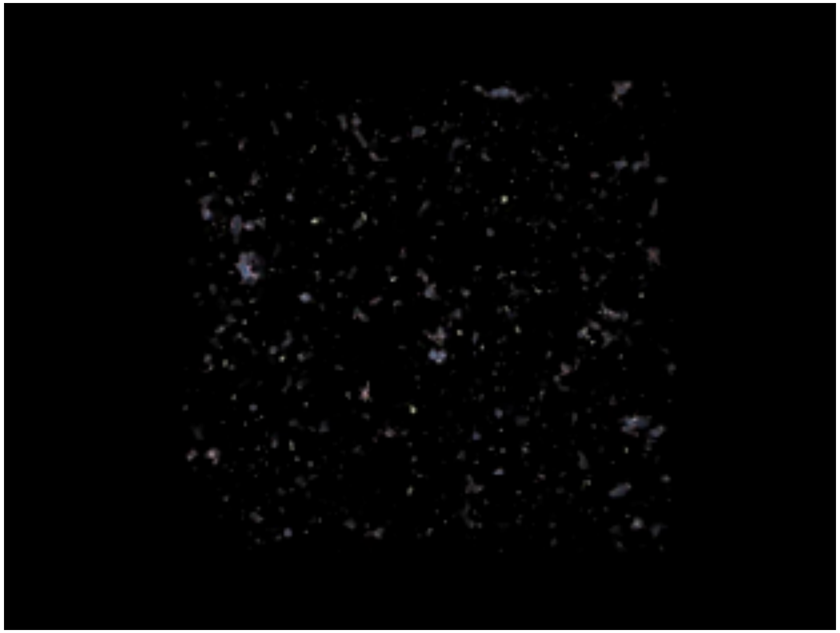
Ray-tracing methods

Abel & Wandelt (2002)



Integrate radiative transfer equation along individual light-rays
Limited by the angular discretisation (phase-space approach)
Variants:

1. Long characteristics method (Buntemeyer et al. 2016)
2. Short characteristics method (Davis et al. 2012)
3. Adaptive ray-tracing (Abel & Wandelt 2002)
4. Monte-Carlo photon packets (Pawlik et al. 2008)



Cosmic reionization

Pros:
Very good shadow capturing
Complex angular distribution
Infinite speed of light

Cons:
Sensitive to the number of sources
Infinite speed of light
Scattering and diffuse radiation

Evolutions: ray splitting and merging

The M1 moment closure

Assume that the radiation field is locally a dipole.The photon distribution function is the Lorentz transform of a Planck distribution function. Dipole is aligned locally to the radiation flux (Levermore 1984, Gonzalez et al. 2007).

The M1 Eddington tensor writes: $\mathbb{P} = \mathbb{D}E$ $\qquad \mathbb{D} = \frac{1-\chi}{2}\mathbb{I} + \frac{3\chi - 1}{2}\mathbf{n}\otimes\mathbf{n}$

where the reduced flux $\mathbf{f} = \frac{\mathbf{F}}{cE} = f\mathbf{n}$ sets $\chi = \frac{3+4f^2}{5+2\sqrt{4-3f^2}}$

Limiting cases: f=0 gives *X*=1/3 and f=1 gives *X*=1

We have a *hyperbolic* system of 4 conservations laws:

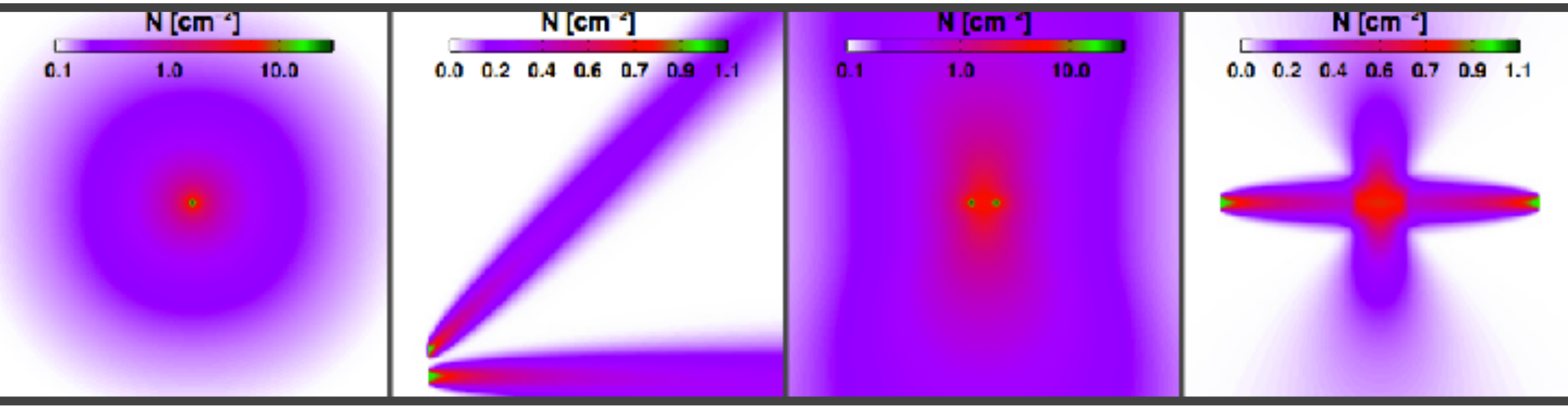
$$\frac{\partial E_\nu}{\partial t} + \nabla\cdot\mathbf{F}_\nu = 0 \qquad \frac{\partial \mathbf{F}_\nu}{\partial t} + c^2\nabla\cdot\mathbb{P}_\nu = 0$$

Godunov's method for M1 radiation transport (Aubert & Teyssier 2008, 2010)

$$\frac{(E_\nu)_i^{n+1} - (E_\nu)_i^n}{\Delta t} + \frac{(F_\nu)_{i+1/2}^{n+1/2} - (F_\nu)_{i-1/2}^{n+1/2}}{\Delta x} = 0 \qquad (F_\nu)_{i+1/2}^{n+1/2} = \int_{t^n}^{t^{n+1}} F_\nu(x_{i+1/2}, t)\mathrm{d}t$$

$$\frac{(F_\nu)_i^{n+1} - (F_\nu)_i^n}{\Delta t} + c^2\frac{(P_\nu)_{i+1/2}^{n+1/2} - (P_\nu)_{i-1/2}^{n+1/2}}{\Delta x} = 0 \qquad (P_\nu)_{i+1/2}^{n+1/2} = \int_{t^n}^{t^{n+1}} P_\nu(x_{i+1/2}, t)\mathrm{d}t$$

A Godunov scheme for M1 radiation transport



Pure radiation transport problems: successes and failures of the M1 closure

Reduced speed of light approximation

Consider a Strömgren sphere $R_s = \left(\frac{3\dot{N}_\gamma}{4\pi}\frac{1}{n_{\rm H}^2\alpha_B}\right)^{1/3}$

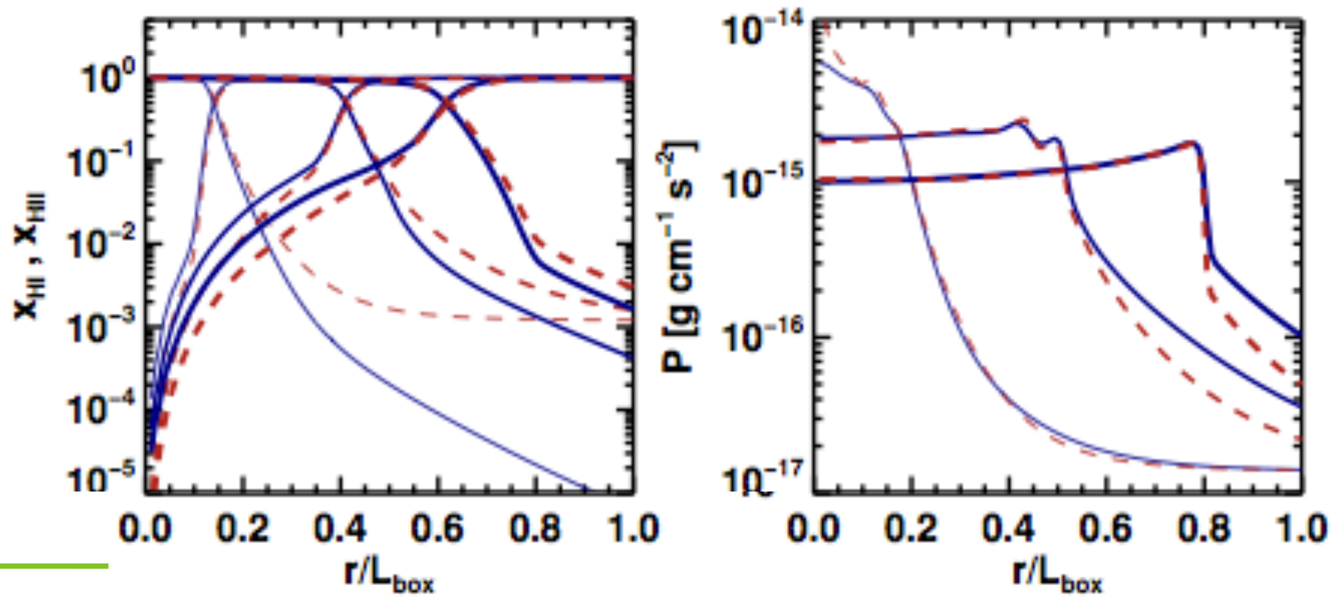
The solution becomes independent on the speed of light when $t \geq t_{\rm cross} = \frac{R_s}{c}$

