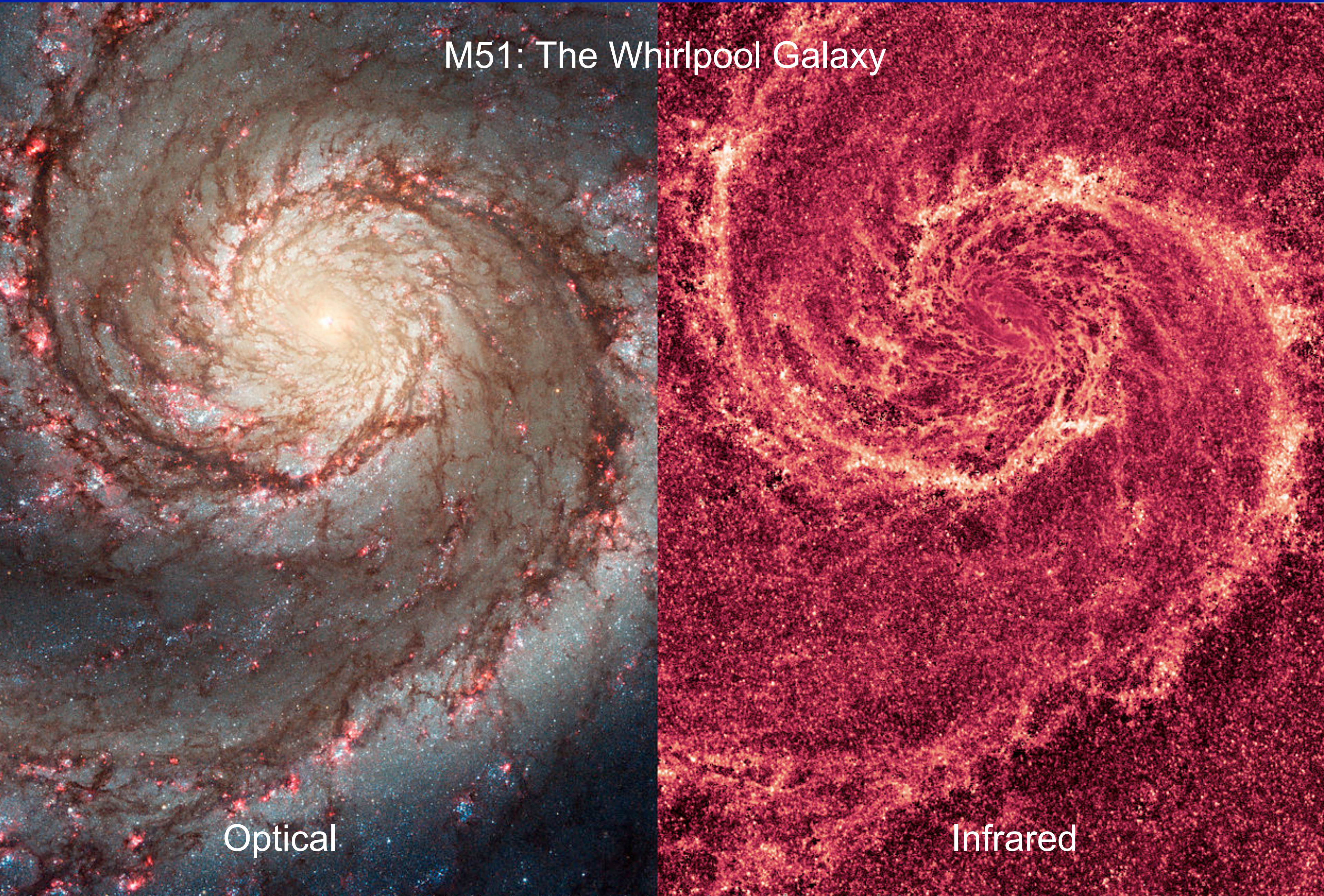
Grid-based Hydrodynamics

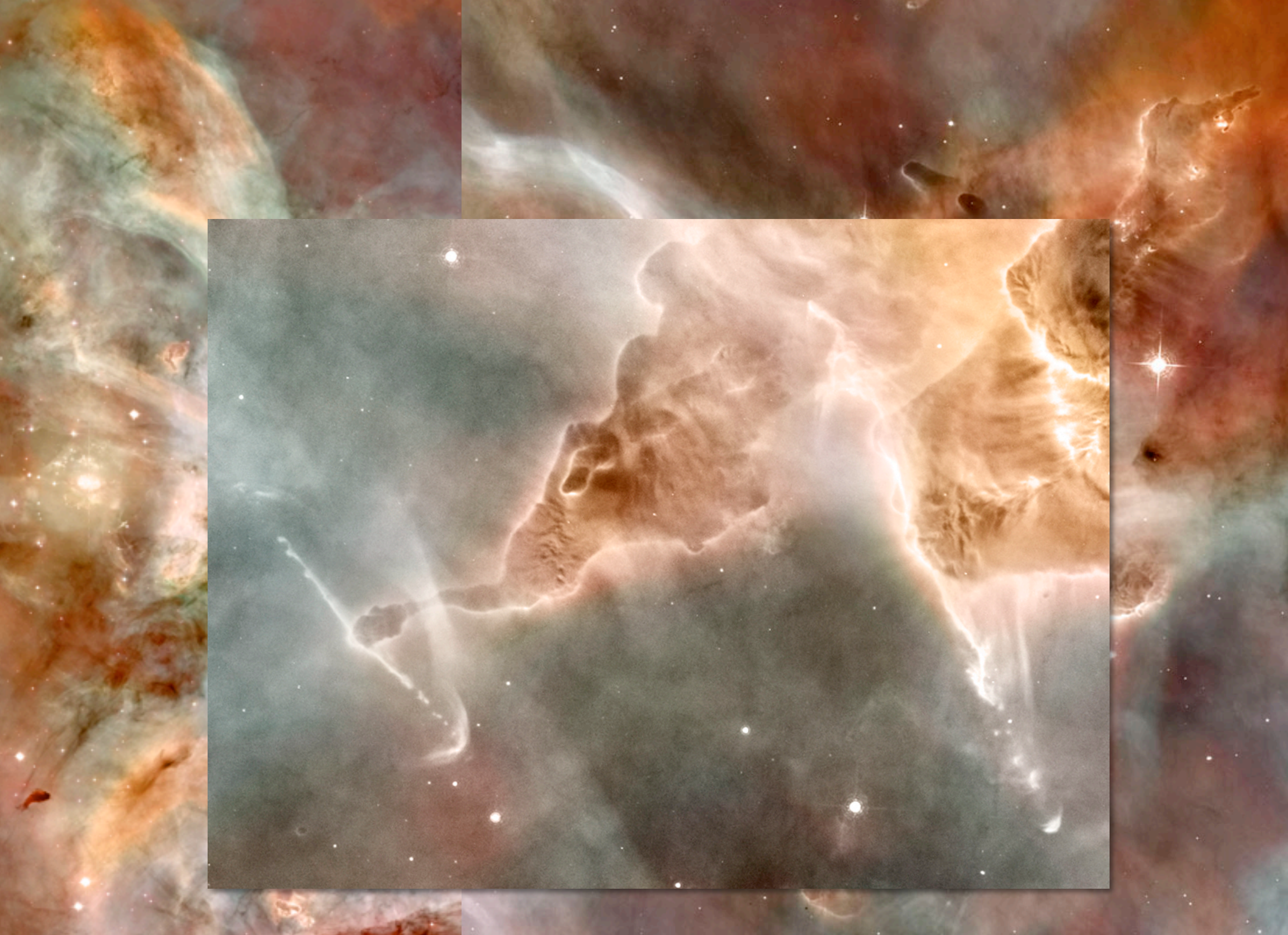
Christoph Federrath

- Complex fluid dynamics (equations are non-linear, 3D)
- Complex physics: turbulence, gravity, radiation, magnetic fields, etc.
- Large spatial and temporal scales involved

Star Formation



Infrared: NASA, ESA, M. Regan & B. Whitmore (STScI), & R. Chandar (U. Toledo);Optical: NASA, ESA, S. Beckwith (STScI), & the Hubble Heritage Team (STScI/AURA)

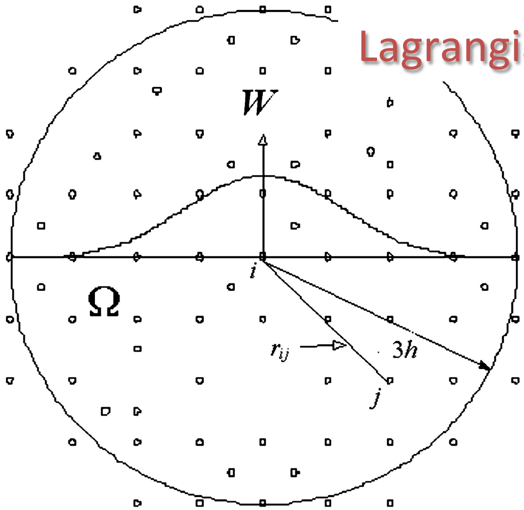


Carina Nebula, NASA, ESA, N. Smith (University of California, Berkeley),The Hubble Heritage Team (STScI/AURA), and NOAO/AURA/NSF

Fundamentals of SPH and grid

Smoothed Particle Hydrodynamics (SPH)
(Lucy 1977; Gingold & Monaghan 1977)

Lagrangian



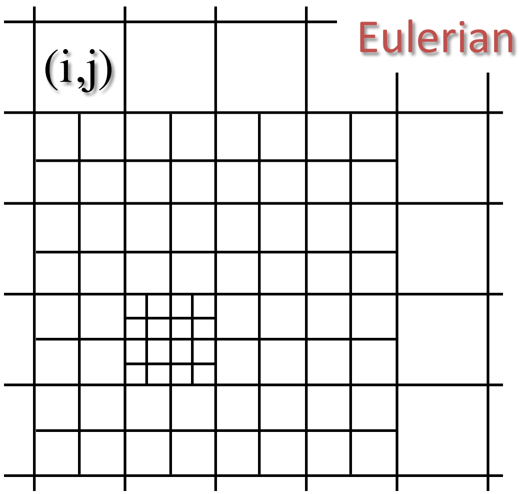
$$\rho(\mathbf{r}) = \sum_b m_b W(\mathbf{r} - \mathbf{r}_b, h)$$

$$\nabla A(\mathbf{r}) = \sum_b m_b \frac{A_b}{\rho_b} \nabla W(\mathbf{r} - \mathbf{r}_b, h)$$

$$W(x, h) = \frac{1}{h\sqrt{\pi}} e^{-(x^2/h^2)}$$

Adaptive Mesh Refinement (AMR)
(Berger & Collela 1989)

Eulerian



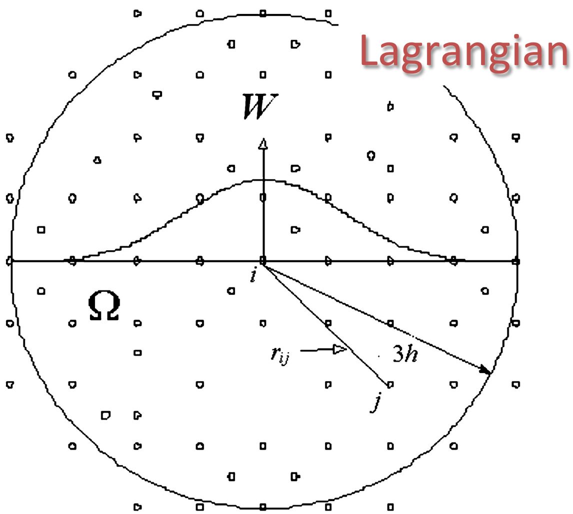
- Hydro variables are averages in cells
- Compute fluxes through cell faces
- Simple data structure: indexing
- Finite Volume vs. Finite Difference

R. Leveque: „Nonlinear Conservation Laws and Finite Volume Methods for Astrophysical Fluid Flow“

Fundamentals of SPH and grid

Smoothed Particle Hydrodynamics (SPH)
(Lucy 1977; Gingold & Monaghan 1977)

Lagrangian



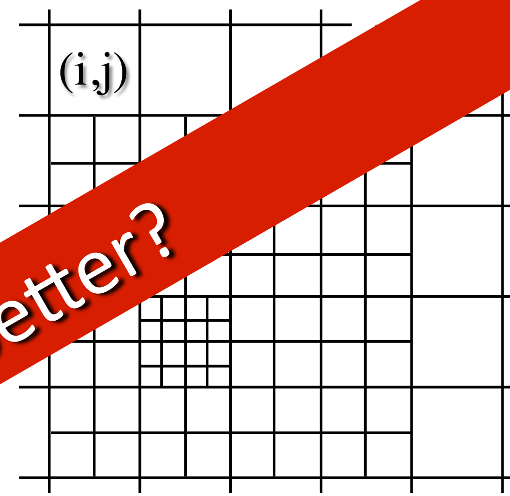
$$\rho(\mathbf{r}) = \sum_b m_b W(\mathbf{r} - \mathbf{r} \ldots$$

$$\nabla A(\mathbf{r}) \ldots \frac{\ldots}{\rho_b} \nabla W(\mathbf{r} - \mathbf{r}_b, h)$$

$$W(\ldots, h) = \frac{1}{h\sqrt{\pi}} e^{-(x^2/h^2)}$$

Which method is better?

