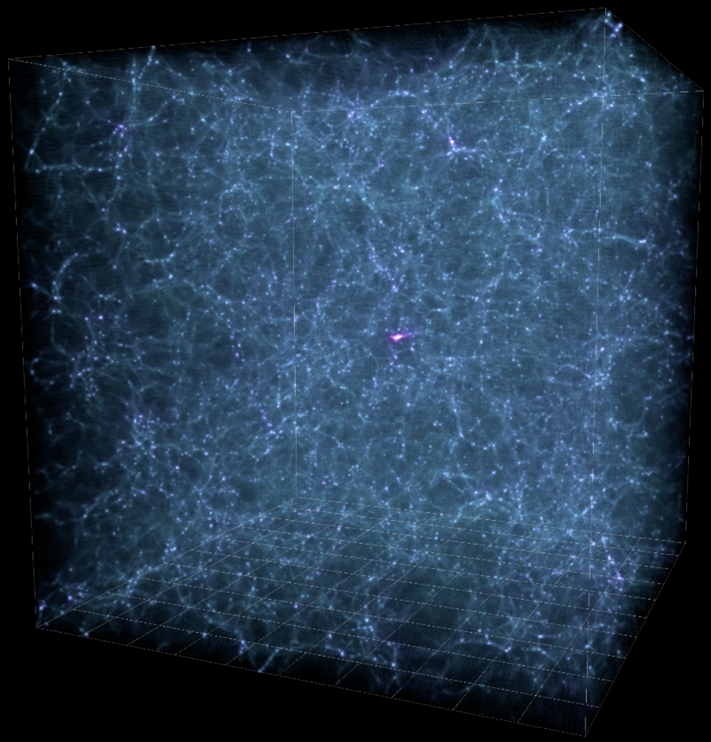
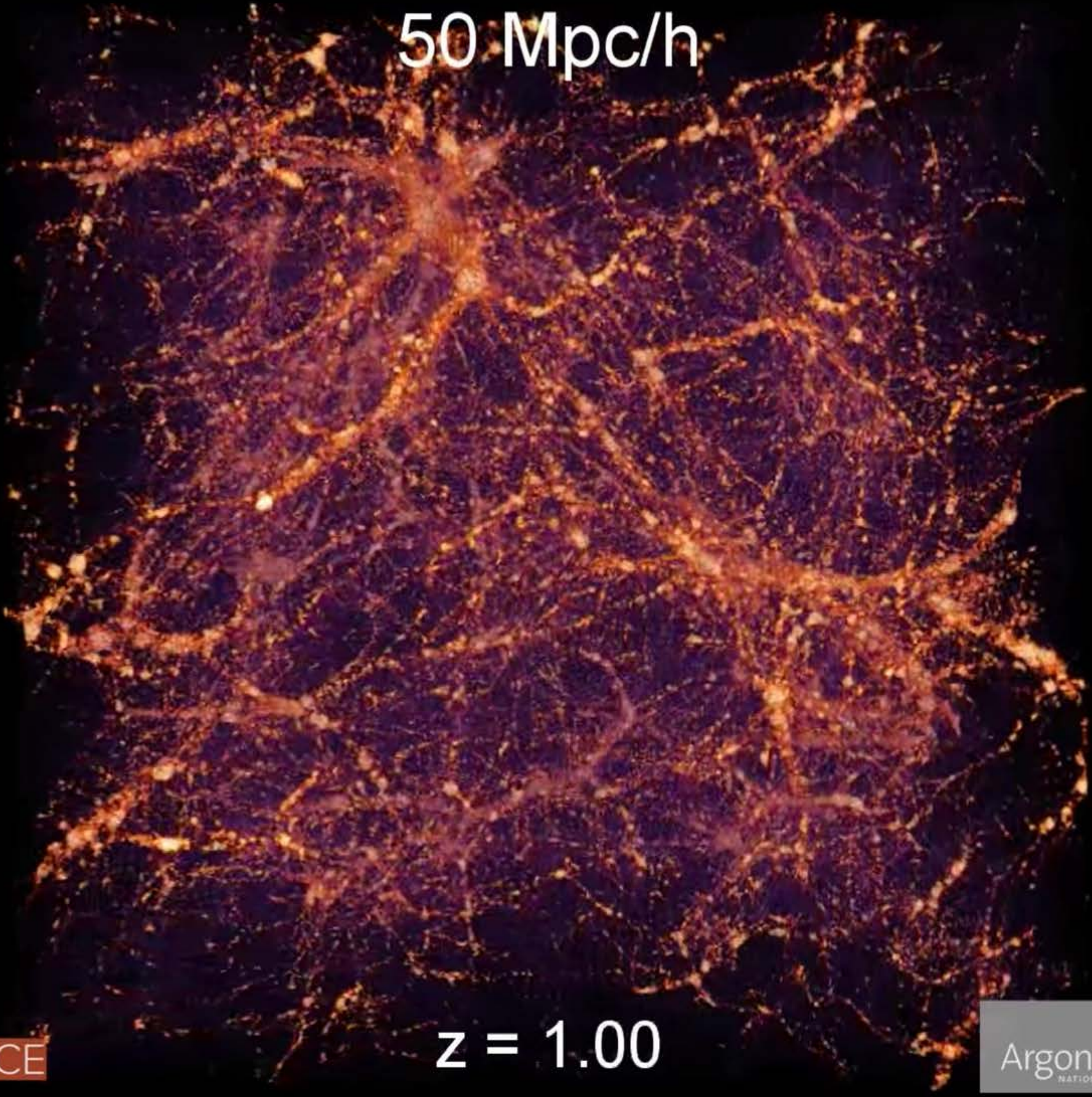
A walk through cosmological simulations and their evolution