Trick: use $c_{\rm reduced} = \min(c, \frac{R_s}{t})$

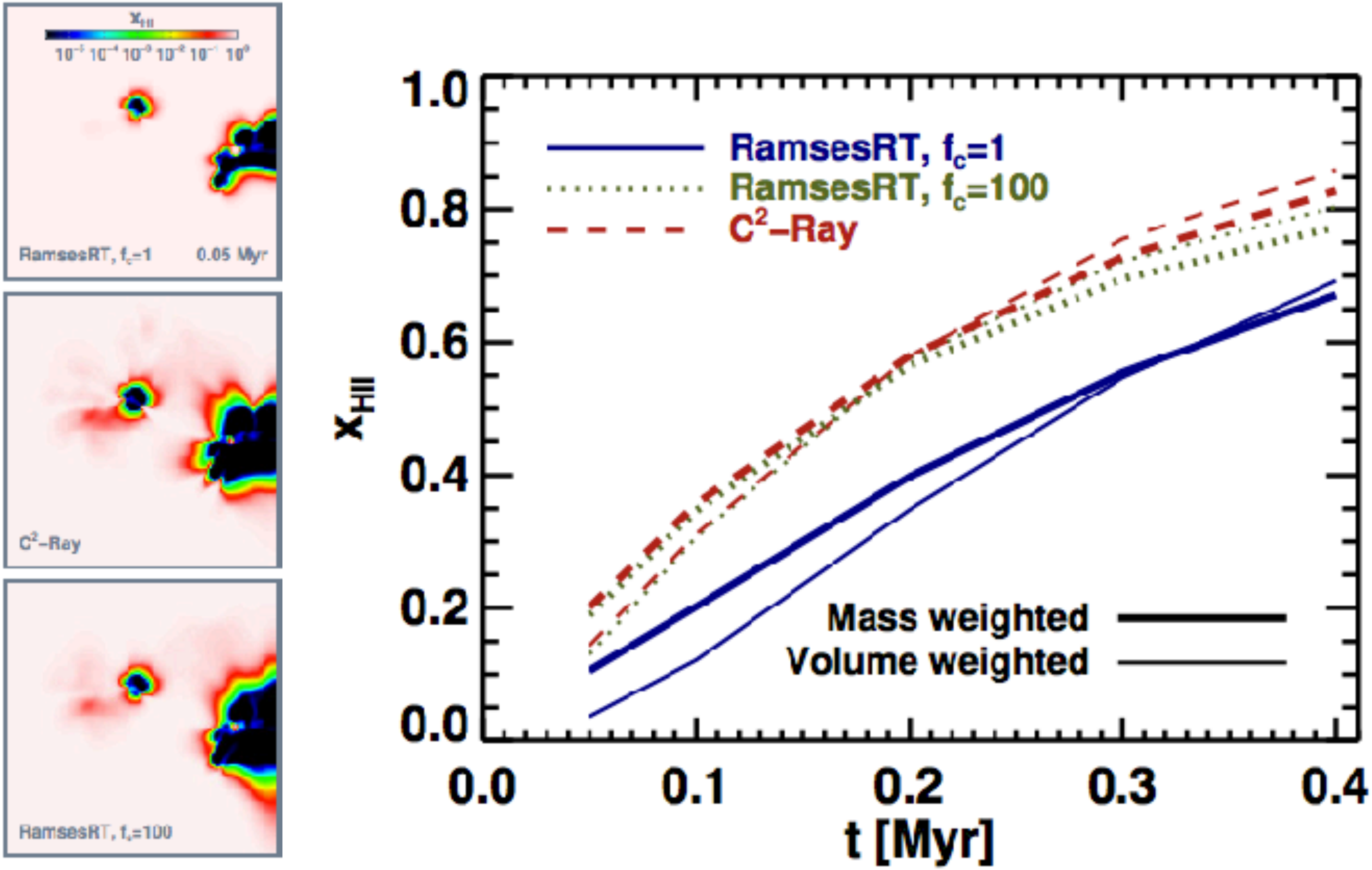
A few important examples:

	$n_{\rm H}$ [cm^{-3}]	$\dot{N}_\gamma$ [photons/sec]	r_s [kpc]	$t_{\rm cross}$ [Myr]
OB star in cloud	100	10^{48}	10^{-3}	10^{-5}
Star cluster in disc	0.1	10^{50}	1	1
Milky Way at z=0	10^{-7}	10^{52}	10^{4}	100
Iliev Test 4	10^{-4}	10^{53}	400	1

Iliev (2009) Test 9
RAMSES-RT c=0.01
versus
C2Ray
(Rosdahl et al. 2013)



Test 4 (Iliev et al. 2006) with c=1

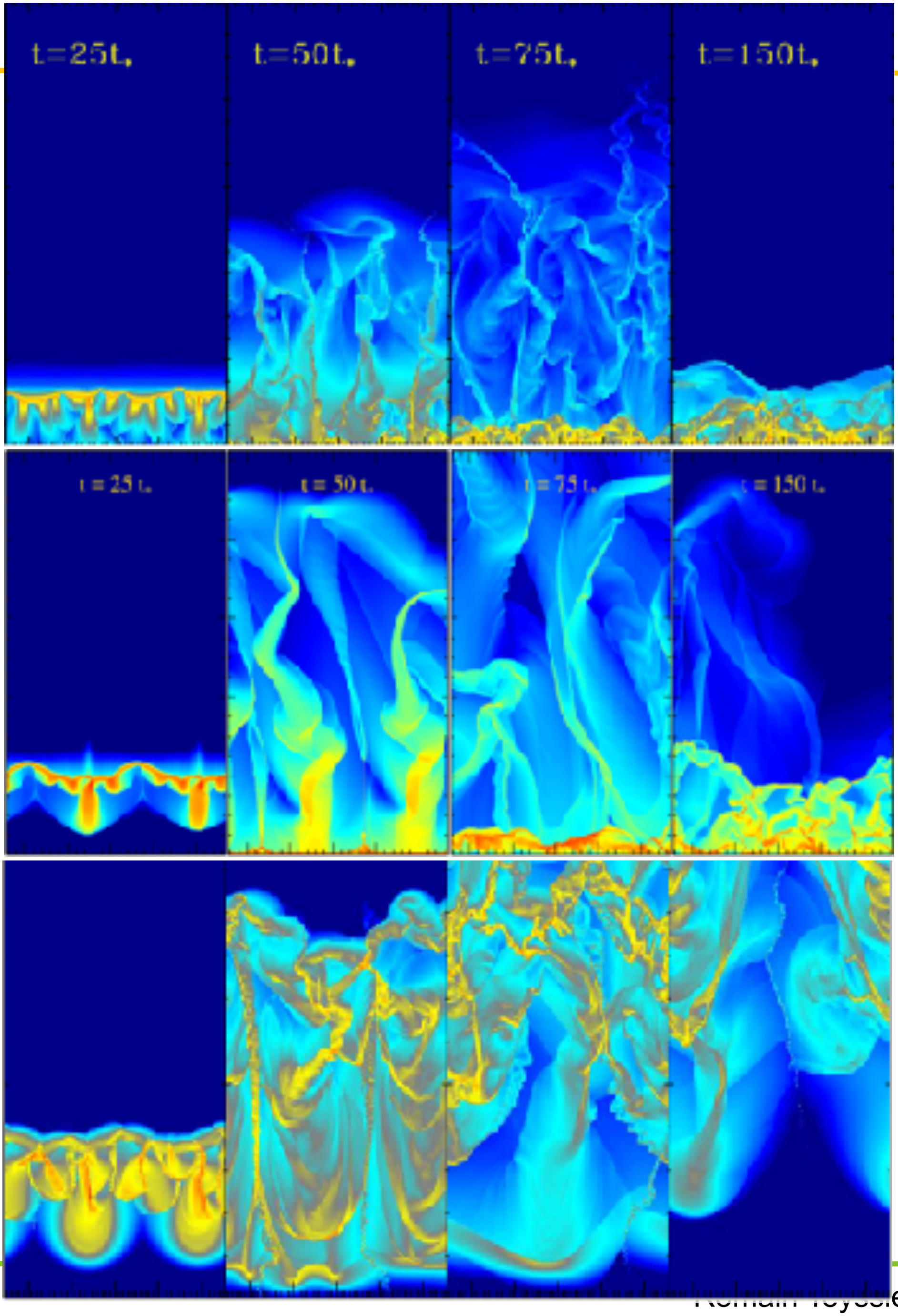


Infrared radiation pressure

FLD with ORION
Krumholz & Thomson (2012)

M1 closure with RAMSES
Rosdahl & Teyssier (2014)

VET closure with ATHENA
Davis *et al.* (2014)