Adaptive Mesh Refinement (AMR)
(Berger & Collela 1989)



- Hydro variables are averages in cells
- Compute fluxes through cell faces
- Simple data structure: indexing
- Finite Volume vs. Finite Difference

R. Leveque: „Nonlinear Conservation Laws and Finite Volume Methods for Astrophysical Fluid Flow“

Comparison of SPH and grid in supersonic turbulence

TWO REGIMES OF TURBULENT FRAGMENTATION AND THE STELLAR INITIAL MASS FUNCTION FROM PRIMORDIAL TO PRESENT-DAY STAR FORMATION

Paolo Padoan,[1] Åke Nordlund,[2] Alexei G. Kritsuk,[1] Michael L. Norman,[1] and Pak Shing Li[3]
Received 2006 October 16; accepted 2007 February 16

Their conclusion:

" SPH simulations of large scale star formation to date fail in all three fronts: numerical diffusivity, numerical resolution, and presence of magnetic fields. This should cast serious doubts on the value of comparing predictions based on SPH simulations with observational data (see also Agertz et al. 2006). "

Driven turbulence comparison of SPH and grid

Motivation (role of supersonic turbulence for star formation)

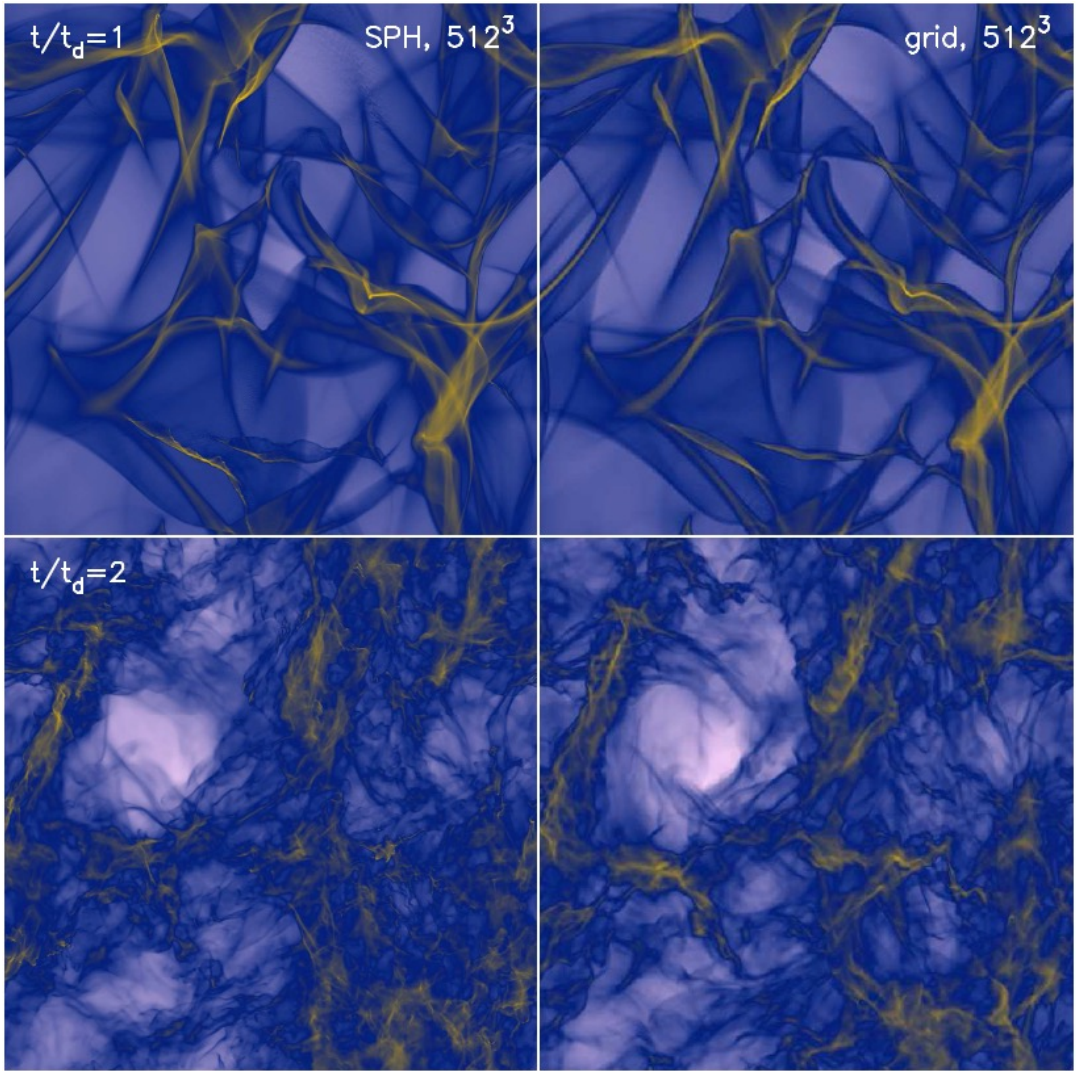
Setup (Phantom and FLASH):

1. Same initial conditions: uniform density, zero velocities
2. Same turbulence forcing!
3. Driven to Mach number 10
4. Resolutions: 128^3, 256^3 and 512^3 (134,217,728) both grid and SPH

(Price & Federrath 2010, MNRAS 406, 1659)

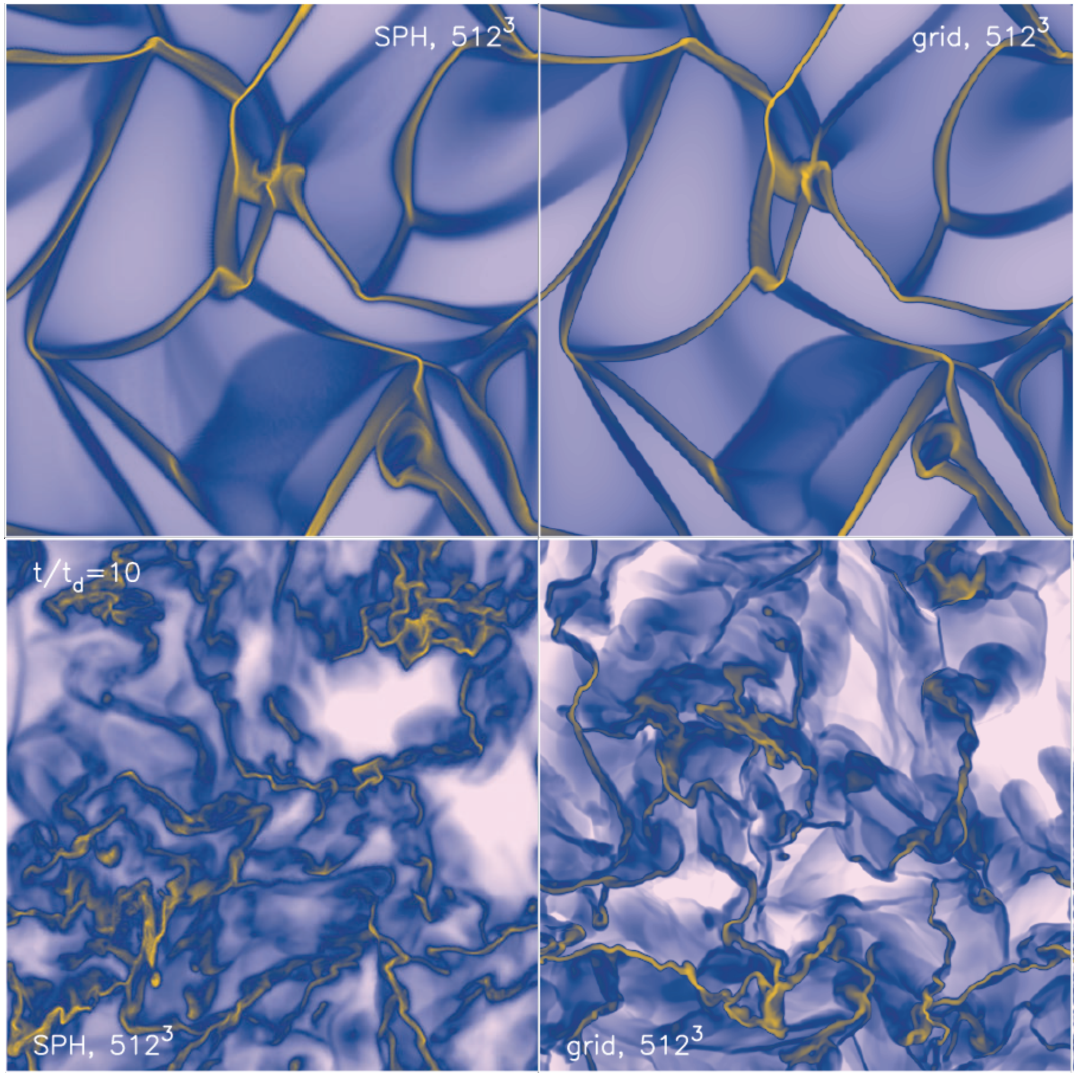
Driven turbulence comparison of SPH and grid

Phantom Column density FLASH



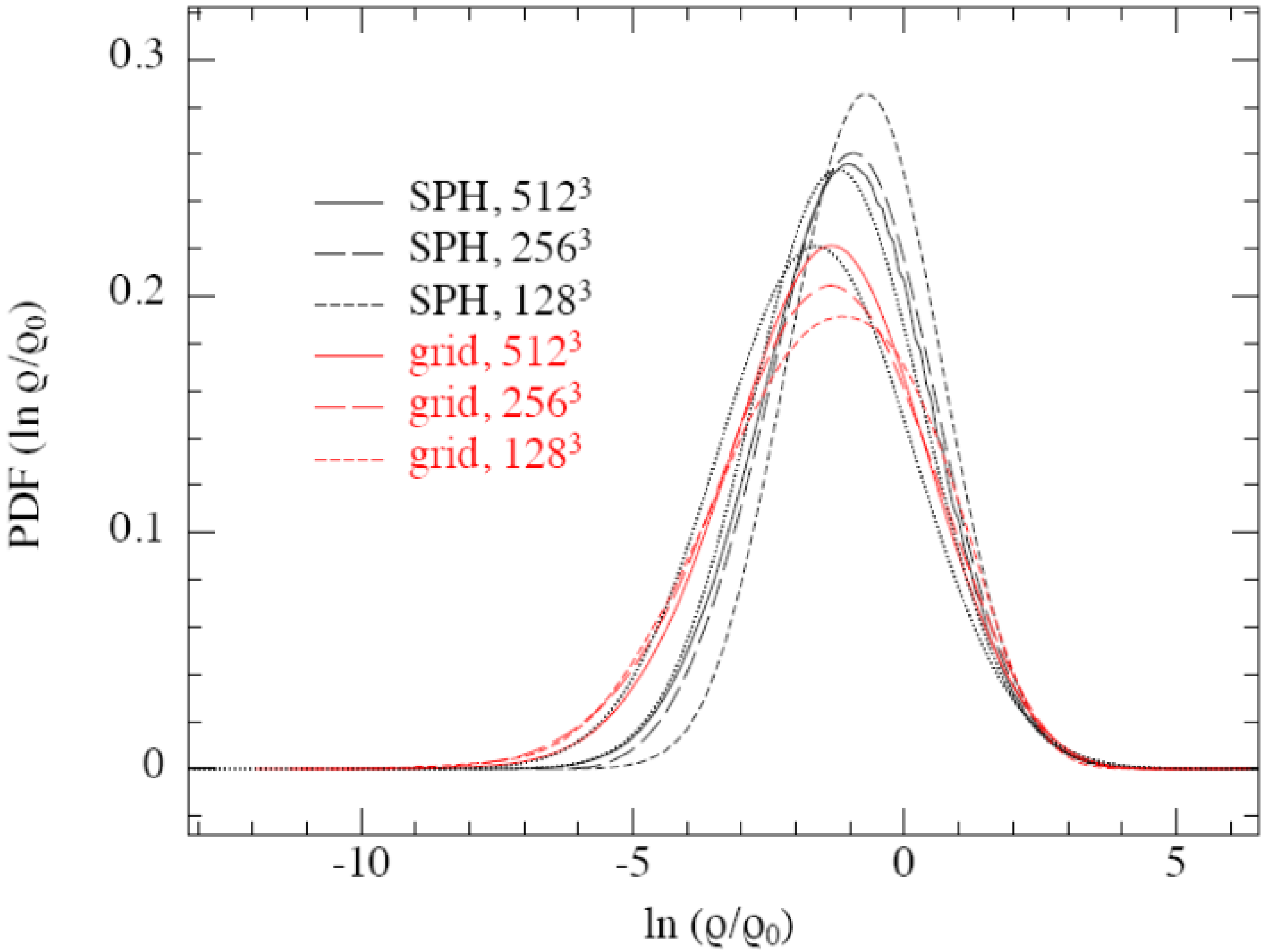
Driven turbulence comparison of SPH and grid