LSST-FRANCE Meeting – May 18th, 2022 – LAPP, Annecy

Vincent Reverdy



LAPP l'Observatoire de Paris LUTH | PSL ILLINOIS NCSA



ALCF: EARLY SCIENCE

Argonne NATIONAL LABORATORY
Los Alamos NATIONAL LABORATORY
BERKELEY LAB

A practical summary

What this talk is about

What you need to know about cosmological simulations **in practice**.

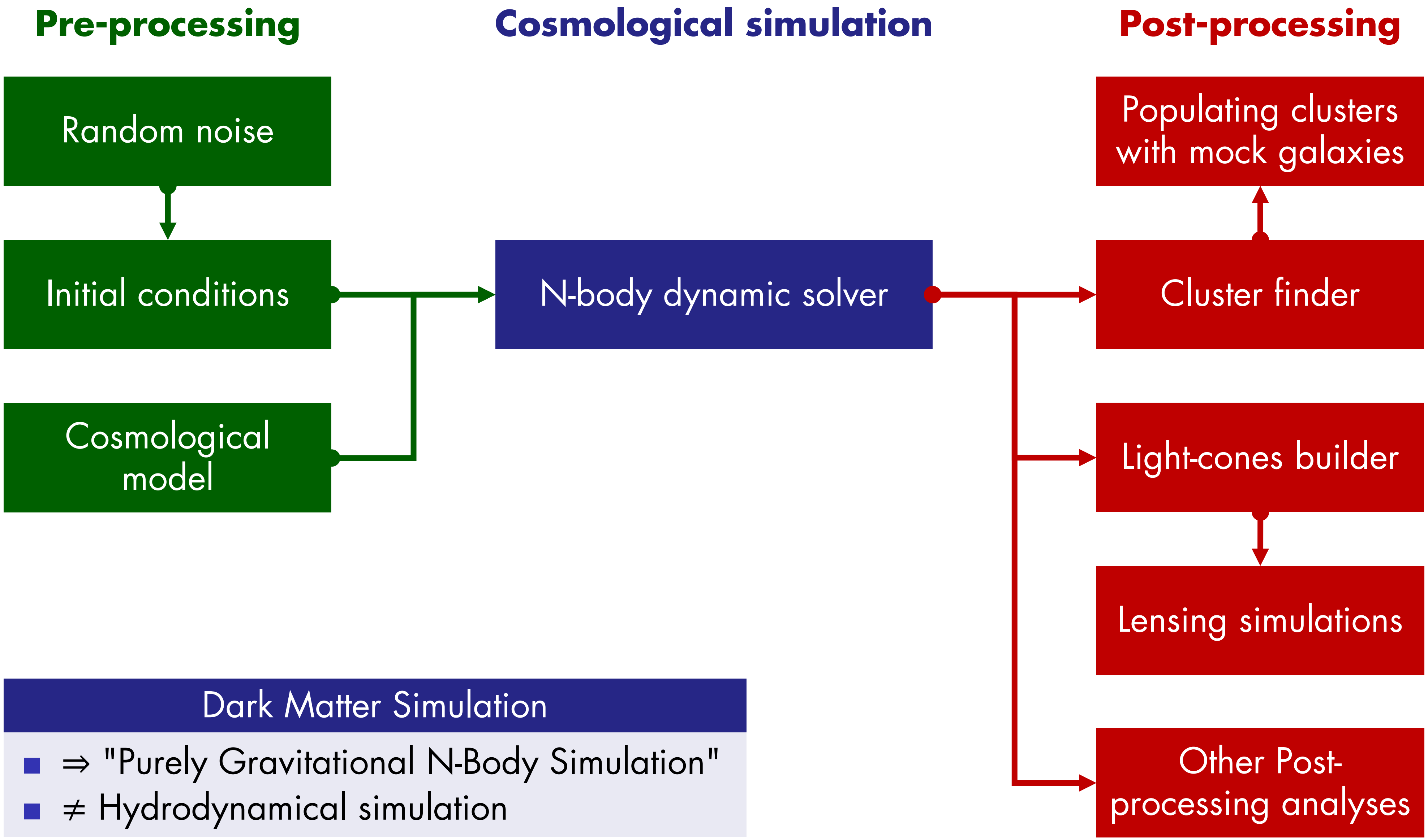
What this talk is NOT about

A theoretical, rigorous, exhaustive presentation on cosmological simulations.

Take-home messages

- Cosmological simulations are **not perfect**