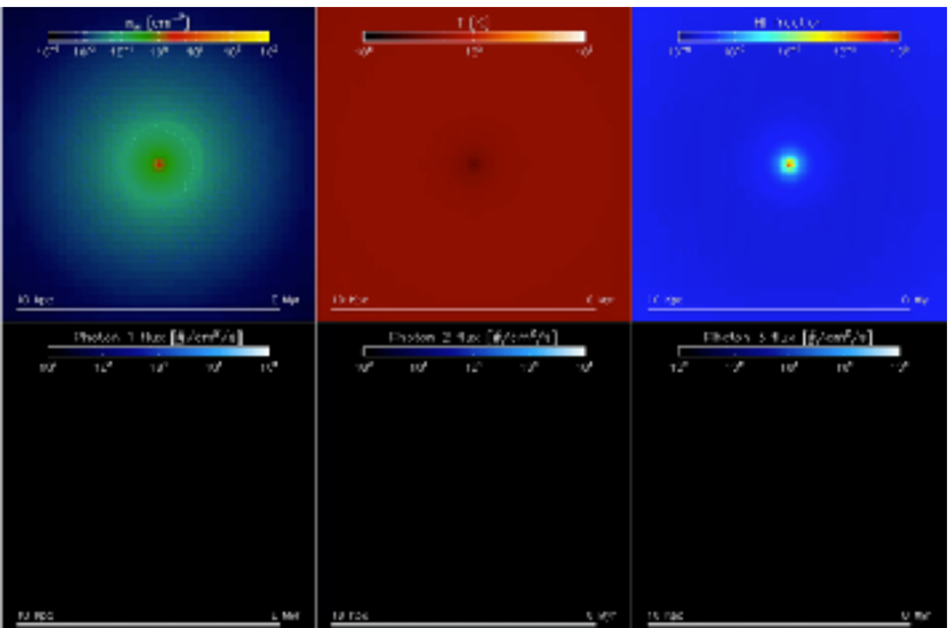
Galaxy formation with radiative MHD

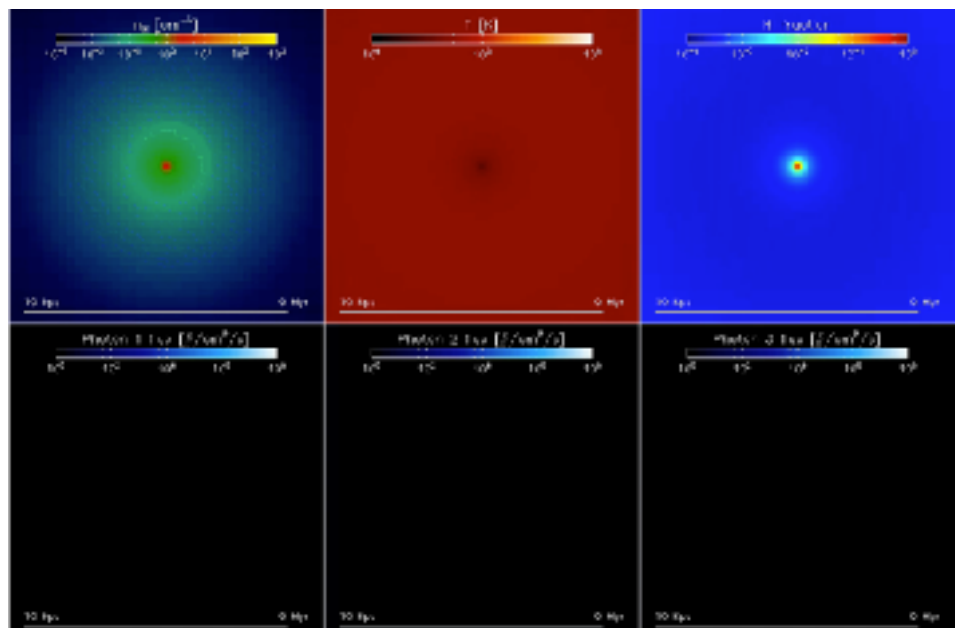
Dark matter dynamics: Tree code versus Adaptive Particle Mesh technique

Hydrodynamics: Adaptive Mesh Refinement versus Moving Meshes versus Smoothed Particle Hydrodynamics

Radiation: Ray Tracing scheme versus moment-based methods

Missing ingredients for galaxy formation simulations: star formation, stellar and SMBH feedback: subgrid scale processes





Galaxies that shine (Rosdahl et al. 2012, 2015, 2018)

Subgrid processes

Sub-Grid Scale (SGS) turbulence model

Define an averaging filter W on a smoothing scale Δ equal to the grid size.

$$\bar{\rho}(\mathbf{x},t) = \int \rho(\mathbf{x},\mathbf{x}',t) W_\Delta(\mathbf{x}-\mathbf{x}')\,\mathrm{d}^3x'$$

For compressible flows, we define the Favre average as $\tilde{\mathbf{v}} = \dfrac{\overline{\rho\mathbf{v}}}{\bar{\rho}}$.

The fluctuations are define as:

$$\rho(\mathbf{x}') = \bar{\rho} + \delta\rho$$

$$\mathbf{v}(\mathbf{x}') = \tilde{\mathbf{v}} + \delta\mathbf{v}$$

$$\mathbf{B}(\mathbf{x}') = \overline{\mathbf{B}} + \delta\mathbf{B}$$

Injecting these in the ideal MHD equation and averaging them, one finds:

$$\partial_t\left(\bar{\rho}\right) + \nabla\cdot\left(\bar{\rho}\tilde{\mathbf{v}}\right) = 0$$

$$\partial_t\left(\bar{\rho}\tilde{\mathbf{v}}\right) + \nabla\cdot\left(\bar{\rho}\tilde{\mathbf{v}}\otimes\tilde{\mathbf{v}} + (\bar{P} + \frac{\overline{B}^2}{2})\mathbb{I} - \overline{\mathbf{B}}\otimes\overline{\mathbf{B}}\right) = \nabla\cdot\mathbb{R}_\mathrm{T} + \nabla\cdot\mathbb{M}_\mathrm{T}$$

$$\partial_t\left(\overline{\mathbf{B}}\right) + \nabla\times\left(\overline{\mathbf{B}}\times\tilde{\mathbf{v}}\right) = \nabla\times\mathbf{E}_\mathrm{T}$$

Sub-Grid Scale (SGS) turbulence model

Define the turbulent kinetic energy as $K_{\rm T} = \frac{1}{2}\overline{\rho \delta \mathrm{v}^2} = \frac{1}{2}\bar{\rho}\sigma_{\rm T}^2$.

The Reynold stress tensor is defined as $\mathbb{R}_{\rm T} = -\overline{\rho \delta \mathbf{v} \otimes \delta \mathbf{v}}$.

The Maxwell stress tensor is defined as $\mathbb{M}_{\rm T} = \overline{\delta \mathbf{B} \otimes \delta \mathbf{B}} - \frac{1}{2}\overline{\delta B^2}\mathbb{I}$.

We have $\mathrm{Tr}\ \mathbb{R}_{\rm T} = -2K_{\rm T}$ and $\mathrm{Tr}\ \mathbb{M}_{\rm T} = -M_{\rm T}$.

The turbulent electromotive forced is $E_{\rm T} = \overline{\delta \mathbf{v} \times \mathbf{B}}$.

One can derive an evolution equation for the turbulent kinetic energy

$$\partial_t \left(K_{\rm T}\right) + \nabla \cdot \left(K_{\rm T}\tilde{\mathbf{v}}\right) + \frac{2}{3}K_{\rm T}(\nabla \cdot \mathbf{v}) = \mathbb{R}_{\rm T} : \nabla \tilde{\mathbf{v}} + \nabla \cdot \mathbf{Q}_{\rm T} - \frac{K_{\rm T}}{t_{\rm diss}}$$

where the dissipation time scale is given by $t_{\rm diss} = \frac{\Delta}{\sigma_{\rm T}}$ and where MHD terms have been neglected. The turbulent heat flux reads $\mathbf{Q}_{\rm T} = \frac{1}{2}\overline{\rho \delta \mathrm{v}^2 \delta \mathbf{v}}$.

One can also derive an evolution equation for the turbulent magnetic helicity

(see for example Yokoi et al. 2013)

$$H_{\rm T} = \overline{\delta \mathbf{v} \cdot \delta \mathbf{B}}$$

Sub-Grid Scale (SGS) turbulence model

Boussinesq's approximation: closure analogue to fluid viscosity

$$\mathbb{R}_{\mathrm{T}} = \boxed{\sqrt{\bar{\rho}}\Delta\sqrt{K_{\mathrm{T}}}\left(\nabla\tilde{\mathbf{v}} + \nabla\tilde{\mathbf{v}}^{\mathrm{T}} - \frac{2}{3}(\nabla\cdot\tilde{\mathbf{v}})\mathbb{I}\right)}$$

Turbulent viscosity

$$\mathbb{M}_{\mathrm{T}} = \boxed{\sqrt{\bar{\rho}}\Delta\sqrt{H_{\mathrm{T}}}\left(\nabla\overline{\mathbf{B}} + \nabla\overline{\mathbf{B}}^{\mathrm{T}}\right)}$$

Turbulent magnetic viscosity

$$\mathbf{E}_{\mathrm{T}} = \boxed{\alpha_{\mathrm{T}}\overline{\mathbf{B}}} - \boxed{\eta_{\mathrm{T}}\nabla\times\overline{\mathbf{B}}}$$

Small-scale dynamo
Turbulent resistivity

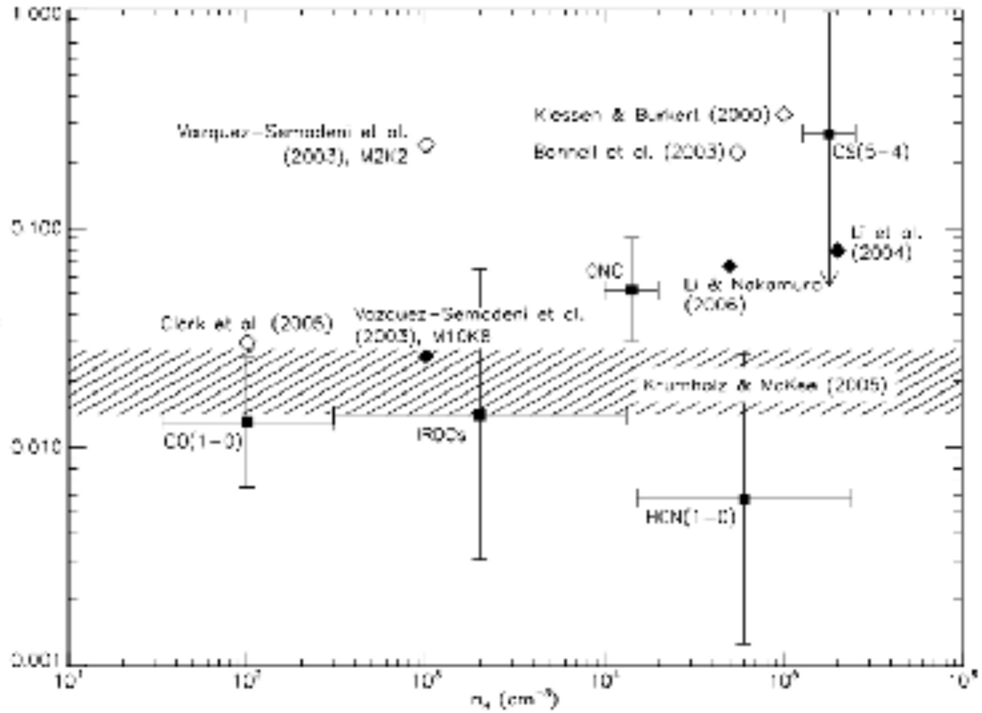
Most of these terms are already dominated by numerical diffusion (implicit SGS). No need to add them in the equations, except if one needs to compute the exact turbulent energy for other reasons.