Phantom | Slices of density | FLASH



Driven turbulence comparison of SPH and grid

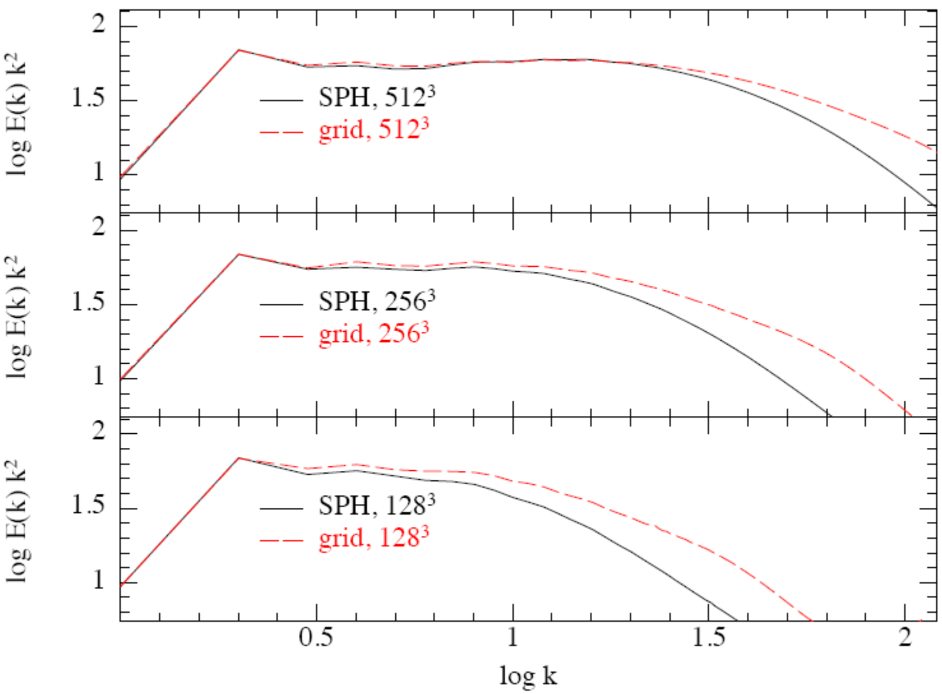
Density Probability Distribution Function (PDF):



PDFs converge with higher resolution

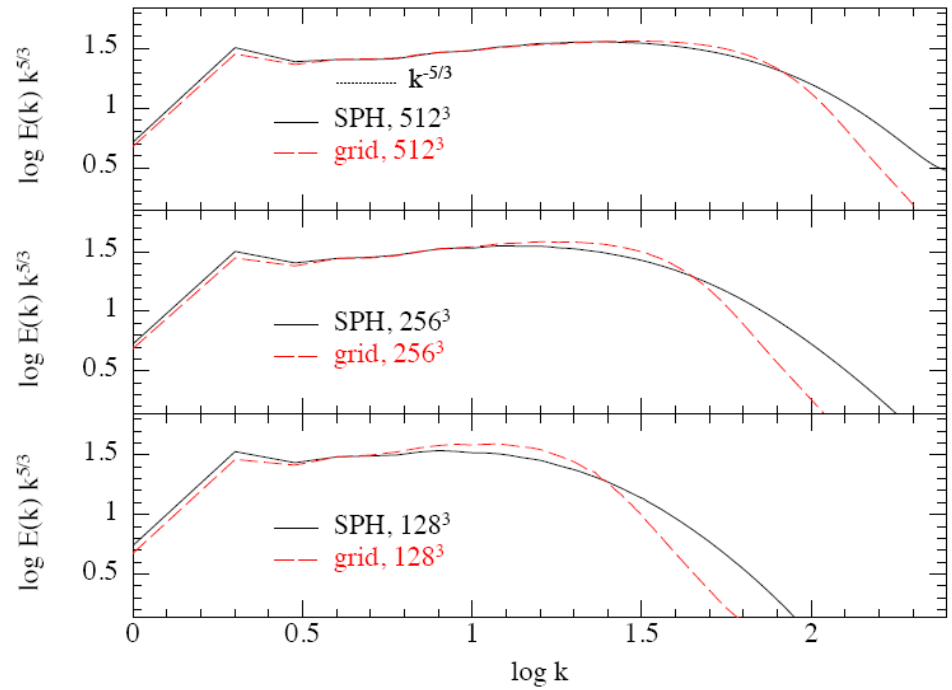
Driven turbulence comparison of SPH and grid

Velocity spectra, v (VOLUME-weighted)



Grid code less dissipative

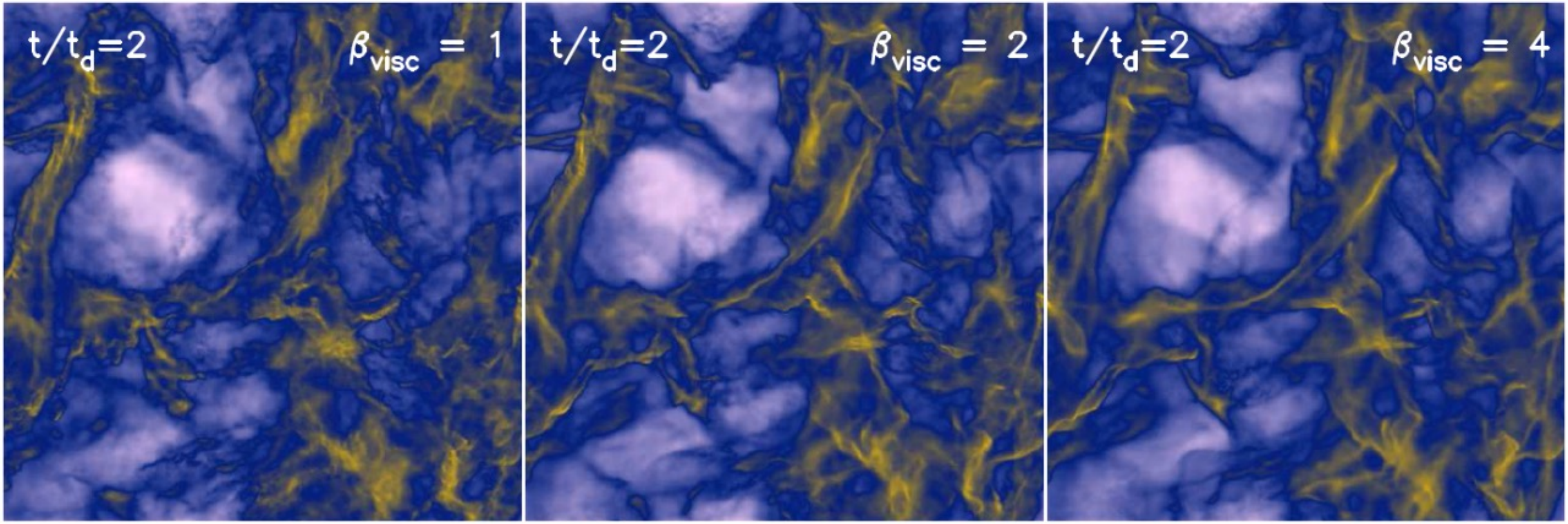
Velocity spectra, $\rho^{1/3}v$ (DENSITY-weighted)



SPH code slightly less dissipative

Driven turbulence comparison of SPH and grid

Influence of β-viscosity in SPH on the modelling of strong shocks



β=1 β=2 β=4

Particle interpenetration for β<4

Driven turbulence comparison of SPH and grid

Conclusion

(Price & Federrath 2010, MNRAS 406, 1659)

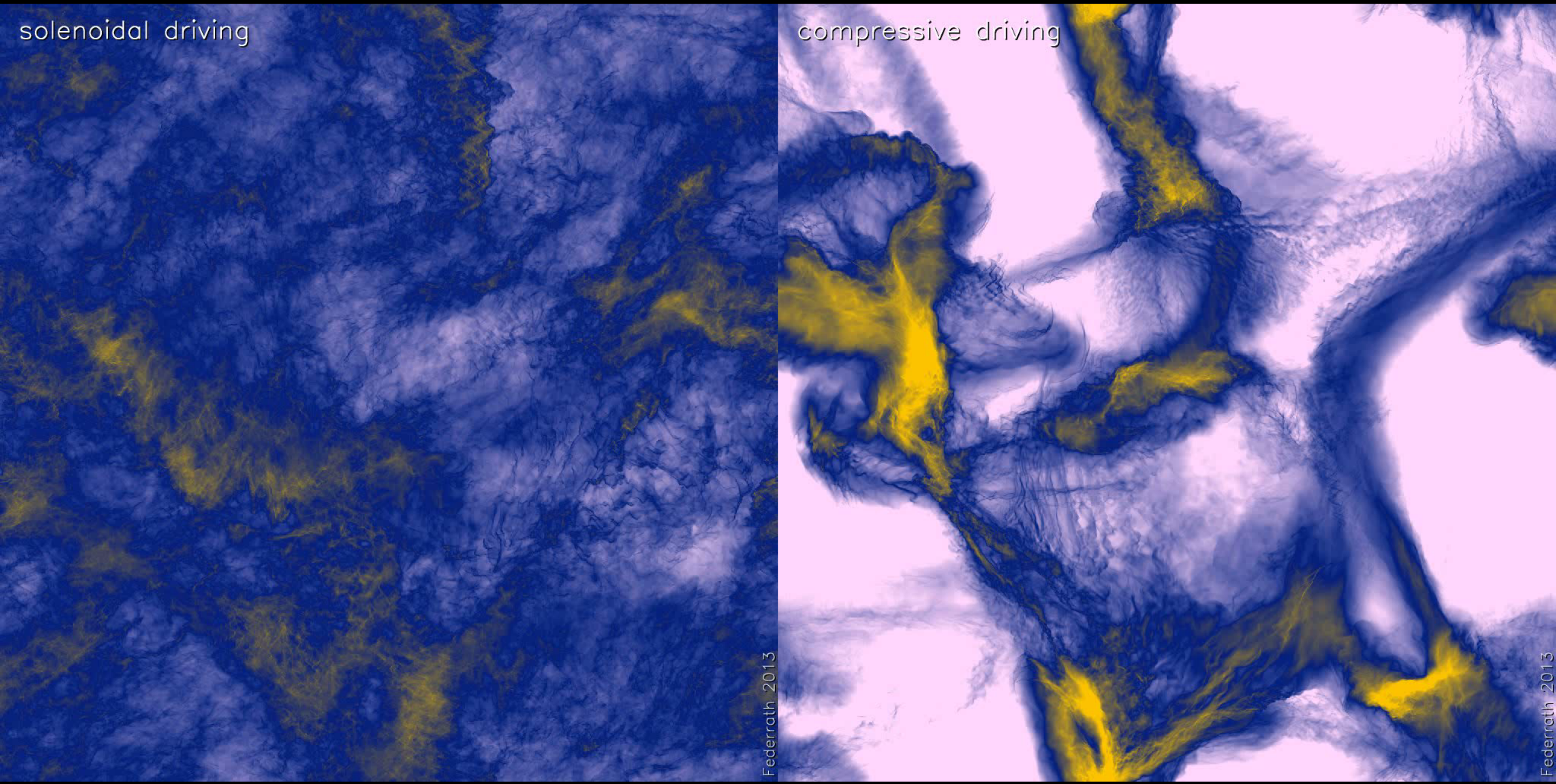
Convergence of SPH and grid

Computational time pure hydro (no gravity):

FLASH grid about 20 times faster than Phantom SPH

Hydrodynamical Turbulence

World's largest simulations of turbulence using 4096^3 grid cells



(Federrath 2013, MNRAS 436, 1245)

The basics of grid-based hydrodynamics

1. Introduction
2. Equations of hydrodynamics
3. Advection
4. Flux conservation and flux limiters
5. Conservative grid-based hydrodynamics
6. Basics of Riemann problem -> Riemann solvers
7. Adaptive-mesh refinement and sink particles

Lecture based on a lecture given by Kees Dullemond, 2009/2010, Heidelberg

Literature: Randall J. LeVeque, "Finite Volume Methods for Hyperbolic Problems" (Cambridge Texts in Applied Mathematics)

Advection test, IDL code

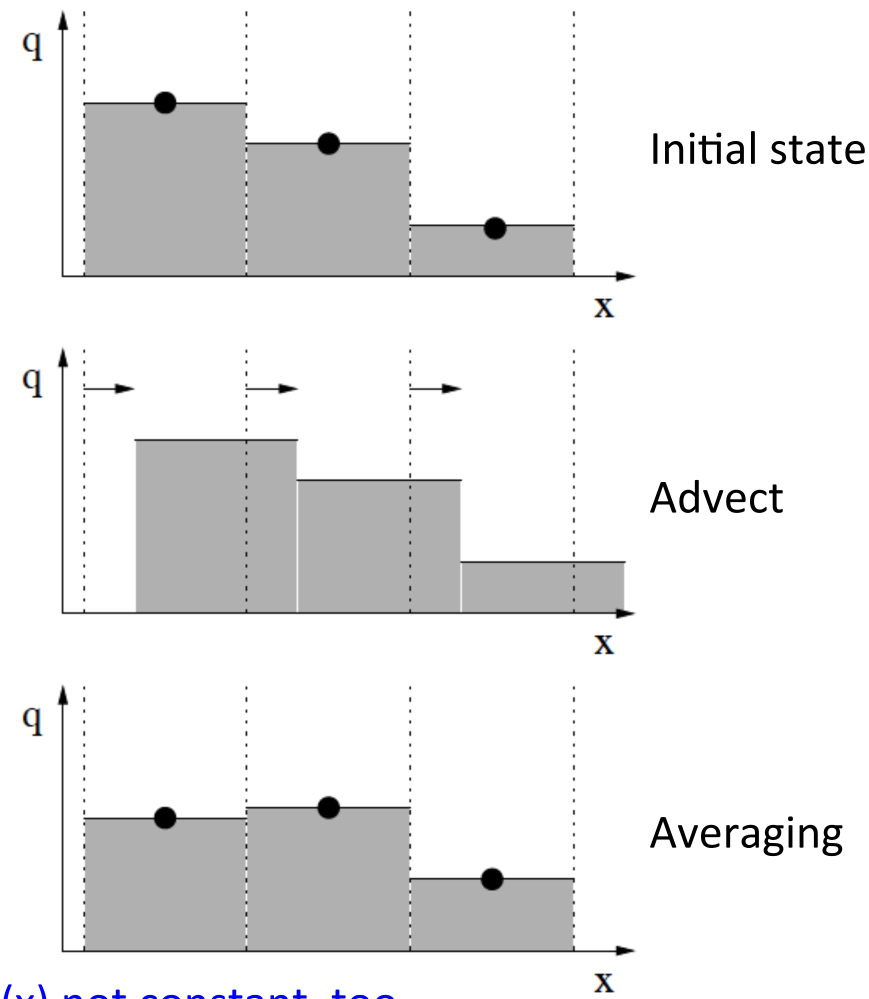
Donor-cell advection:

Piecewise constant subgrid model:

$$\tilde{q}^{n+1/2}_{i+1/2} = \begin{cases} q_i^n & \text{for } u_{i+1/2} > 0 \\ q_{i+1}^n & \text{for } u_{i+1/2} < 0 \end{cases}$$

Flux:

$$f^{n+1/2}_{i+1/2} = \begin{cases} u_{i+1/2}\, q_i^n & \text{for } u_{i+1/2} > 0 \\ u_{i+1/2}\, q_{i+1}^n & \text{for } u_{i+1/2} < 0 \end{cases}$$



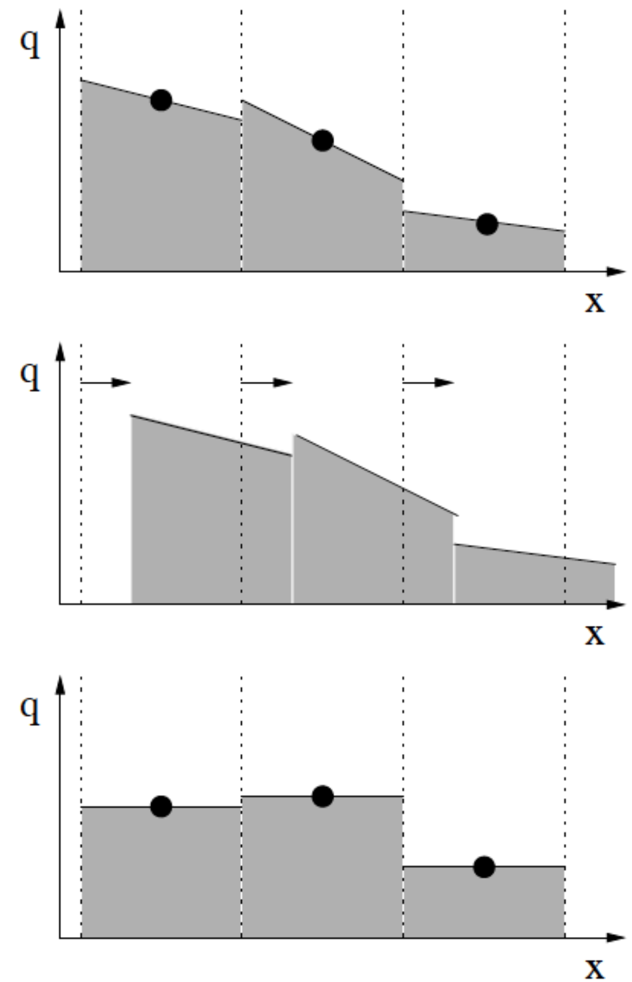
Like upwind scheme, but works for u(x) not constant, too.

Piecewise linear subgrid model for flux:

- Donor-cell is quite diffusive ->
  Use higher-order subgrid model

$$q(x, t = t_n) = q_i^n + \underbrace{\sigma_i^n}_{\text{(slope)}}(x - x_i)$$

Choice of slope

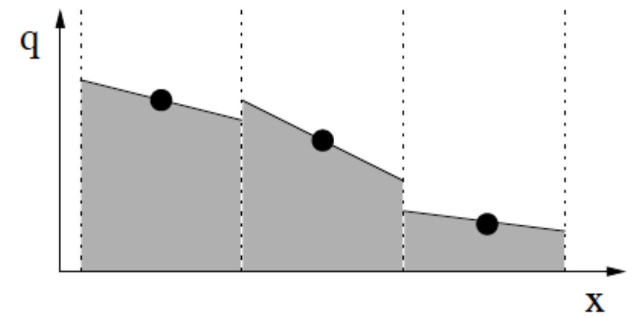


„MUSCL (Monotonic Upwind-centered Scheme for Consveration Laws)“

Flux-conserving grid-based hydrodynamics

Piecewise linear subgrid model for flux:

- Donor-cell is quite diffusive ->
  Use higher-order subgrid model



$$q(x, t = t_n) = q_i^n + \sigma_i^n (x - x_i)$$

Different slope choices:

Centered slope:	$\sigma_i^n = \frac{q_{i+1}^n - q_{i-1}^n}{2\Delta x}$	(Fromm's method)
Upwind slope:	$\sigma_i^n = \frac{q_i^n - q_{i-1}^n}{\Delta x}$	(Beam-Warming method)
Downwind slope:	$\sigma_i^n = \frac{q_{i+1}^n - q_i^n}{\Delta x}$	(Lax-Wendroff method)

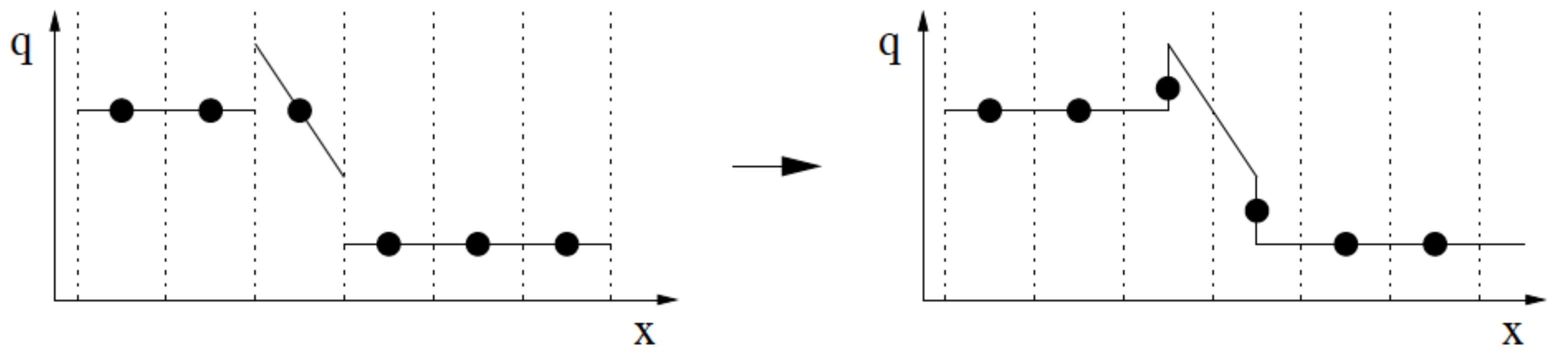
Higher-order now, but beware oscillations

„MUSCL (Monotonic Upwind-centered Scheme for Consveration Laws)"

Flux-conserving grid-based hydrodynamics

Piecewise linear subgrid model for flux:

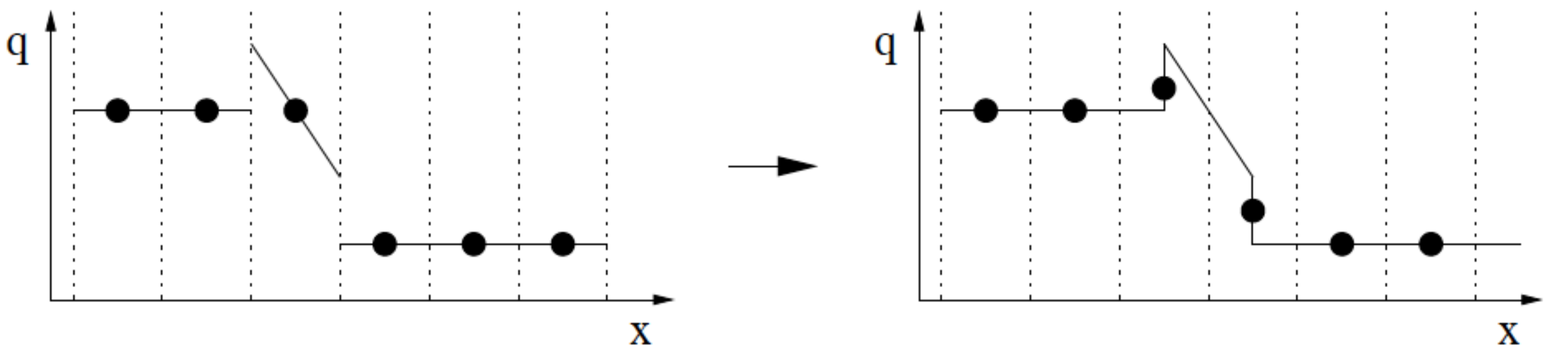
- can produce overshoots



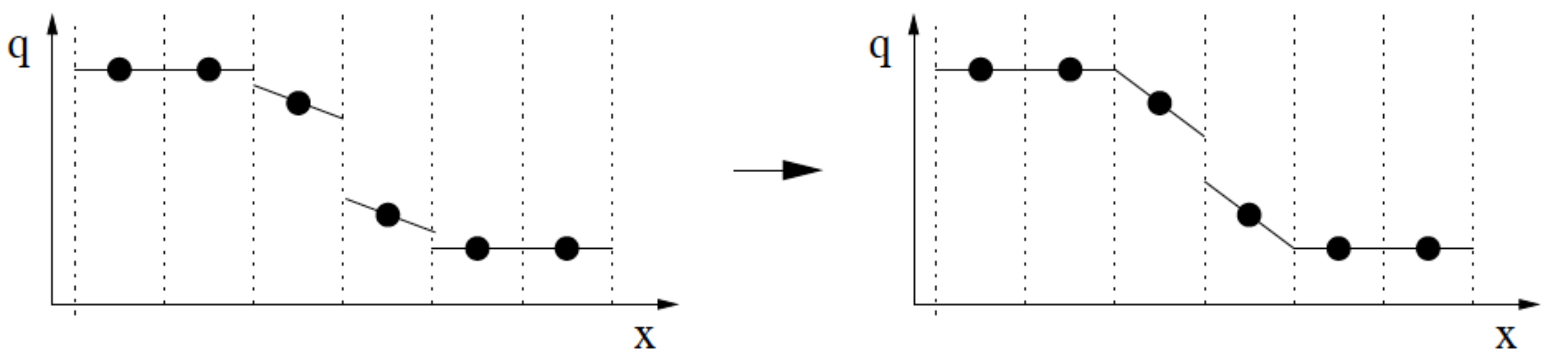
Flux-conserving grid-based hydrodynamics

Piecewise linear subgrid model for flux:

- can produce overshoots



Fix: slope limiters -> flux limiters



minmod slope superbee slope

Flux-conserving grid-based hydrodynamics

Flux limiters:

- Normal flux:

$$f_{i+1/2}^{n+1/2} = \begin{cases} u_{i+1/2}\, q_i^n & \text{for } u_{i+1/2} > 0 \\ u_{i+1/2}\, q_{i+1}^n & \text{for } u_{i+1/2} < 0 \end{cases}$$

- Flux correction due to limiter Φ_i

$$\frac{1}{2}\,|u_i|\left(1 - |u_i|\,\frac{\Delta t}{\Delta x}\right)(q_i - q_{i-1})\,\Phi_i$$

Flux-conserving grid-based hydrodynamics

Flux limiters:

- Flux correction due to limiter Φ_i : $\frac{1}{2}|u_i|\left(1-|u_i|\frac{\Delta t}{\Delta x}\right)(q_i - q_{i-1})\,\Phi_i$

$$r^n_{i-1/2} = \begin{cases} \frac{q^n_{i-1}-q^n_{i-2}}{q^n_i - q^n_{i-1}} & \text{for } u_{i-1/2} \geq 0 \\ \frac{q^n_{i+1}-q^n_{i}}{q^n_i - q^n_{i-1}} & \text{for } u_{i-1/2} \leq 0 \end{cases}$$

$$\begin{aligned}
\text{donor-cell}: &\quad \phi(r) = 0 \\
\text{Lax-Wendroff}: &\quad \phi(r) = 1 \\
\text{Beam-Warming}: &\quad \phi(r) = r \\
\text{Fromm}: &\quad \phi(r) = \tfrac{1}{2}(1+r)
\end{aligned}$$

linear

---

non-linear

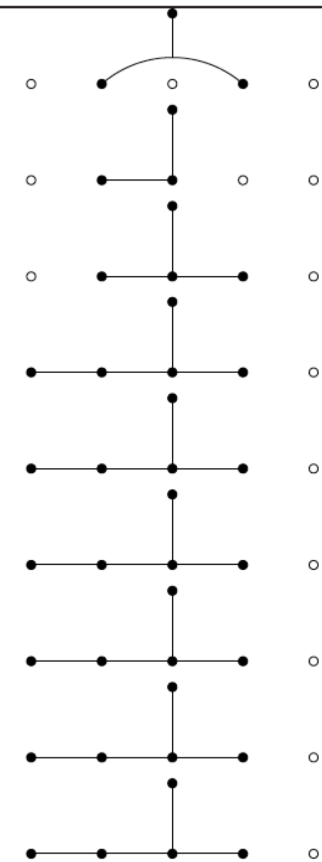
$$\begin{aligned}
\text{minmod}: &\quad \phi(r) = \text{minmod}(1, r) \\
\text{superbee}: &\quad \phi(r) = \max(0, \min(1, 2r), \min(2, r)) \\
\text{MC}: &\quad \phi(r) = \max(0, \min((1+r)/2, 2, 2r)) \\
\text{van Leer}: &\quad \phi(r) = (r + |r|)/(1 + |r|)
\end{aligned}$$