- Full of subtelties, technical details, **numerical approximations** that can impact results
- The more you know about a simulation, the better the **interpretation** will be
- In some cases, machine learning algorithms can **reverse-engineer semi-analytical models**
- In some cases, machine learning algorithms can **learn numerical effects**

Usual cosmological simulation workflow



Dark Matter Simulation

- ⇒ "Purely Gravitational N-Body Simulation"
- ≠ Hydrodynamical simulation

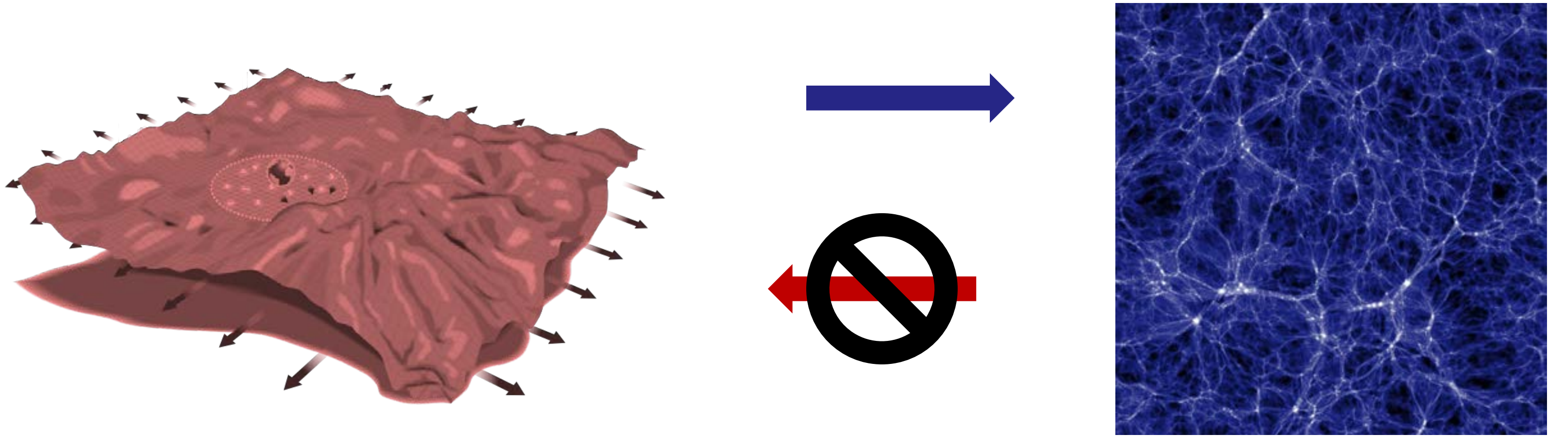
The biggest lie of cosmological simulations