Smagorinsky's model: stationary solution for turbulence

$$\mathbb{R}_{\mathrm{T}} : \nabla\tilde{\mathbf{v}} = \frac{K_{\mathrm{T}}}{t_{\mathrm{diss}}} \Rightarrow \bar{\rho}\Delta\sigma_{\mathrm{T}}\mathbb{S}_{\mathrm{V}} : \nabla\tilde{\mathbf{v}} = \frac{\sigma_{\mathrm{T}}}{\Delta}K_{\mathrm{T}} \qquad K_{\mathrm{T}} \simeq \bar{\rho}\Delta^2\Phi_{\mathrm{diss}}$$

Same model for the turbulent magnetic energy: $M_{\mathrm{T}} \simeq \Delta^2|\nabla\overline{\mathbf{B}}|^2$

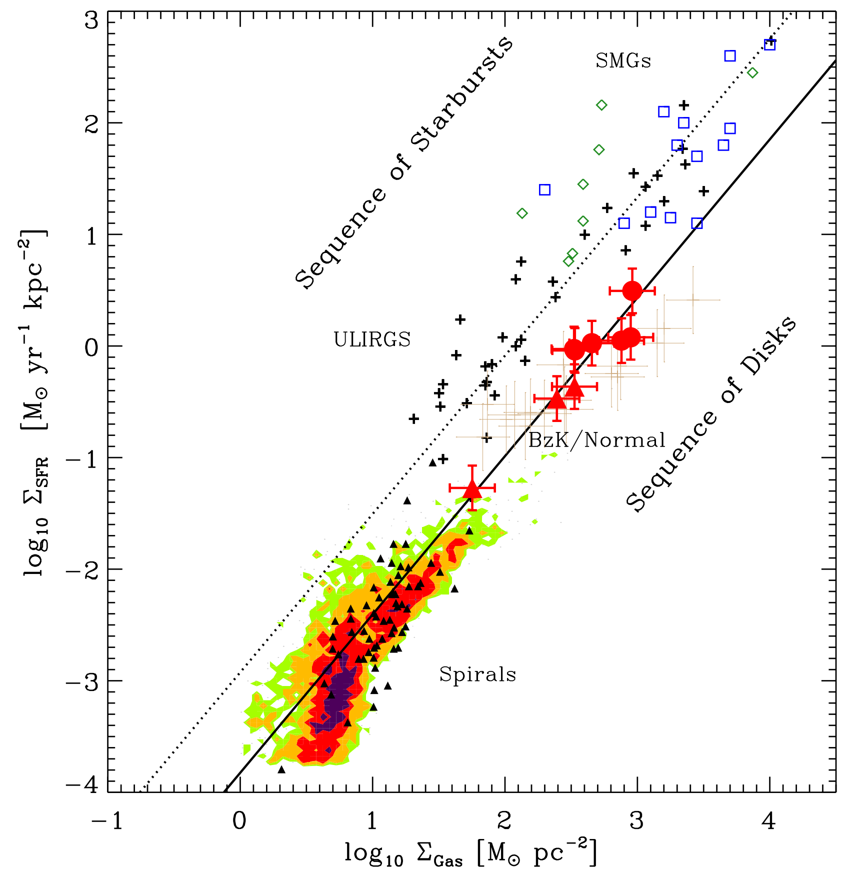
Old subgrid recipe for star formation



Schmidt law for star formation:

$$\dot{\rho}_* = \epsilon_* \frac{\rho_g}{t_{\rm ff}} \text{ for } \rho > \rho_*$$

Krumholz & Tan (2007)

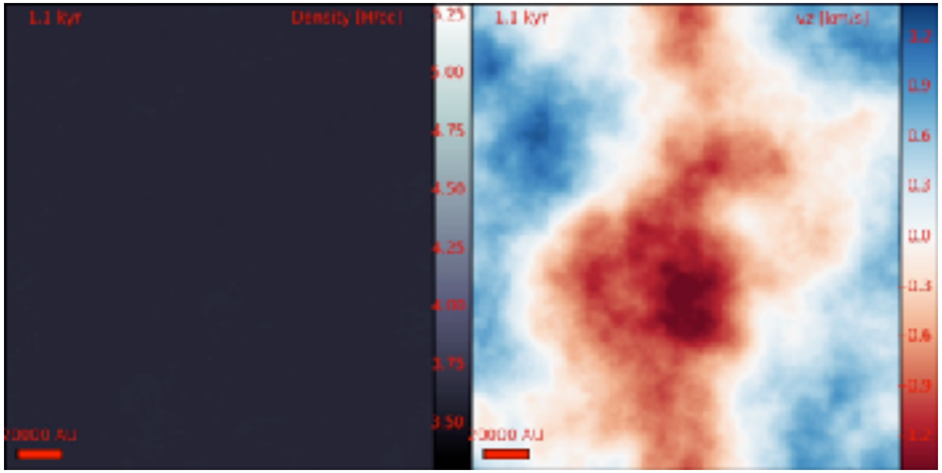


Parameters are calibrated on the Kennicutt (1998) relation

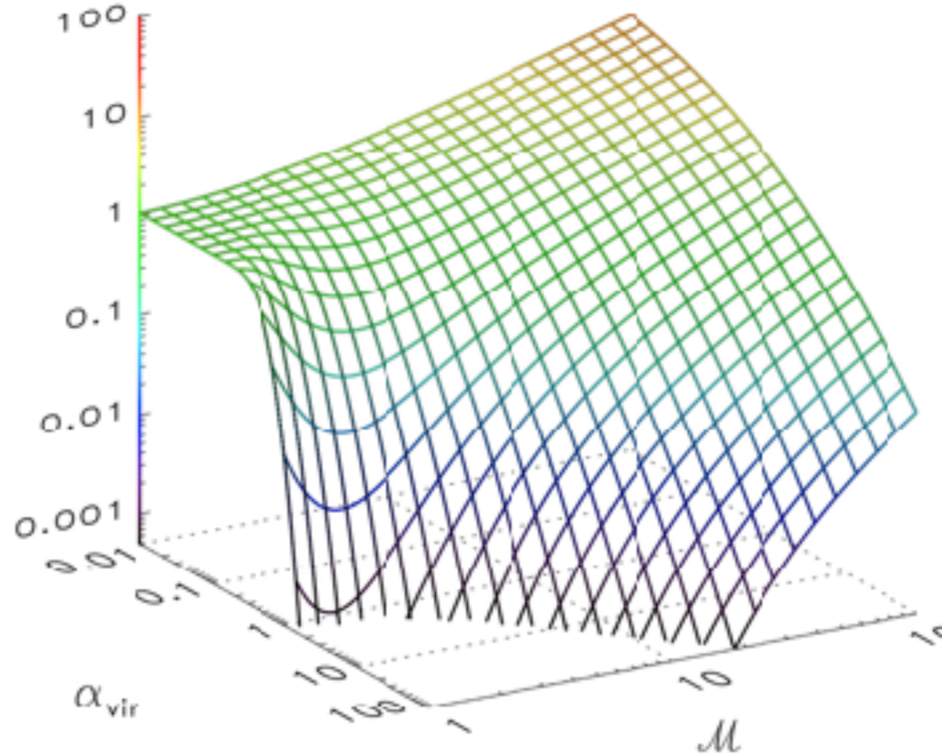
$$\Sigma_{\rm SFR} = (2.5 \pm 0.7) \times 10^{-4} \left(\frac{\Sigma_{\rm gas}}{\rm M_\odot pc^{-2}} \right)^{1.4}$$

Daddi et al. (2010)

New subgrid recipe for star formation



simulations from Tine Colman (UZH)



Federrath & Klessen (2012)

$$p(s) = \frac{1}{\sqrt{2\pi\sigma_s^2}} \exp\left(-\frac{\left(s+\frac{1}{2}\sigma_s^2\right)^2}{2\sigma_s^2}\right)$$

$$\sigma_s^2 = \ln\left(1+b^2\mathcal{M}^2\right) \qquad \mathscr{M} = \frac{\sigma_{\rm T}}{c_s}$$

$$\epsilon_{ff} = \exp\left(3/8\sigma_s^2\right)\left(1+\mathrm{erf}\left(\frac{\sigma_s^2 - s_{\rm crit}}{\sqrt{2\sigma_s^2}}\right)\right)$$

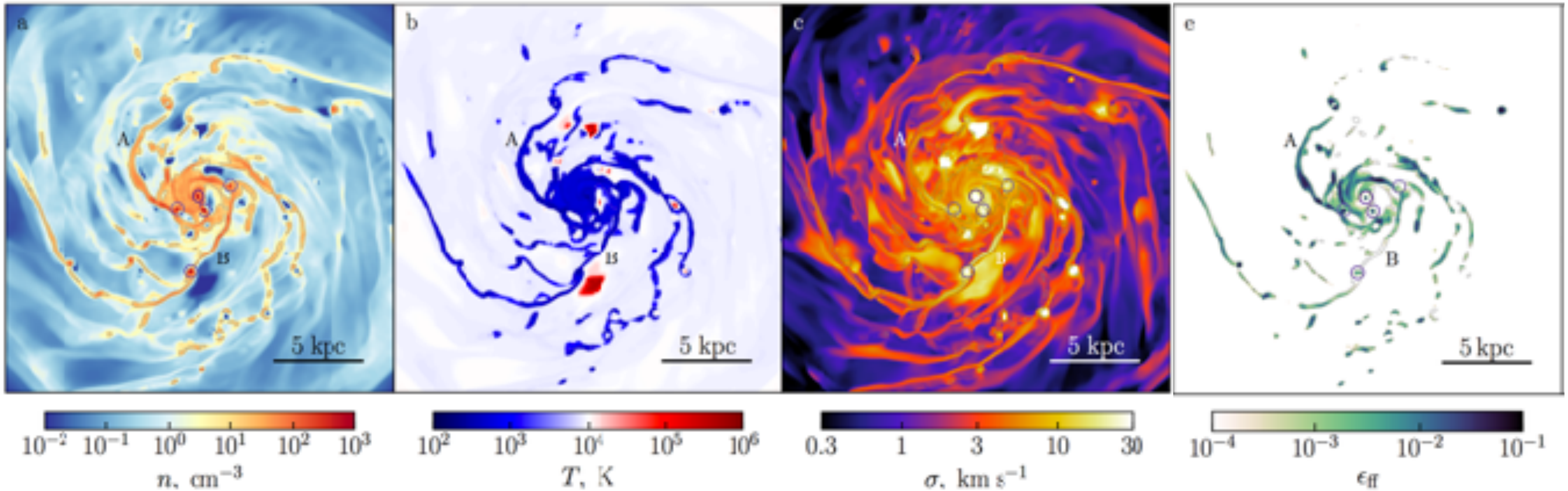
A model for the critical density:
Krumholtz & McKee (2005)

$$\rho_{\rm crit} \propto \alpha_{\rm vir}\mathscr{M}^2 \qquad \alpha_{\rm vir} = \frac{\sigma_{\rm T}^2}{G\rho_0\Delta^2}$$

New subgrid recipe based on turbulent SF theory

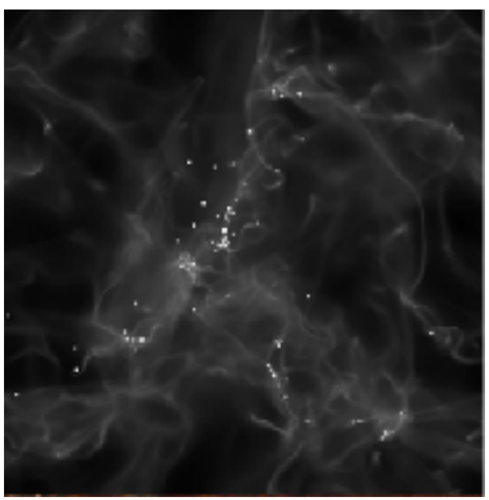
Isolated galaxy simulations with subgrid model for turbulent energy.