Flux-conserving grid-based hydrodynamics

Flux limiters:

- Flux correction due to limiter Φ_i : $\frac{1}{2}|u_i|\left(1-|u_i|\frac{\Delta t}{\Delta x}\right)(q_i - q_{i-1})\,\Phi_i$

Name	Order	Lin?	Stable?	TVD?	Stencil
Two-point symmetric	1	lin	-	-	
Upwind / Donor-cell	1	lin	+	+	
Lax-Wendroff	2	lin	+	-	
Beam-warming	2	lin	+	-	
Fromm	2	lin	+	-	
Minmod	2/1	non-lin	+	+	
Superbee	2/1	non-lin	+	+	
MC	2/1	non-lin	+	+	
van Leer	2/1	non-lin	+	+	



The basics of grid-based hydrodynamics

1. Introduction
2. Equations of hydrodynamics
3. Advection
4. Flux conservation and flux limiters
5. Conservative grid-based hydrodynamics
6. Basics of Riemann problem -> Riemann solvers
7. Adaptive-mesh refinement and sink particles

construction of classic 1D hydro solver

$$
\begin{aligned}
\partial_t \rho + \nabla \cdot (\rho \vec{u}) &= 0 \\
\partial_t (\rho \vec{u}) + \nabla \cdot (\rho \vec{u} \vec{u}) &= -\nabla P \\
\partial_t (\rho e_{\mathrm{tot}}) + \nabla \cdot (\rho e_{\mathrm{tot}} \vec{u}) &= -\nabla \cdot (P \vec{u})
\end{aligned}
$$

Source terms

HYDRO STEP:

1. Use standard advection scheme to advect $\rho,\ \rho\vec{u},\ \rho e_{\mathrm{tot}}$ with zero source

2. Treat source terms separately (operator splitting)

Advantage of operator splitting: source terms cancel exactly (not inside the advection)

construction of classic 1D hydro solver

Code for hydro step; test with interacting sound waves

The basics of grid-based hydrodynamics

1. Introduction
2. Equations of hydrodynamics
3. Advection
4. Flux conservation and flux limiters
5. Conservative grid-based hydrodynamics
6. Basics of Riemann problem -> Riemann solvers
7. Adaptive-mesh refinement and sink particles

Treating shocks – Riemann solvers

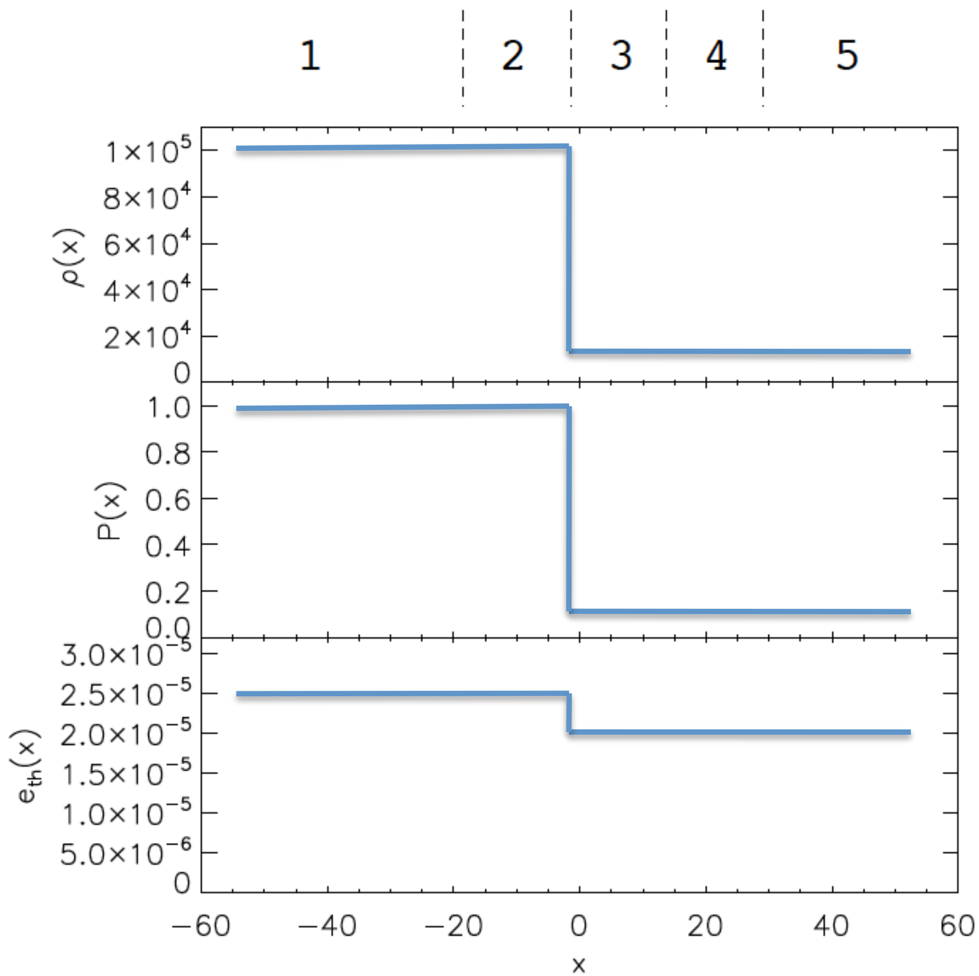
- Code treats smooth flows fairly well
- But shocks are common in astrophysics (e.g., interstellar medium)
- Flow speed is supersonic, i.e., $u > c_s$
- Need to solve Riemann problem
- Leads to Riemann solvers (e.g., Piecewise Parabolic Method)

Collela & Woodward (1984)

Difference to previous solver:
pressure terms are included in the advection

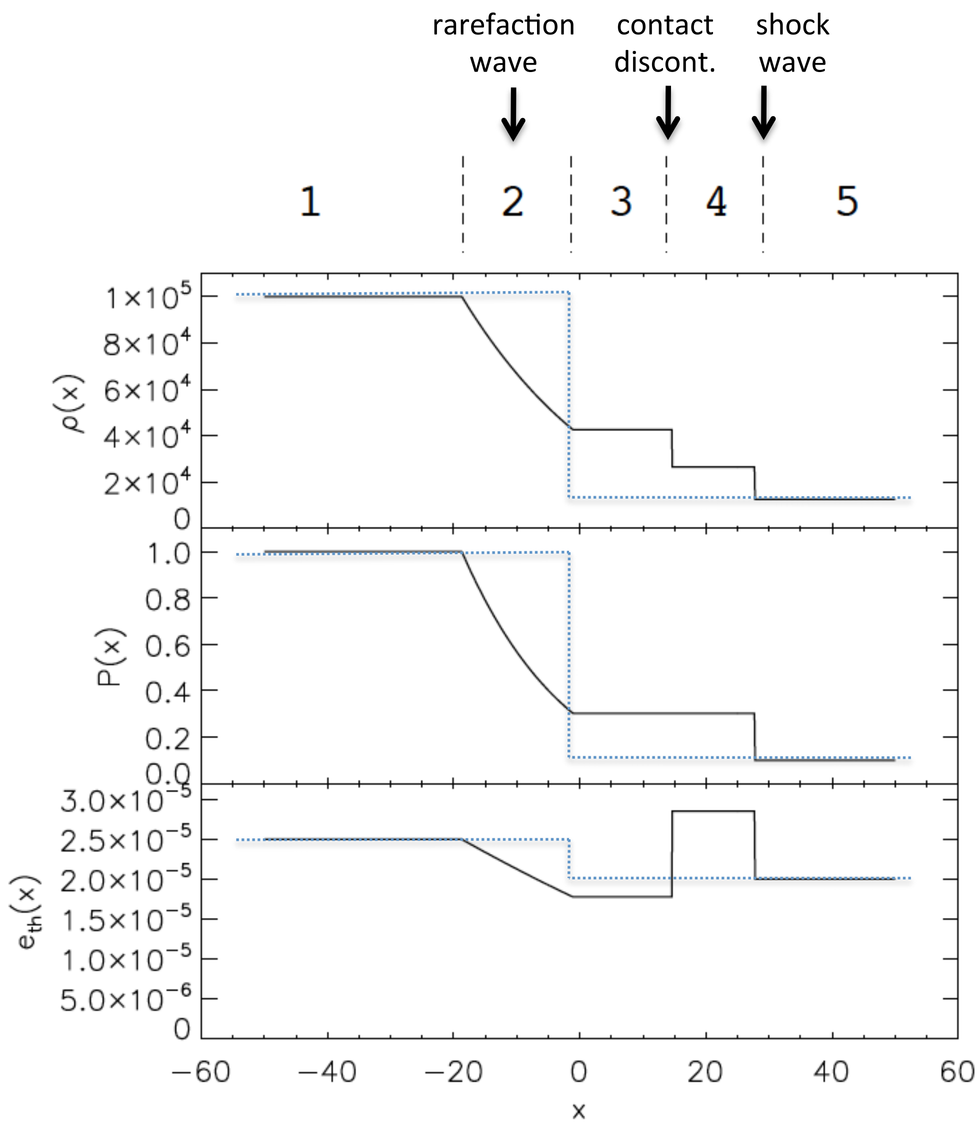
Sod shocktube test: $\rho_l = 10^5, P_l = 1$ $\rho_r = 1.25 \times 10^4$ and $P_r = 0.1$

(Sod 1978)

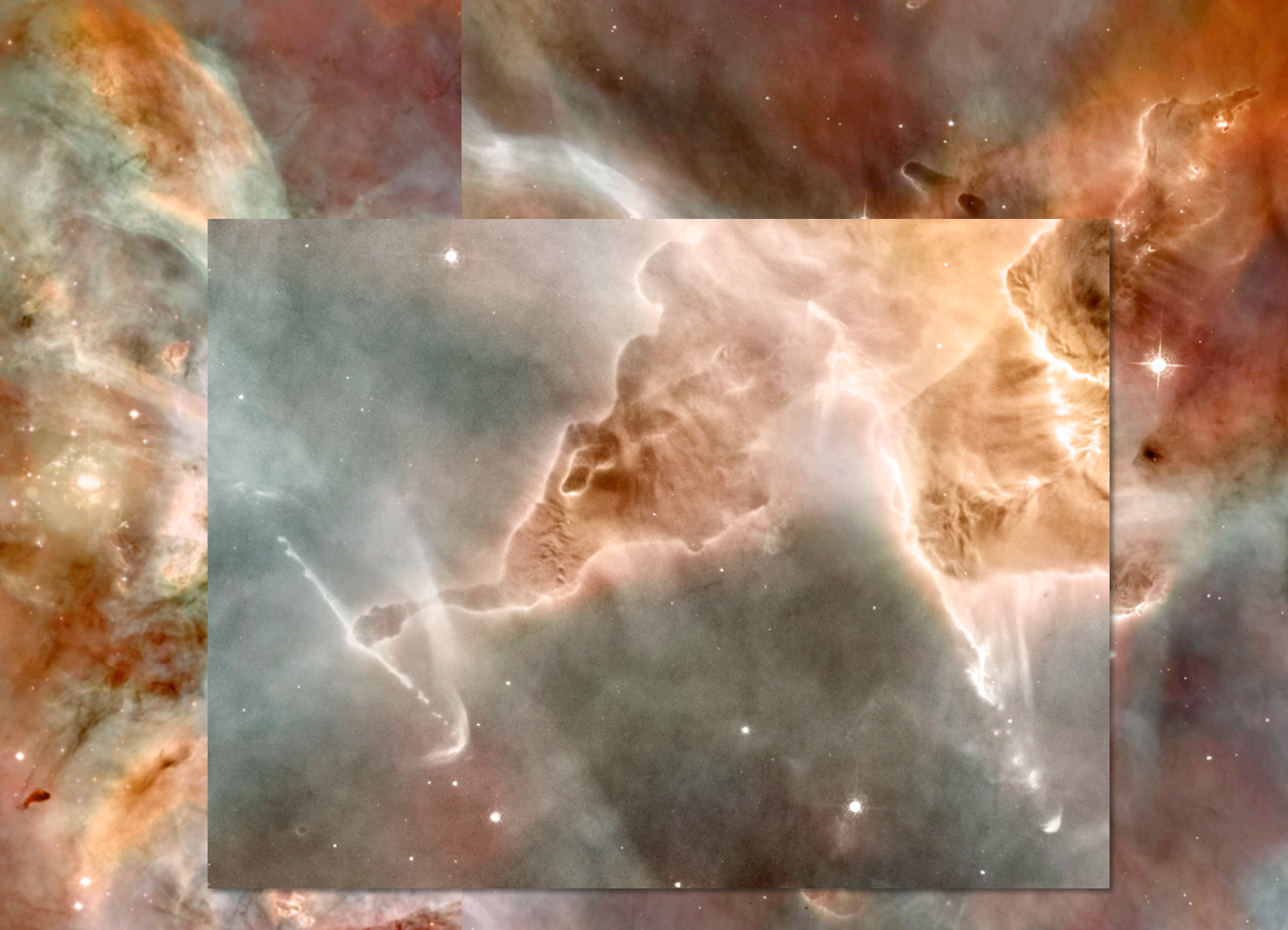


Sod shocktube test: $\rho_l = 10^5, P_l = 1$ $\rho_r = 1.25 \times 10^4$ and $P_r = 0.1$

(Sod 1978)



Sod shocktube test in 1D and 2D with AMR



Carina Nebula, NASA, ESA, N. Smith (University of California, Berkeley),The Hubble Heritage Team (STScI/AURA), and NOAO/AURA/NSF

Why sink particles?