What is NOT done in cosmological simulations

- Cosmological simulations are NOT solving general relativity

What is done in cosmological simulation

- Solve newtonian gravity in a homogeneous expanding background
- Expansion is pre-computed (FLRW solver)
- Instantaneous propagation of gravity
- $\Rightarrow$ *see debates on the Backreaction Conjecture*

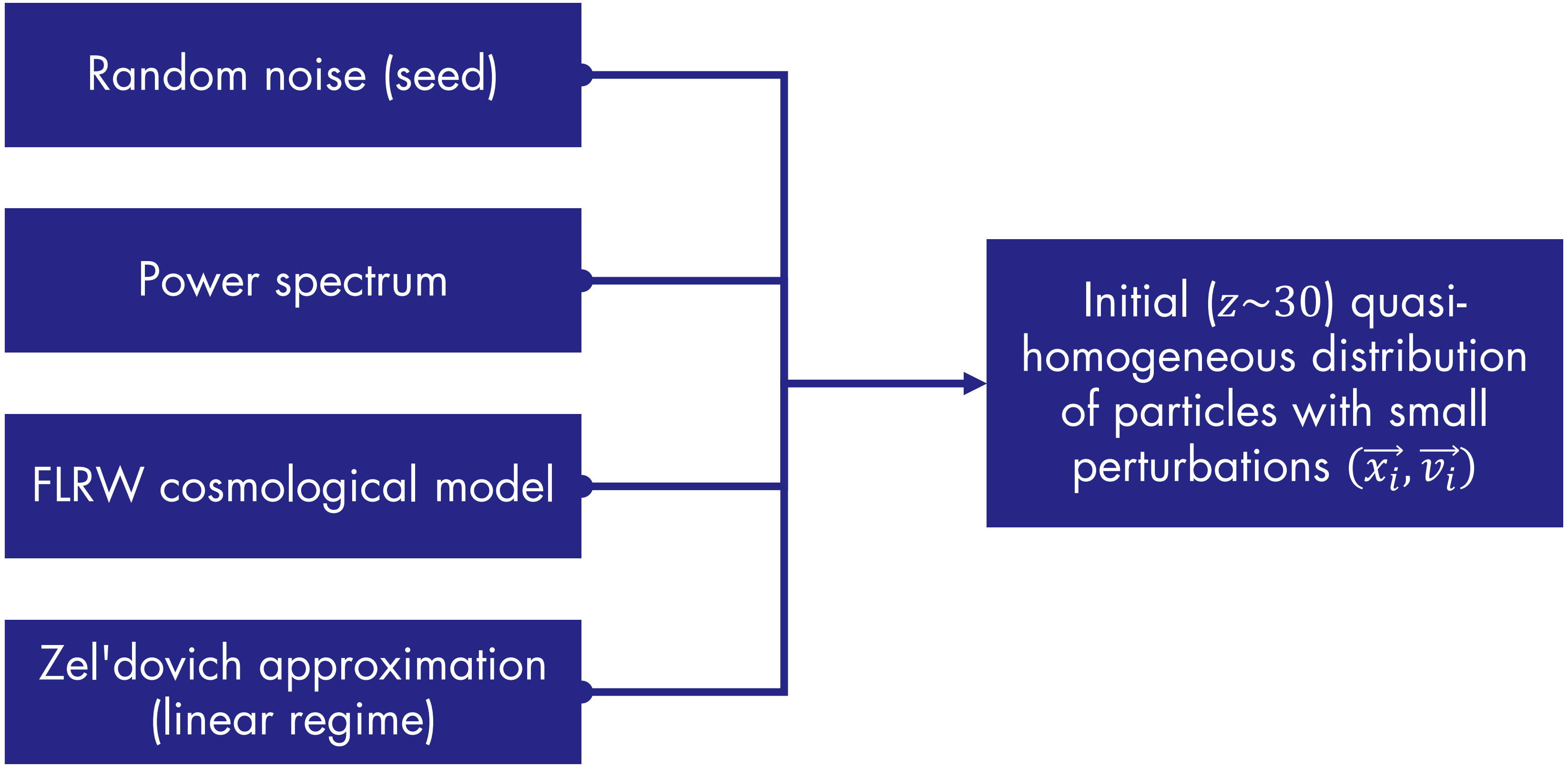


Precomputed FLRW metric

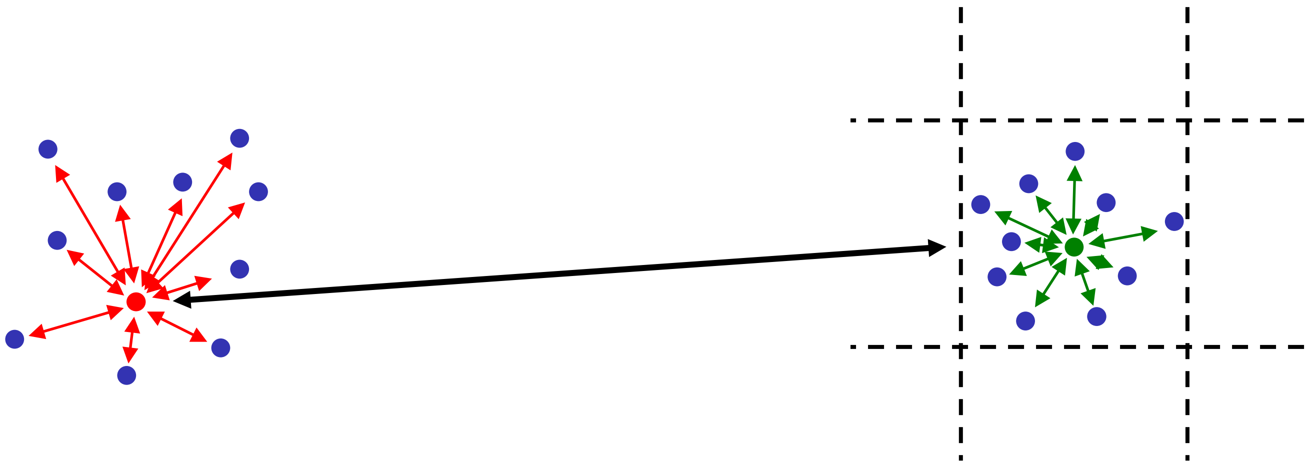
$$G_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu}$$

Matter density dynamics

Initial conditions



N-body gravitational solvers: particles and meshes



PP: Particle-Particle

- $\mathcal{O}(N^2)$
- Short distance: Particle-Particle
- Long distance: Particle-Particle