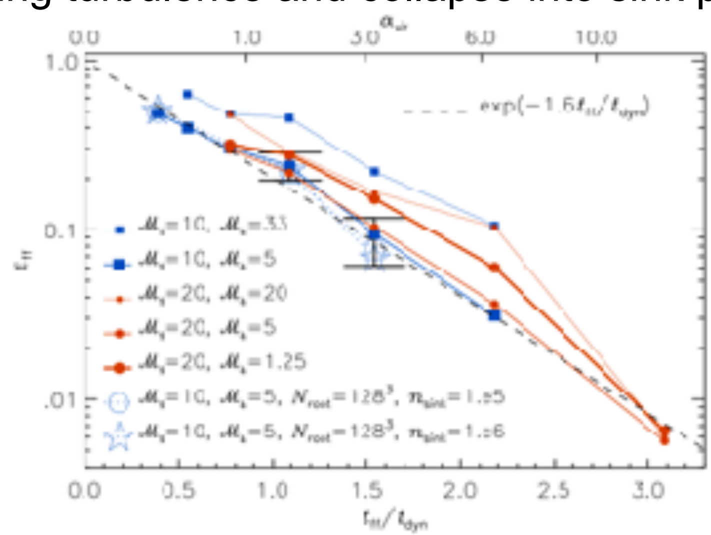
Semenov, Gnedin, Kravtsov 2016



Periodic box simulations with decaying turbulence and collapse into sink particles

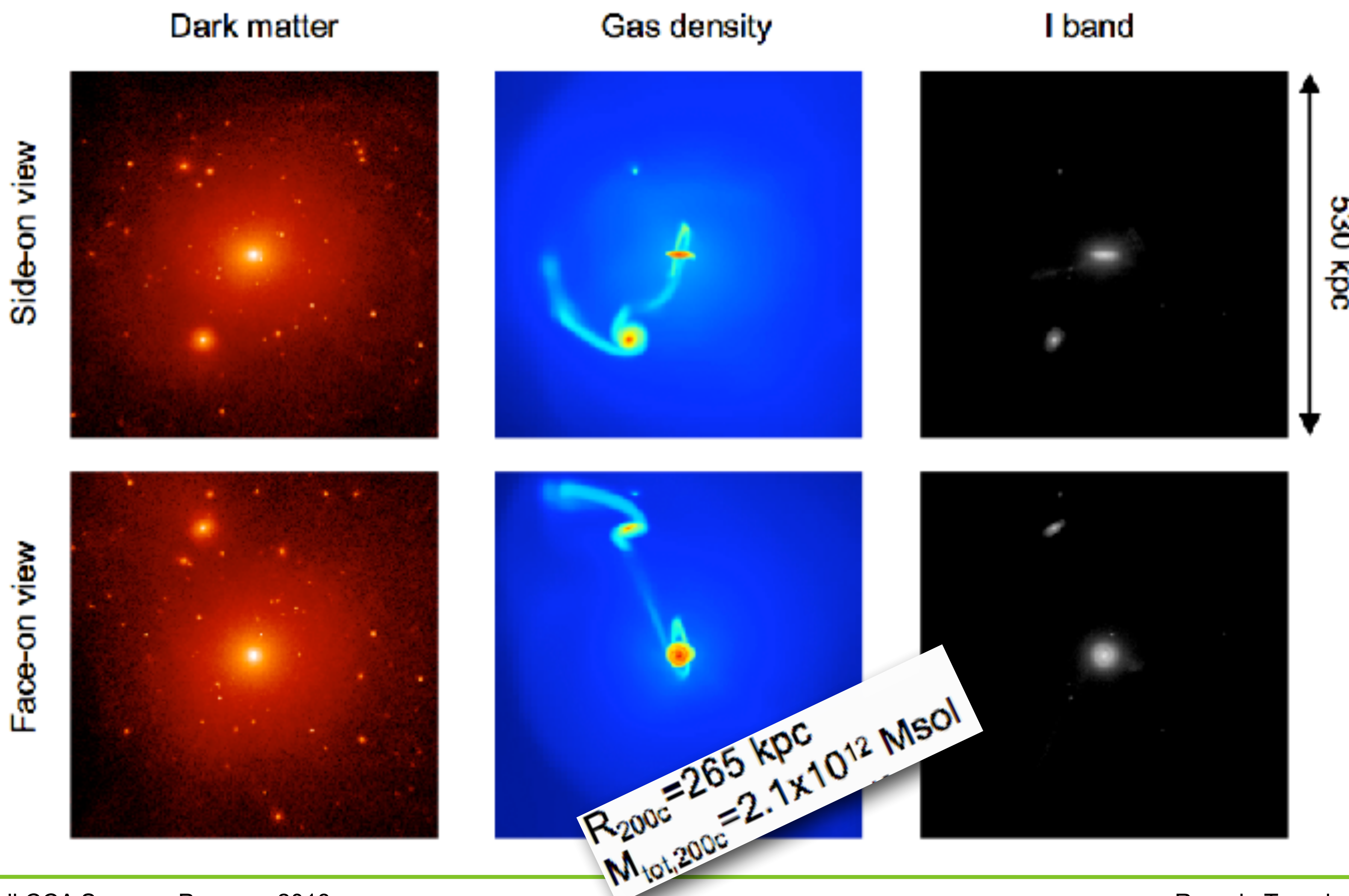
Padoan, Haugbolle, Nordlund 2012





The Aquila comparison project

Scannapieco *et al.* (2012)



The Aquila project: varying codes and physics

TABLE 1
SUMMARY OF CODE CHARACTERISTICS AND IMPLEMENTED PHYSICS.

Code	Reference	Type	UV background		Cooling	Feedback
			$z_{\rm UV}$	spectrum		
G3 (GADGET3)	[1]	SPH	6	[10]	primordial [13]	SN (thermal)
G3-BH	[1]	SPH	6	[10]	primordial [13]	SN (thermal), BH
G3-CR	[1]	SPH	6	[10]	primordial [13]	SN (thermal), BH, CR
G3-CS	[2]	SPH	6	[10]	metal-dependent [14]	SN (thermal)
G3-TO	[3]	SPH	9	[11]	metal-dependent [15]	SN (thermal+kinetic)
G3-GIMIC	[4]	SPH	9	[11]	metal-dependent [15]	SN (kinetic)
G3-MM	[5]	SPH	6	[10]	primordial [13]	SN (thermal)
G3-CK	[6]	SPH	6	[10]	metal-dependent [14]	SN (thermal)
GAS (GASOLINE)	[7]	SPH	10	[12]	metal-dependent [16]	SN (thermal)
R (RAMSES)	[8]	AMR	12	[10]	metal-dependent [14]	SN (thermal)
R-LSFE	[8]	AMR	12	[10]	metal-dependent [14]	SN (thermal)
R-AGN	[8]	AMR	12	[10]	metal-dependent [14]	SN (thermal), BH
Arepo	[9]	Moving Mesh	6	[10]	primordial [12]	SN (thermal)

NOTE. — [1] Springel et al. (2008); [2] Scannapieco et al. (2005); Scannapieco et al. (2006); [3] Okamoto et al. (2010); [4] Crain et al. (2009); [5] Murante et al. (2010); [6] Kobayashi (2007); [7] Stinson et al. (2006); [8] Teyssier (2002); Rasera & Teyssier (2006); Dubois & Teyssier (2008); [9] Springel (2010a); [10] Haardt & Madau (1996); [11] Haardt & Madau (2001); [12] Haardt & Madau (2005, private communication); [13] Katz et al. (1996); [14] Sutherland & Dopita (1993); [15] Wiersma et al. (2009a); [16] Shen et al (2010).

The Aquila project contenders

AREPO: moving mesh code (Springel 2010)

Lagrangian mesh points : Galilean invariance

Voronoï tessellation to define interfaces between finite volume elements. Based on the Godunov methodology: Riemann solver + slope limiter.

Feedback models similar to GADGET.

RAMSES: AMR Eulerian code (Teyssier 2002)

Different stellar feedback implementations, all inefficient at large halo masses.