- Quantify fragmentation and accretion
- Prevent code from stalling

$$t_{\rm ff} = \left(\frac{3\pi}{32G\rho}\right)^{1/2}$$

Why sink particles?

- Quantify fragmentation and accretion
- Prevent code from stalling

$$t_{\rm ff} = \left(\frac{3\pi}{32 G\rho} \right)^{1/2}$$

- Resolve fragmentation scale

$$\lambda_{\rm J} = \left(\frac{\pi c_{\rm s}^2}{G\rho} \right)^{1/2}$$

$$M_{\rm J}(\rho) = \frac{4\pi}{3} \left(\frac{\lambda_{\rm J}(\rho)}{2} \right)^3 \rho$$

Truelove et al. (1997)

Bate & Burkert (1997)

Why sink particles?

- Quantify fragmentation and accretion
- Prevent code from stalling

$$t_{\mathrm{ff}} = \left(\frac{\cdots}{32 G \rho} \right)^{1/2}$$

- Resolve fragmentati[illegible]

$$\lambda_{\mathrm{J}} \cdots \left(\frac{\cdots_{\mathrm{s}}}{G \rho} \right)^{1/2} \qquad M_{\mathrm{J}}(\rho) = \frac{4\pi}{3} \left(\frac{\lambda_{\mathrm{J}}(\rho)}{2} \right)^{3} \rho$$

Truelove et al. (1997) Bate & Burkert (1997)

Cut off runaway collapse

Cut off runaway collapse

$$\rho \quad > \quad \rho_{\rm res}$$

Sink Particles

Heat up gas

Cut off runaway collapse

$$\rho \; > \; \rho_{\text{res}}$$

Sink Particles

Heat up gas

1. **Problem**:
   Courant time step
   $$\min_{i,j,k}\left(\frac{\Delta x}{\max(|\mathbf{v}(i,\,j,\,k)|,\,c_{\mathrm{s}})}\right)$$
2. **Problem**:
   changes EOS,
   unless
   $$\rho_{\text{res}} > 10^{-14}\,\mathrm{g\,cm^{-3}}$$

Cut off runaway collapse

$$\rho \quad > \quad \rho_{\mathrm{res}}$$

Sink Particles

Problem:

Spurious sink creation in shocks that DON'T go into free fall collapse

e.g., isothermal shock:

(Density~Mach^2)

Heat up gas

1. **Problem**:

   Courant time step

$$\min_{i,j,k}\left(\frac{\Delta x}{\max(|\mathbf{v}(i,\,j,\,k)|,\,c_{\mathrm{s}})}\right)$$

2. **Problem**:

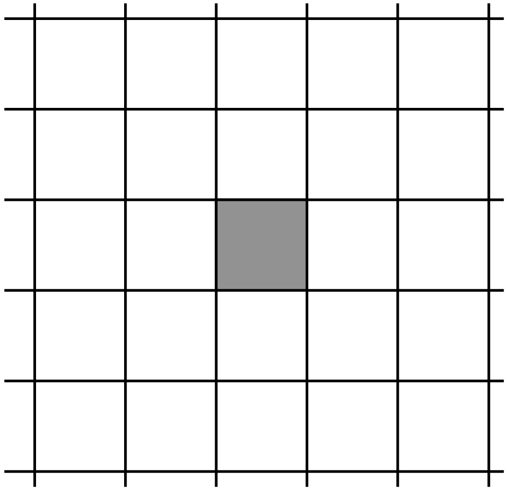
   changes EOS,
   unless

$$\rho_{\mathrm{res}} > 10^{-14}\,\mathrm{g\,cm^{-3}}$$

Sink particle implementation in FLASH

Collapse checks to avoid spurious sink creation

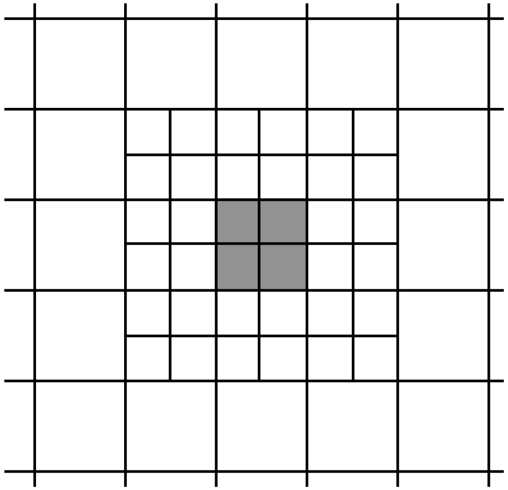
1. Cell exceeds density threshold, $\rho > \rho_{\mathrm{res}}$



Federrath, Banerjee, Clark, Klessen (2010, ApJ 713, 269)

Collapse checks to avoid spurious sink creation

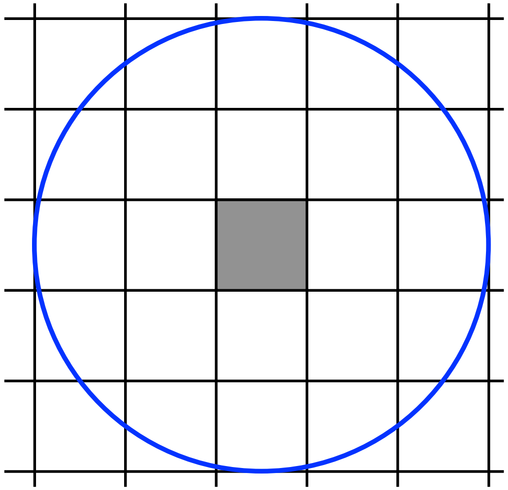
1. Cell exceeds density threshold, $\rho > \rho_{\mathrm{res}}$



Federrath, Banerjee, Clark, Klessen (2010, ApJ 713, 269)

Collapse checks to avoid spurious sink creation

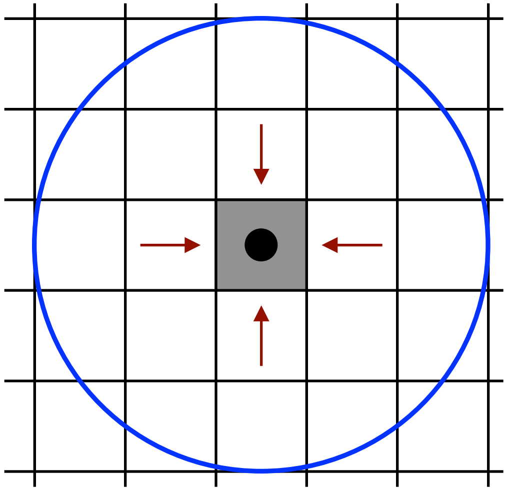
1. Cell exceeds density threshold, $\rho > \rho_{\mathrm{res}}$



Federrath, Banerjee, Clark, Klessen (2010, ApJ 713, 269)

Sink particle implementation in FLASH

Collapse checks to avoid spurious sink creation

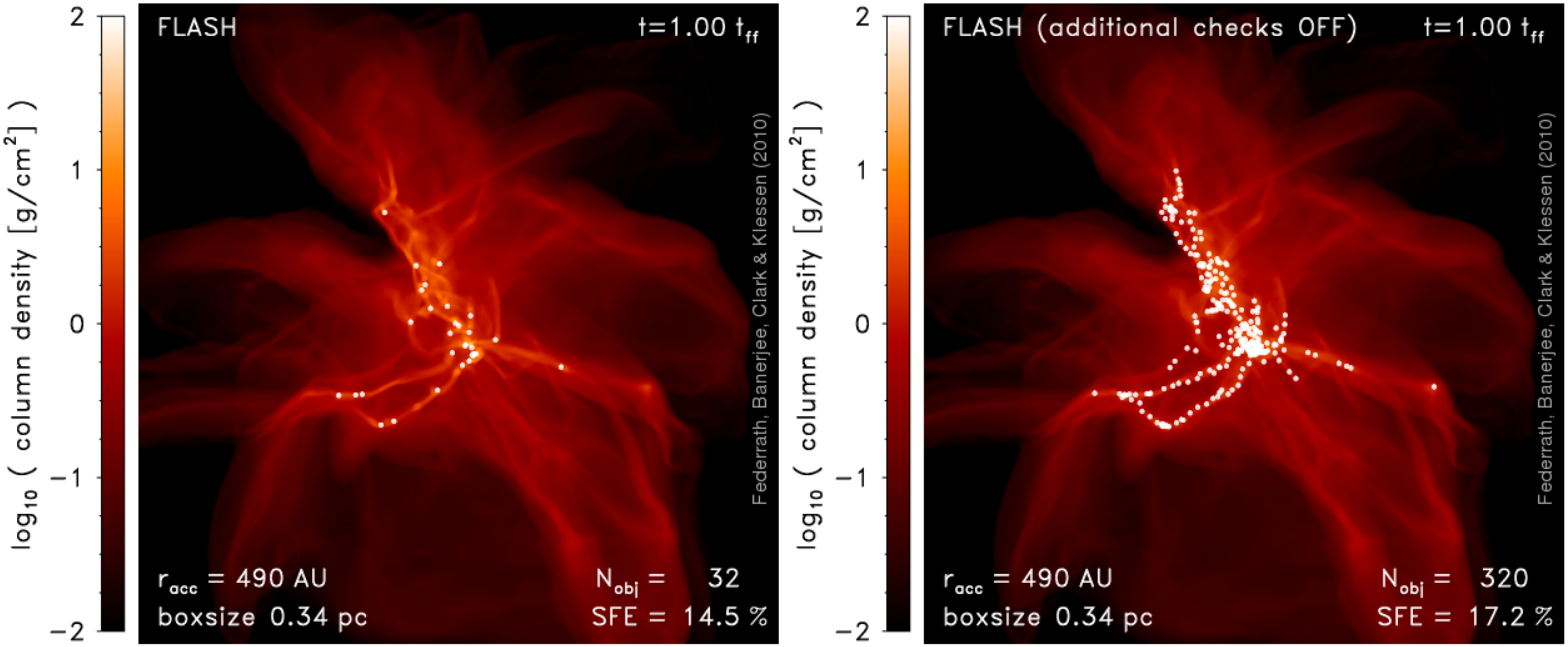


1. Cell exceeds density threshold, $\rho > \rho_{\rm res}$
2. Highest level of AMR
3. Converging toward the center
4. Central minimum in gravitational potential
5. Jeans unstable, $|E_{\rm grav}| > 2E_{\rm th}$
6. Bound, $E_{\rm grav} + E_{\rm th} + E_{\rm kin} + E_{\rm mag} < 0$
7. Not within the accretion radius of an existing sink particle

Federrath, Banerjee, Clark, Klessen (2010, ApJ 713, 269)

Sink particle implementation in FLASH

Movies available: http://www.ita.uni-heidelberg.de/~chfeder/pubs/sinks/sinks.shtml



all checks ON

$$\rho > \rho_{res}$$

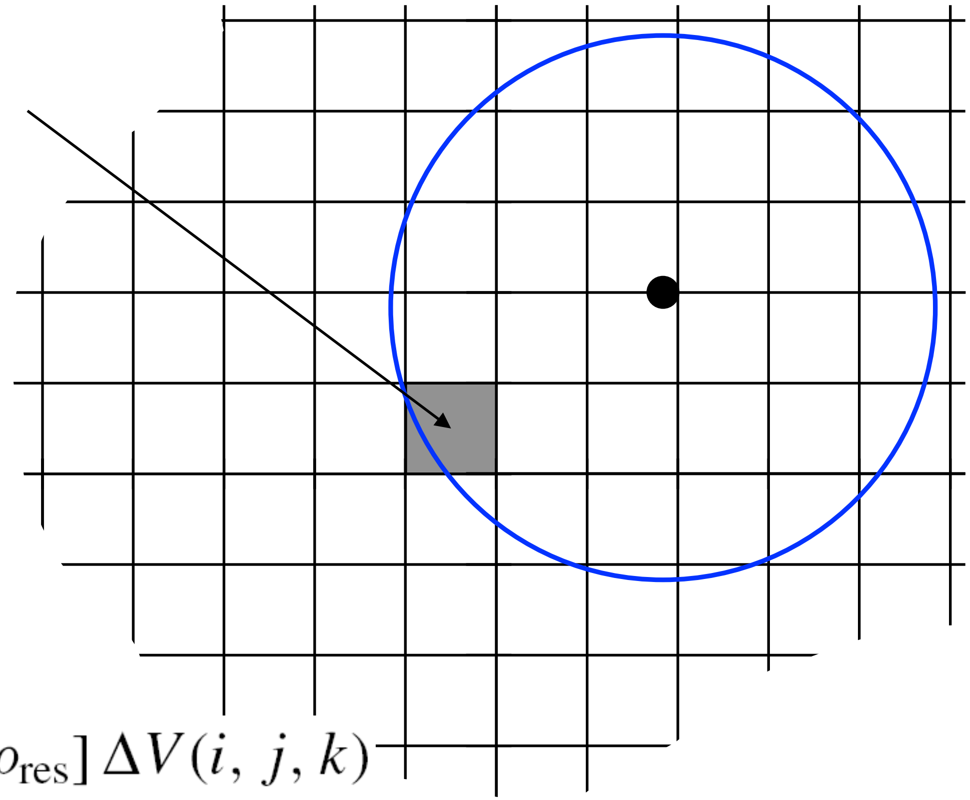
only

Federrath, Banerjee, Clark, Klessen (2010, ApJ 713, 269)