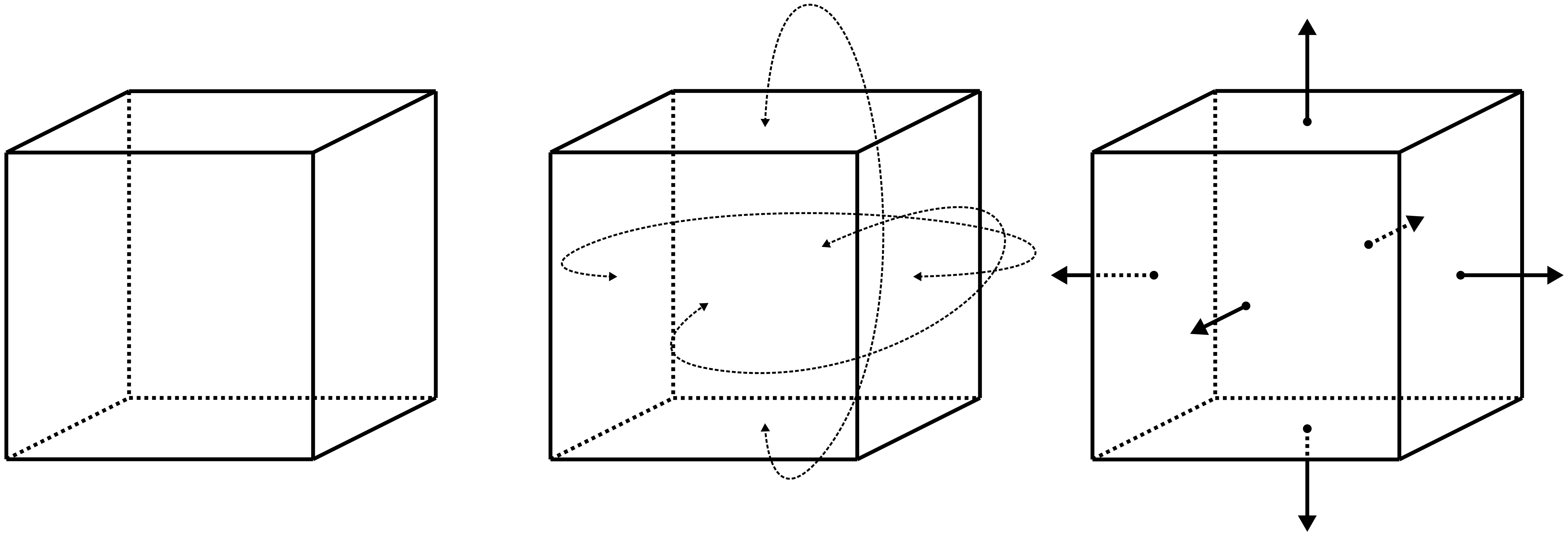
PM: Particle-Mesh

- $\mathcal{O}(N \log N)$
- Short distance: Particle-Mesh
- Long distance: Particle-Mesh

P^3M: Particle-Particle Particle-Mesh

- $\mathcal{O}(N \log N)/\mathcal{O}(N)$
- Short distance: Particle-Particle