AGN feedback model à la Booth & Schaye 2011

GASOLINE: standard SPH (Wadsley, Stadel & Quinn 2004)

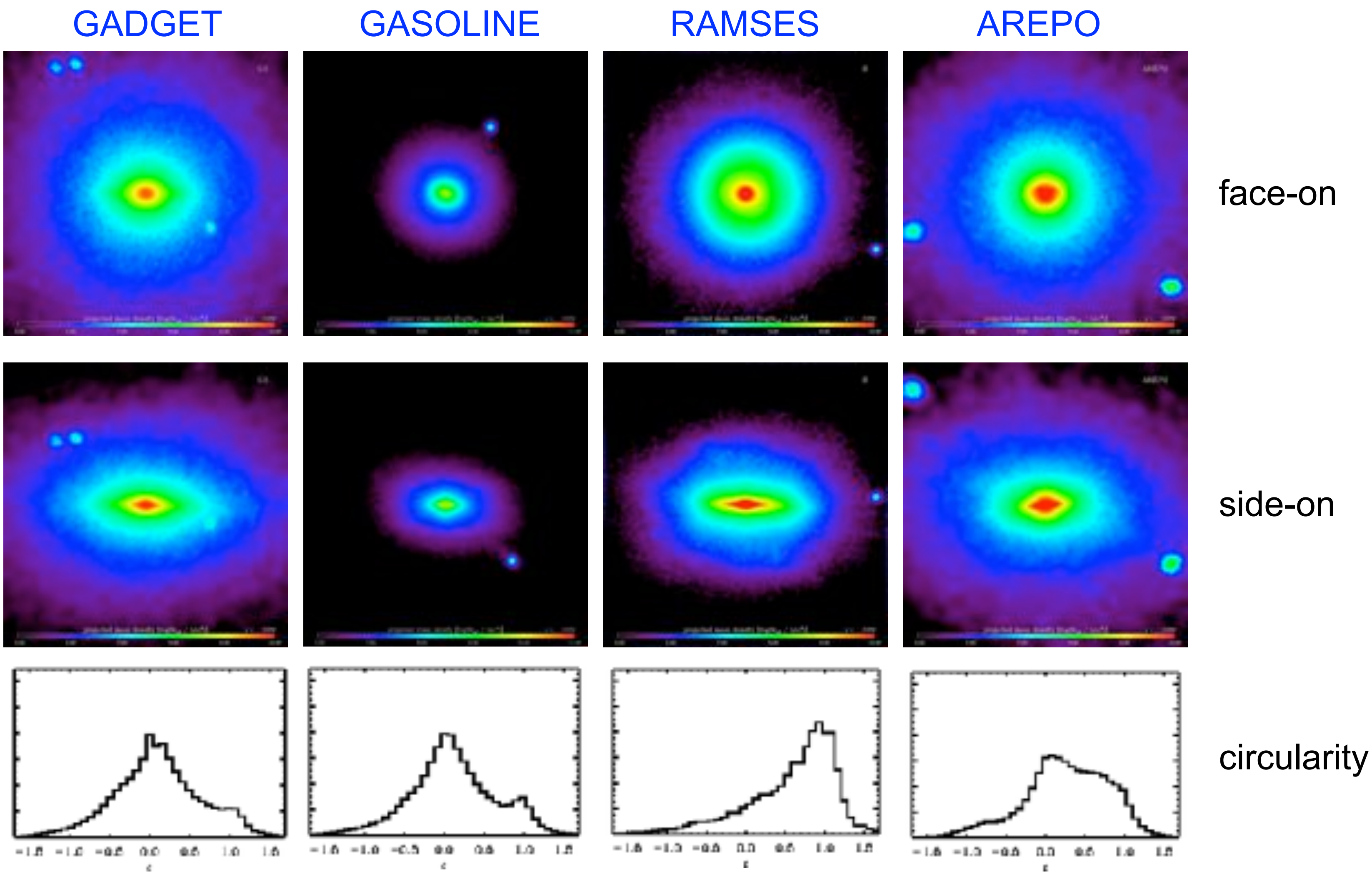
Delayed cooling with blast wave model. Inefficient at large halo masses.

GADGET: standard SPH (Springel 2005)

Many different versions with various feedback recipe.

AGN feedback à la Sijacki et al.

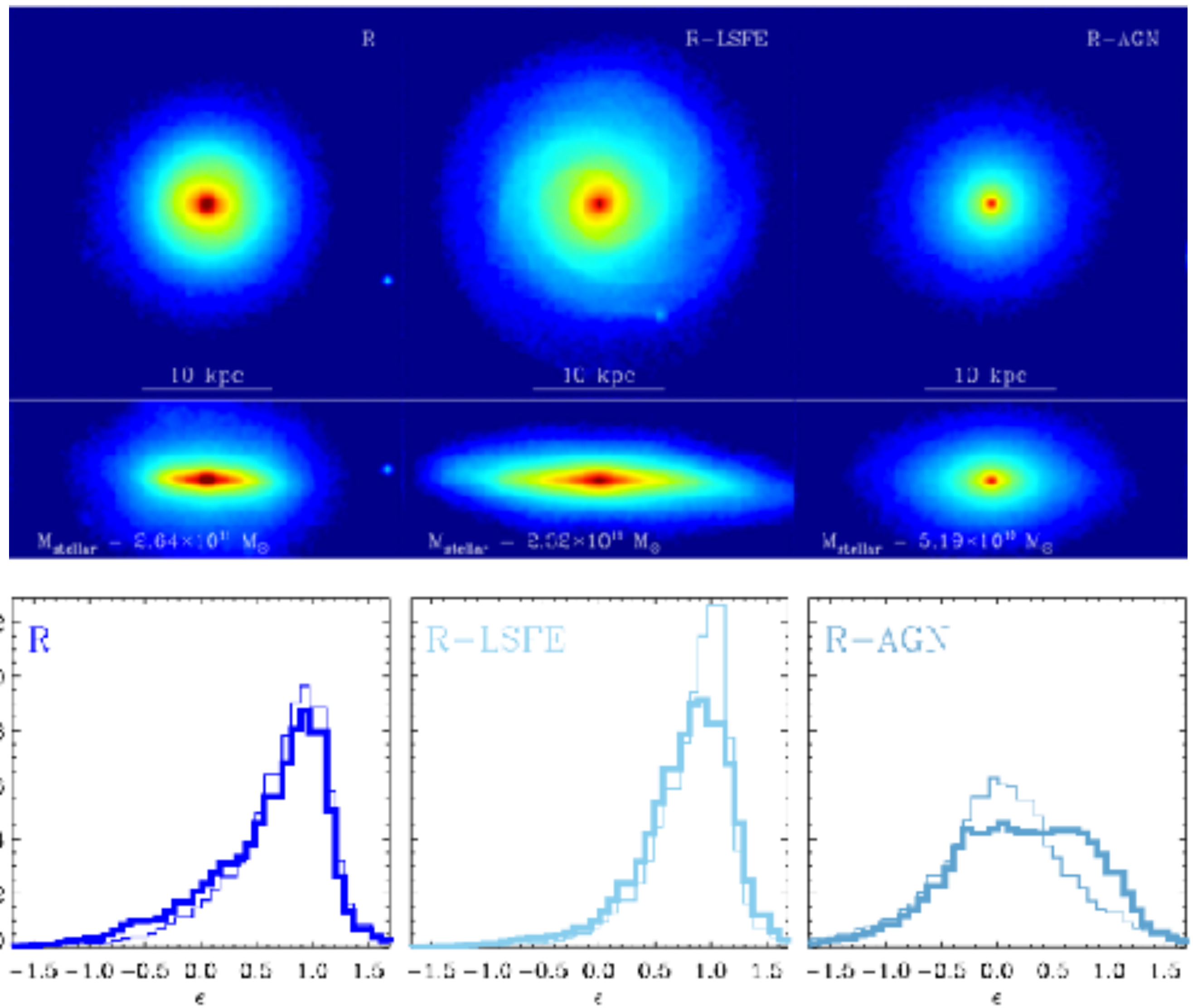
Different codes, same physics, different morphologies...



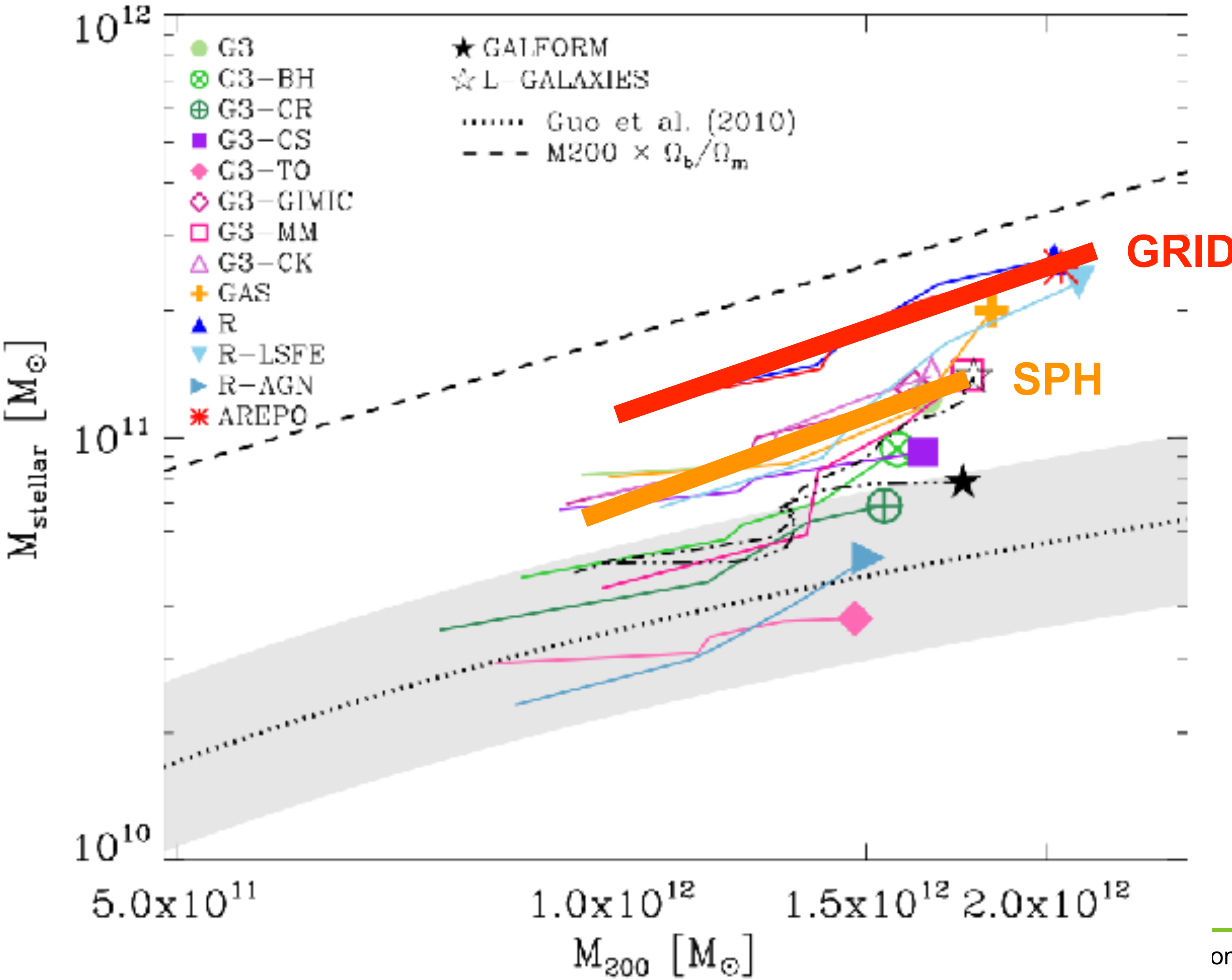
Low resolution runs

Same code, different subgrid models, different morphologies...