Sink particle implementation in FLASH

Gas accretion

$$\rho > \rho_{\mathrm{res}}$$



1. Gas must be bound to the sink

2. Gas must be moving towards the sink

$$\Delta M = [\rho(i,j,k) - \rho_{\mathrm{res}}]\,\Delta V(i,j,k)$$

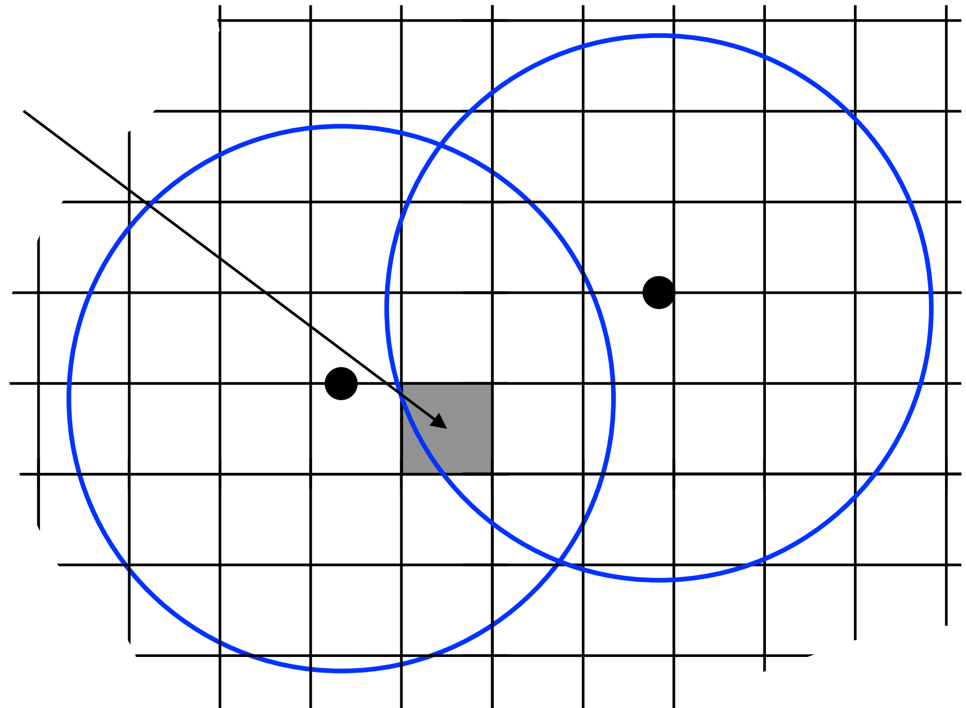
Mass, momentum, angular momentum conservation

Federrath, Banerjee, Clark, Klessen (2010, ApJ 713, 269)

Sink particle implementation in FLASH

Gas accretion

$$\rho > \rho_{\mathrm{res}}$$



1. Gas must be bound to the sink
2. Gas must be moving towards the sink

$$\Delta M = [\rho(i,j,k) - \rho_{\mathrm{res}}]\,\Delta V(i,j,k)$$

Mass, momentum, angular momentum conservation

Federrath, Banerjee, Clark, Klessen (2010, ApJ 713, 269)

Sink particle implementation in FLASH

Gravitational interactions

Gas—Gas	(multigrid solver, tree solver)
Gas—Sinks	(interpolation from grid)
Sinks—Gas	(direct summation, all cells)
Sinks—Sinks	(direct N-Body summation)

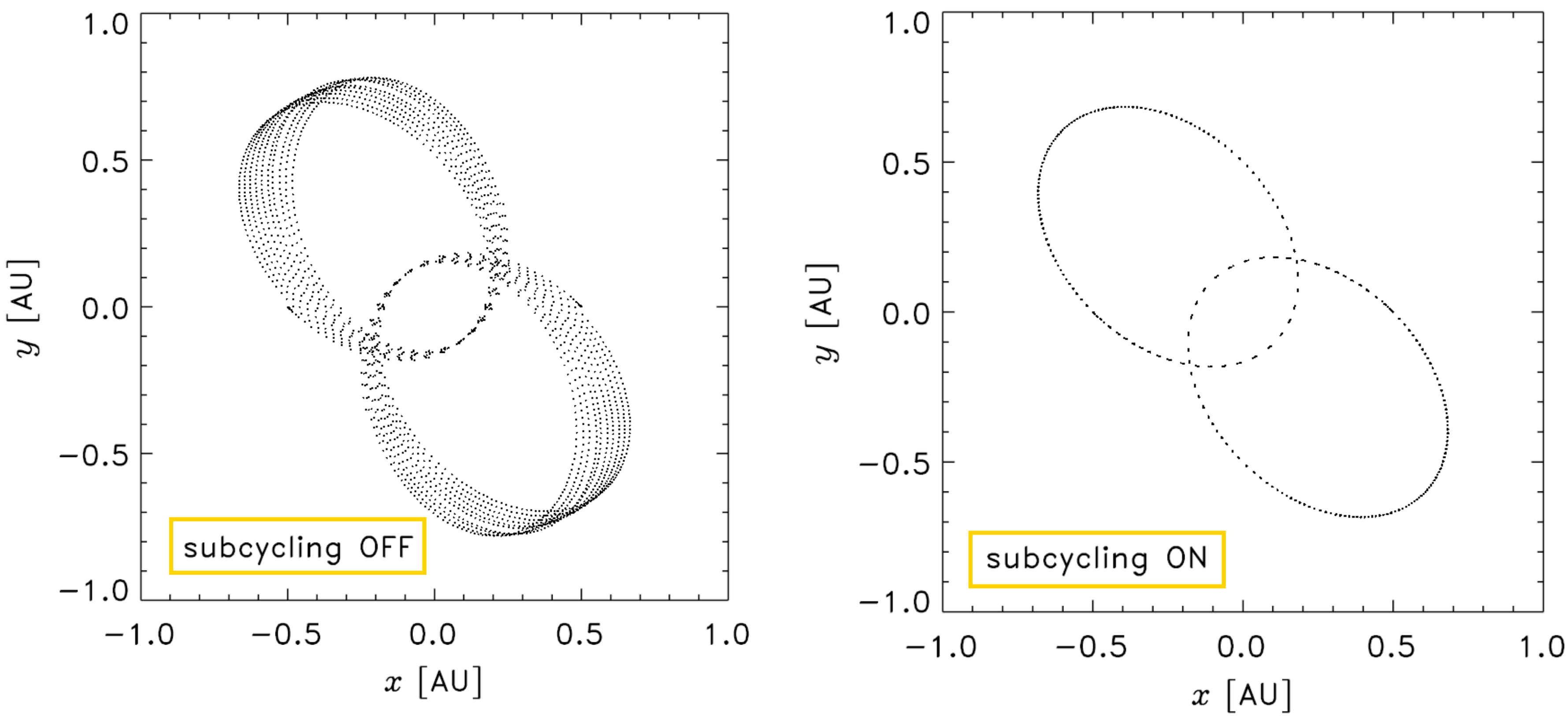
→ Strong constraints on timestep

→ Subcycling with Leapfrog required

Federrath, Banerjee, Clark, Klessen (2010, ApJ 713, 269)

Sink particle implementation in FLASH

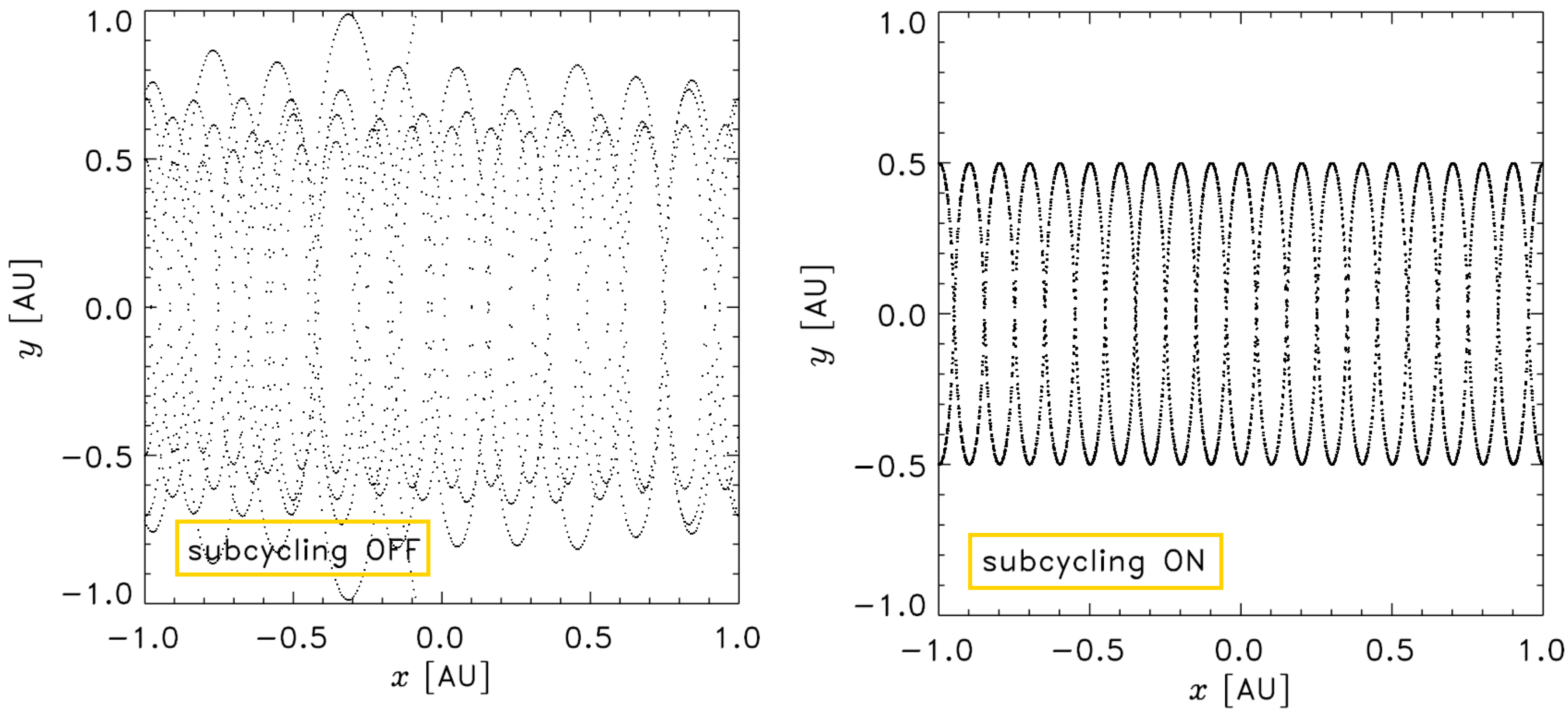
Subcycling required to capture N-Body dynamics



Federrath, Banerjee, Clark, Klessen (2010, ApJ 713, 269)

Sink particle implementation in FLASH

Subcycling required to capture N-Body dynamics



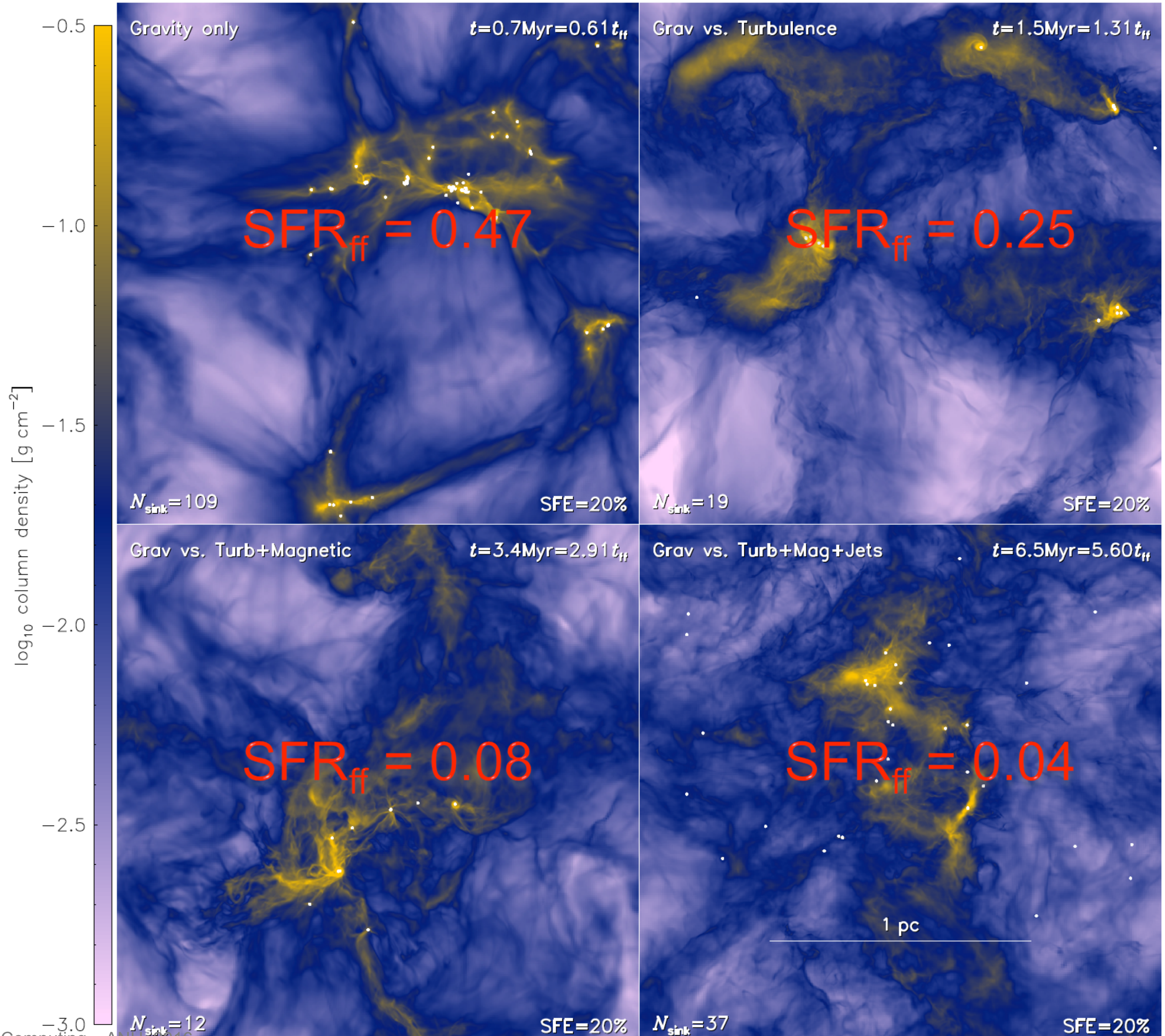
Federrath, Banerjee, Clark, Klessen (2010, ApJ 713, 269)