- Long distance: Particle-Mesh

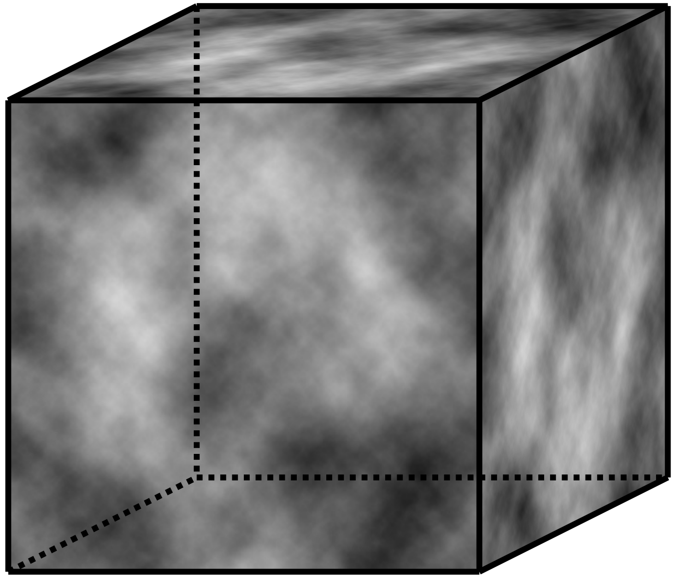
Simulation box: periodic and expanding



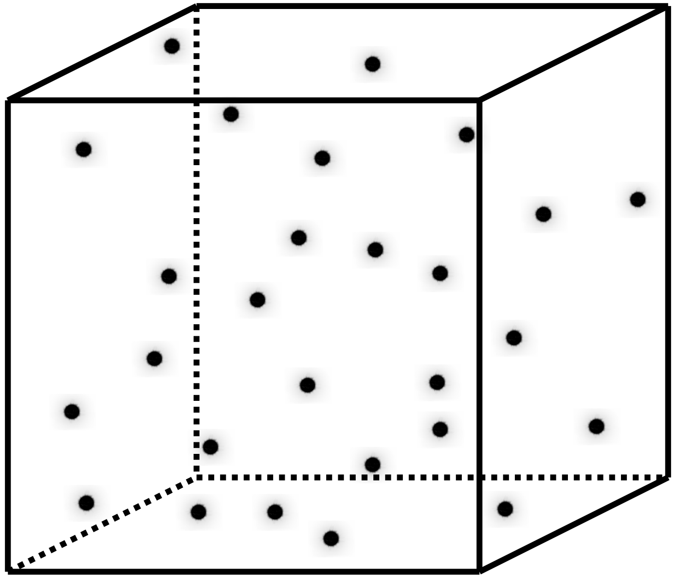
Simulation box

Periodic boundary conditions

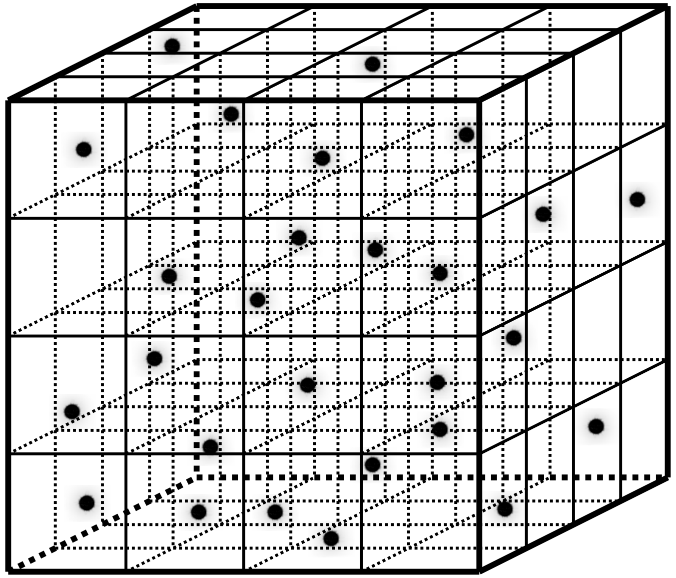
Coordinate system to take expansion into account