RAMSES



Feedback and SF matter more than code type.



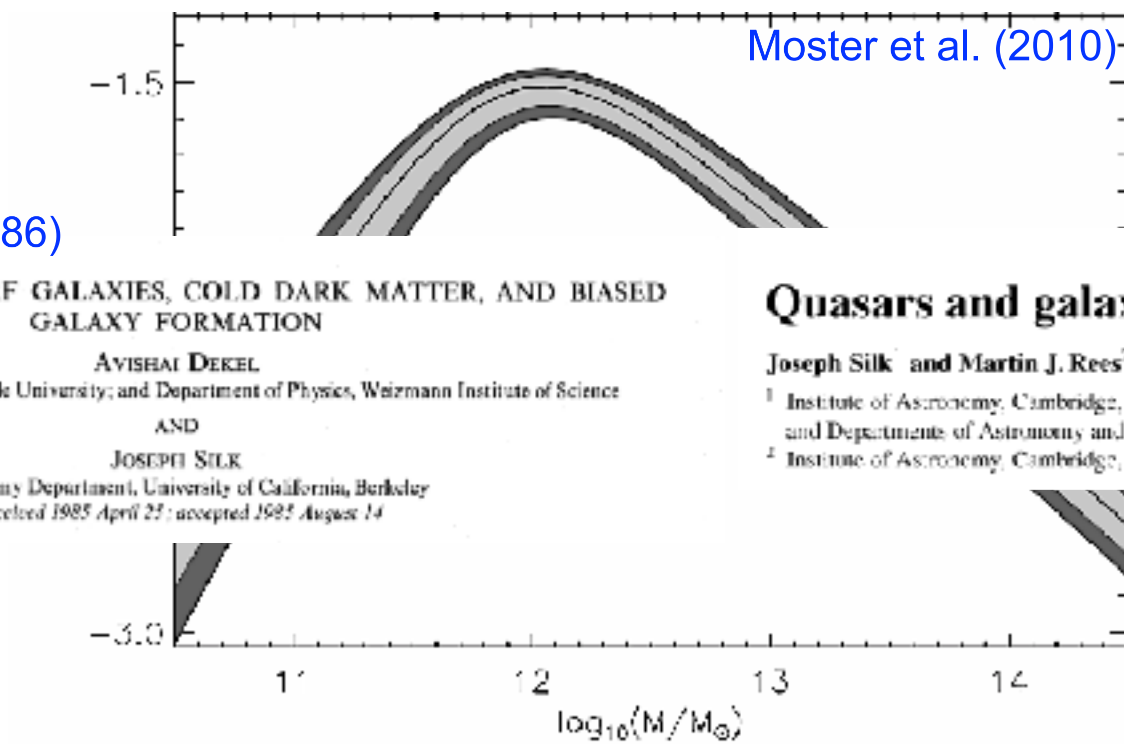
Feedback and star formation in galaxy formation

Very low efficiency of gas conversion into star.

Small mass galaxies are dominated by stellar feedback.

Large mass galaxies are governed by AGN feedback.

Stellar-to-halo mass ratio



Dekel & Silk (1986)

THE ORIGIN OF DWARF GALAXIES, COLD DARK MATTER, AND BIASED GALAXY FORMATION

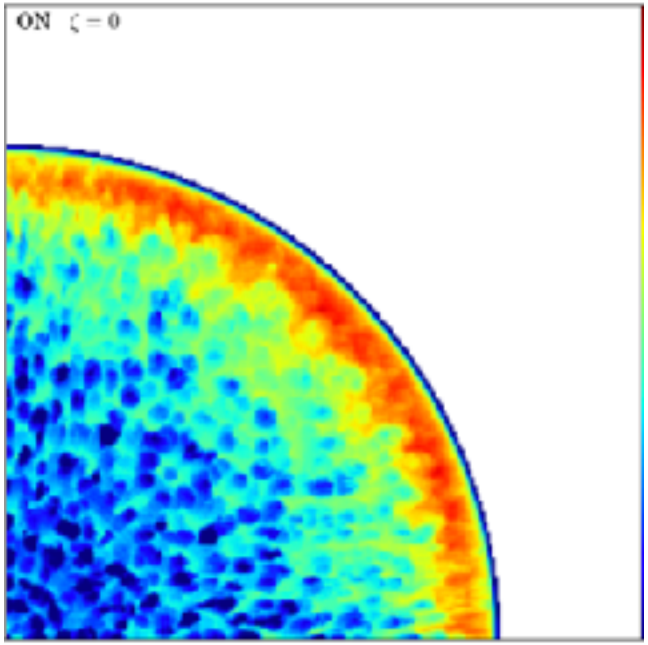
AVISHAI DEKEL
Department of Astronomy, Yale University; and Department of Physics, Weizmann Institute of Science
AND
JOSEPH SILK
Astronomy Department, University of California, Berkeley
Received 1985 April 25; accepted 1985 August 14

Silk & Rees (1998)

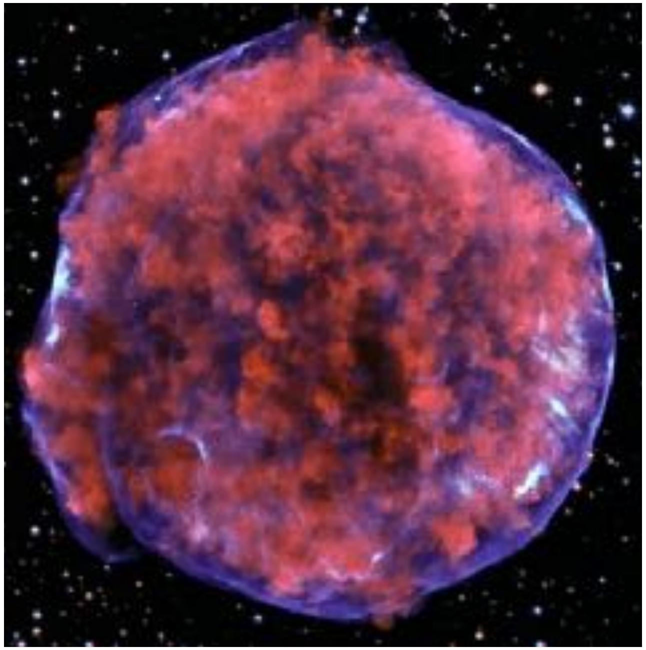
Quasars and galaxy formation

Joseph Silk[1] and Martin J. Rees[2]
[1] Institute of Astronomy, Cambridge, UK, Institut d'Astrophysique de Paris, France, and Departments of Astronomy and Physics, University of California, Berkeley, CA 9
[2] Institute of Astronomy, Cambridge, UK

Implementing supernova feedback



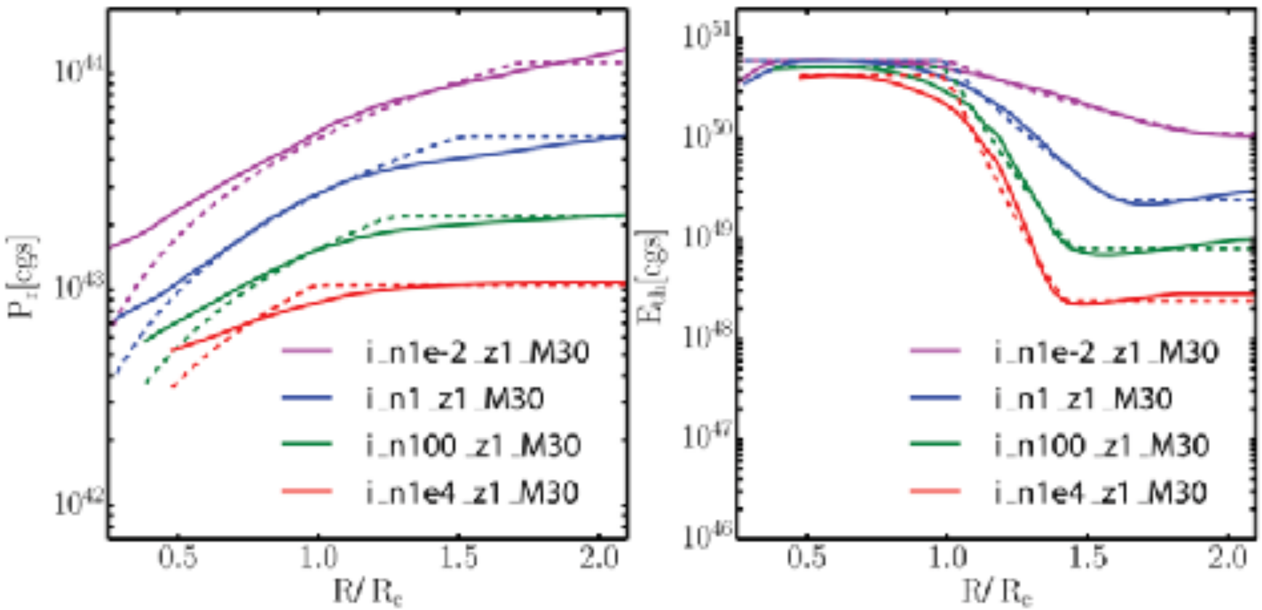
Simulation (Ferrand et al. 2016)



Chandra image of Tycho

Stellar momentum feedback

Martizzi et al. (2015)



At early time, energy-conserving Sedov phase.

At late time, momentum-conserving snow-plow phase.

Cooling radius: $R_c \simeq 3\text{pc} \left(\frac{n_\text{H}}{100\ \text{H/cc}}\right)^{-0.4}$

If cooling radius is not resolved, inject terminal radial momentum

$$P_\text{SN} \simeq 10^4 M_\odot \text{km(sec} \left(\frac{n_\text{H}}{100\ \text{H/cc}}\right)^{-0.2}$$

Stellar feedback with delayed cooling

Other popular approach: delay the snow-plow phase so that enough momentum is injected at the resolution limit.