Grid-based Magnetohydrodynamics with Sink Particles

Movies available: http://www.mso.anu.edu.au/~chfeder/pubs/ineff_sf/ineff_sf.html

Gravity only

Turb



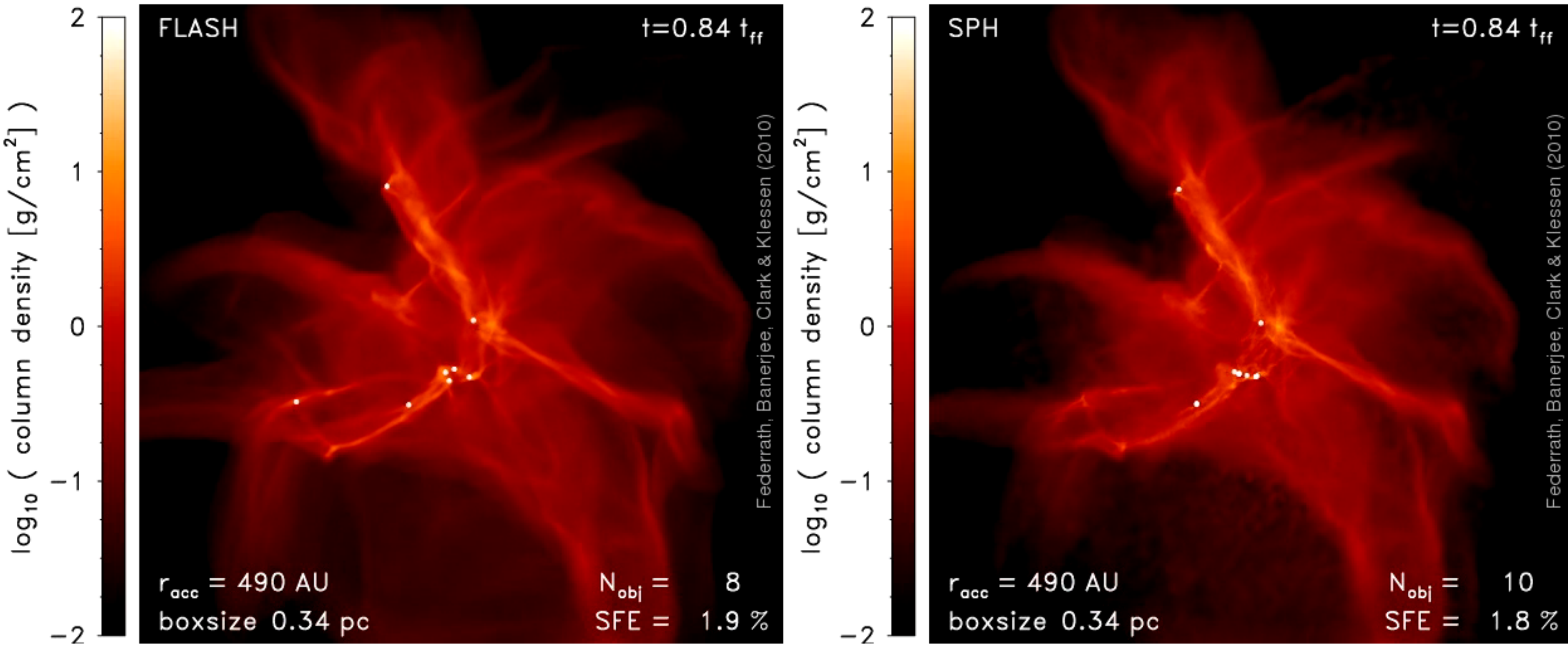
Turb+ Mag

Turb+ Mag+ Jets

Sink particles: AMR versus SPH

Sink particles: AMR versus SPH

Movies available: http://www.ita.uni-heidelberg.de/~chfeder/pubs/sinks/sinks.shtml



FLASH (AMR)

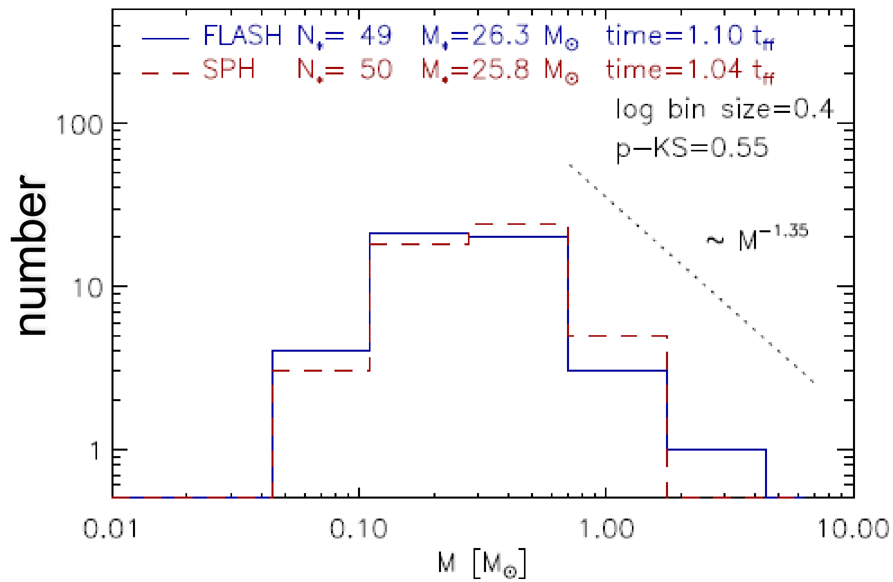
SPH

Federrath, Banerjee, Clark, Klessen (2010, ApJ 713, 269)

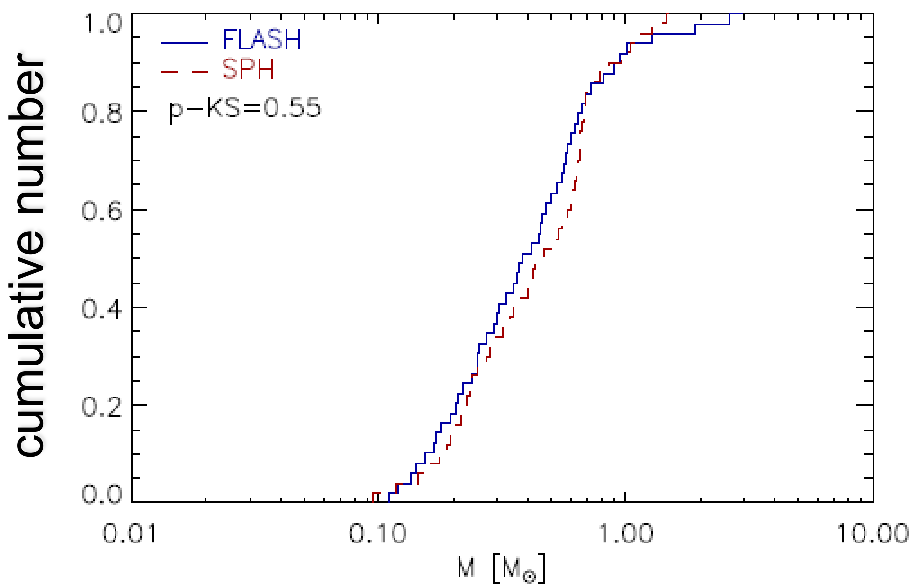
Sink particles: AMR versus SPH

Comparison for SFE~26%

1. Sink mass functions agree well
2. Number of sinks: FLASH 49, SPH 50



sink mass



sink mass

Federrath, Banerjee, Clark, Klessen (2010, ApJ 713, 269)

Sink particle conclusions

(Federrath, Banerjee, Clark, Klessen 2010, ApJ 713, 269)

- Sink creation checks important to avoid spurious sinks in both SPH and AMR

- Encouraging agreement between FLASH and SPH-NG

- computational cost:

  FLASH: 10,300 CPU hours, run on 128 CPUs

  SPH-NG: 2,400 CPU hours, run on 16 CPUs

  (AMR: factor of 30 more resolution elements necessary in FLASH)

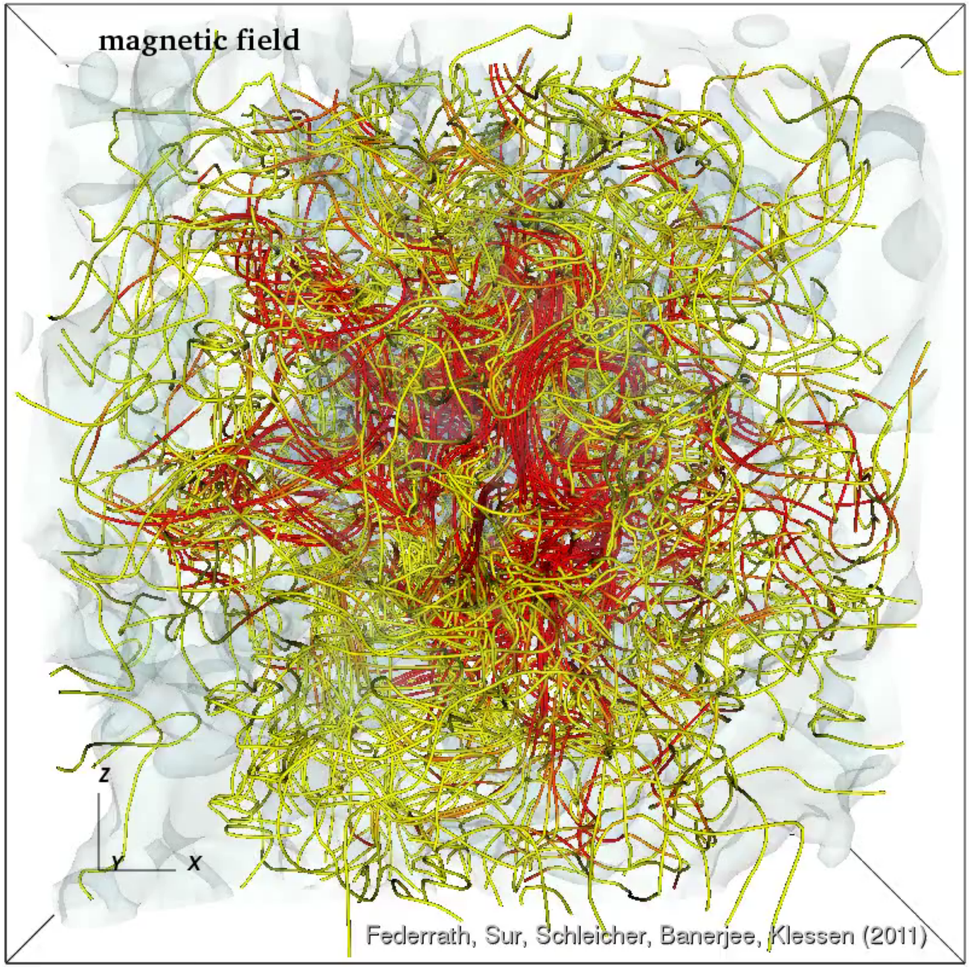
$\longrightarrow$ SPH is faster in collapse calculations

...but what about magnetic fields?

Magnetic fields in SPH and grid

- Problems with magnetic fields in SPH (Price & Federrath 2010)

- Divergence cleaning in SPMHD (ANITA 2012: Tricco & Price 2012, submitted)



Dynamo action
(Federrath, Sur, Schleicher, Banerjee, Klessen 2011, ApJ 731, 62)

Strength and Weaknesses of SPH and grid

SPH	Grid (AMR)
+ Automatic refinement on density	+ Simpler data structure (indexing)
+ Typically faster in collapse calculations	+ Typically faster for pure hydro
+ More robust	+ Refinement on arbitrary quantities (e.g., position, shocks, etc.)
+ Intrinsic mass conservation	+ Magnetic fields
- More complex data structure	- Needs more resolution elements for collapse calculations (AMR)
- Problems with magnetic fields	- Sometimes less robust (solver crashes)

Unstructured Grid (e.g. AREPO)

Springel 2010