Stellar winds, supernovae remnants are highly turbulent environment, filled with cosmic rays and magnetic field. Thermal energy dissipates almost instantaneously through cooling. Non-thermal processes dissipate much more slowly.

Hanasz et al. 2009 ; Scannapieco & Brüggen 2010; Wadepuhl & Springel 2011 and others...

In Teyssier et al. (2013), the non-thermal energy is captured as:

$$\rho \frac{D\epsilon_{turb}}{Dt} = \dot{E}_{inj} - \frac{\rho \epsilon_{turb}}{t_{diss}} \qquad \epsilon_{turb} = \sigma_{turb}^2$$

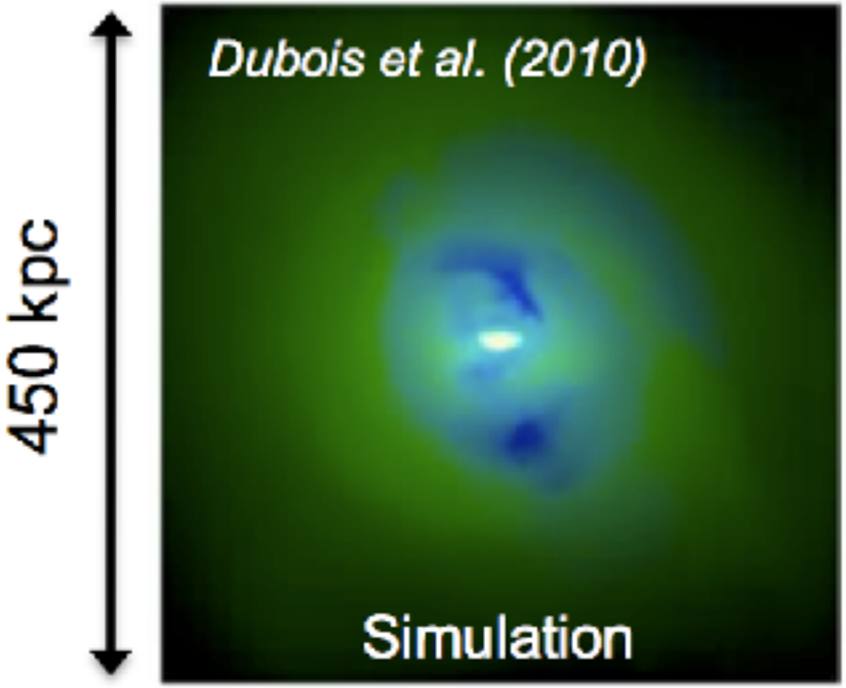
The total dynamical pressure is $P_{tot} = P_{thermal} + P_{turb}$

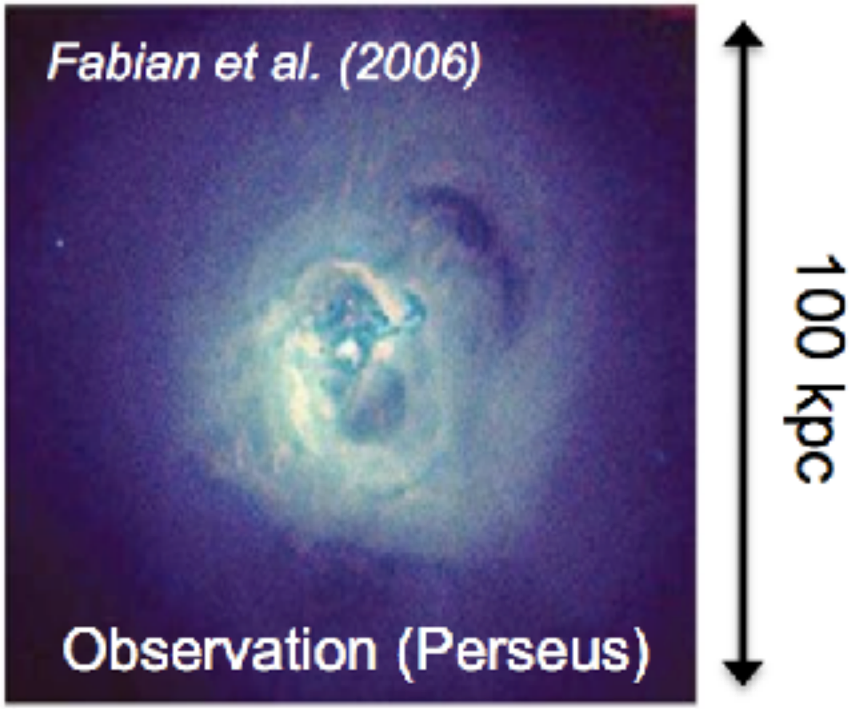
Maximal feedback scenario: $\dot{E}_{inj} = \dot{\rho}_* \eta_{SN} 10^{50}\ \mathrm{erg/M_\odot} \qquad t_{diss} \simeq 10\ \mathrm{Myr}$

We mimic slow dissipation of non-thermal energy using delayed cooling for the thermal energy:

$$\rho \frac{D\epsilon_{thermal}}{Dt} = \dot{E}_{inj} - P_{thermal} \nabla \cdot \mathbf{v} - n_H^2 \Lambda \quad \text{with} \quad \Lambda = 0 \text{ if } \sigma_{turb} > 10\ \mathrm{km/s}$$

Implementing AGN feedback





A simple model for SMBH growth and feedback

Numerical implementation in cosmological simulations: Sijacki et al. 2007; Booth & Schaye 2010, Dubois et al. 2010, Teyssier et al. 2011 and others.

In high density regions with « some criteria », we create seed black holes of a given seed mass, say 10^5 M_{sol}.

Accretion is described using several possible models:

Bondi-Hoyle accretion $$\dot{M}_{\rm BH} = \alpha_{\rm boost} \frac{4\pi G^2 M_{\rm BH}^2 \rho}{(c_s^2 + u^2)^{3/2}}$$

Eddington-limited $$\dot{M}_{\rm ED} = \frac{4\pi G M_{\rm BH} m_{\rm p}}{\epsilon_r \sigma_T c}$$

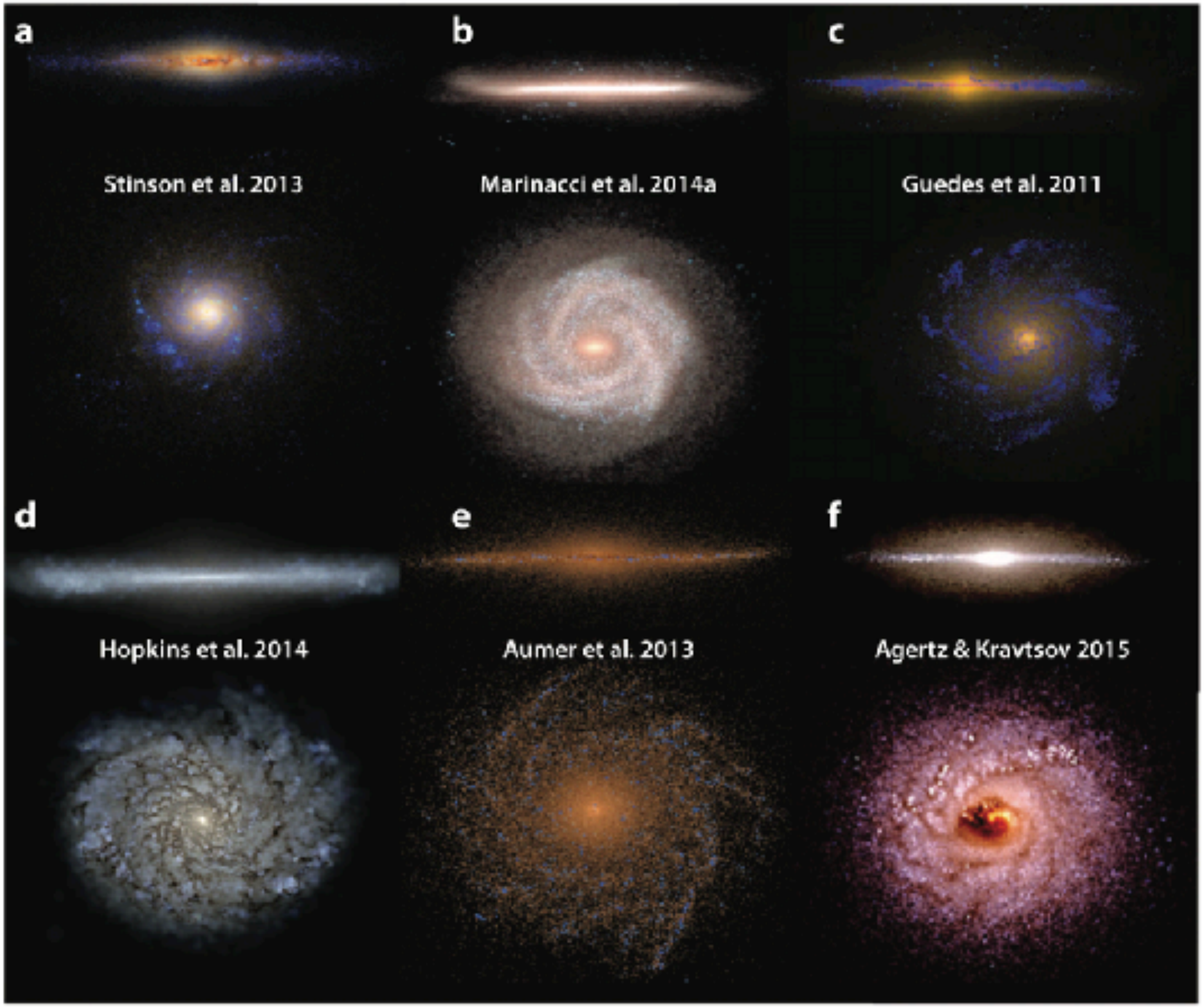
Torque-limited $$\dot{M}_{\rm TA} = 3\pi\delta\Sigma\frac{c_s^2}{\Omega}$$

Feedback performed using either a thermal pulse $\Delta E = \epsilon_c \epsilon_r \dot{M}_{\rm acc} c^2 \Delta t.$

or momentum injection through a jet $\Delta P = \dot{M}_{\rm acc} v_{\rm jet} \Delta t$

Free parameters are calibrated on the M_BH-sigma and M_star-M_halo relations.

Current galaxy formation simulations



Naab & Ostriker (2017)