Initial density distribution

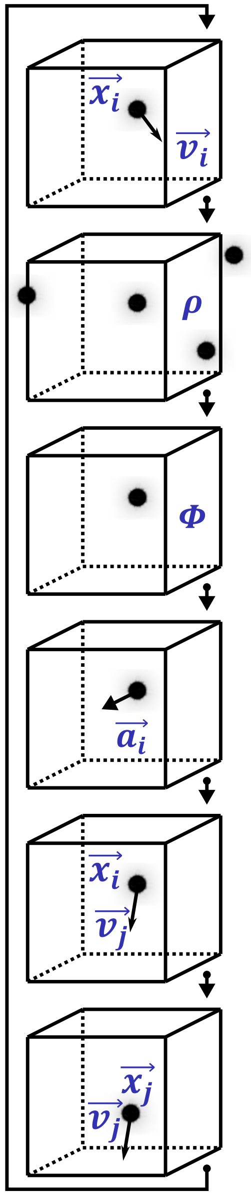


Matter particles



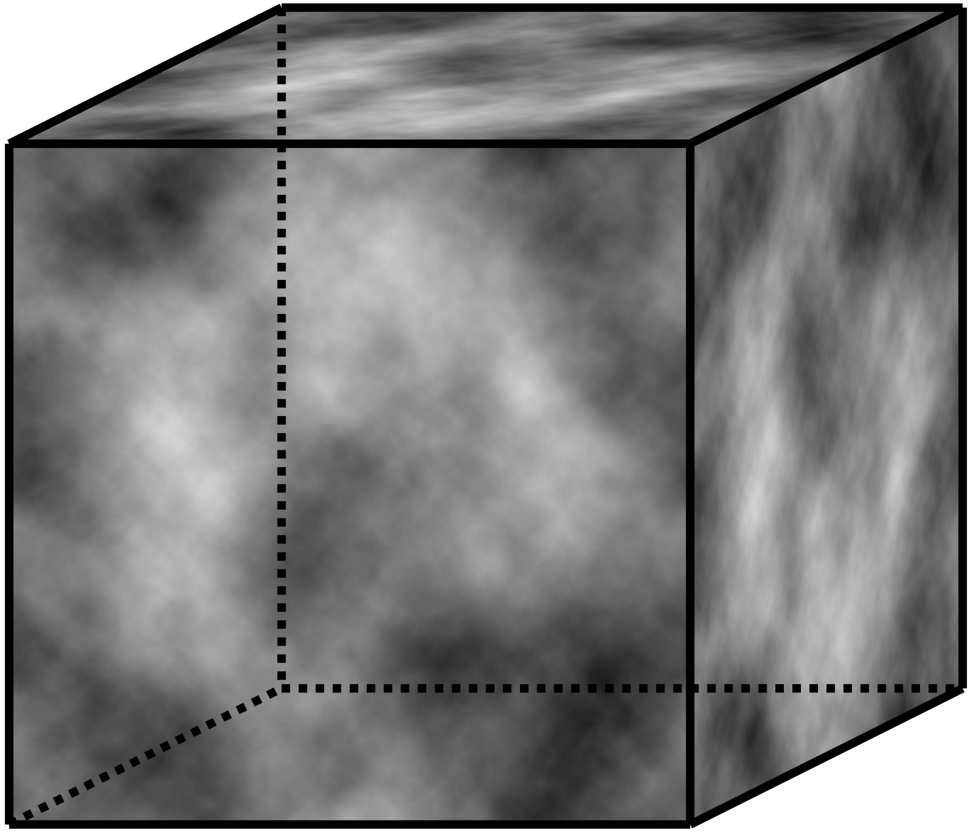
Cubic mesh: Regular or Adaptive Mesh Refinement

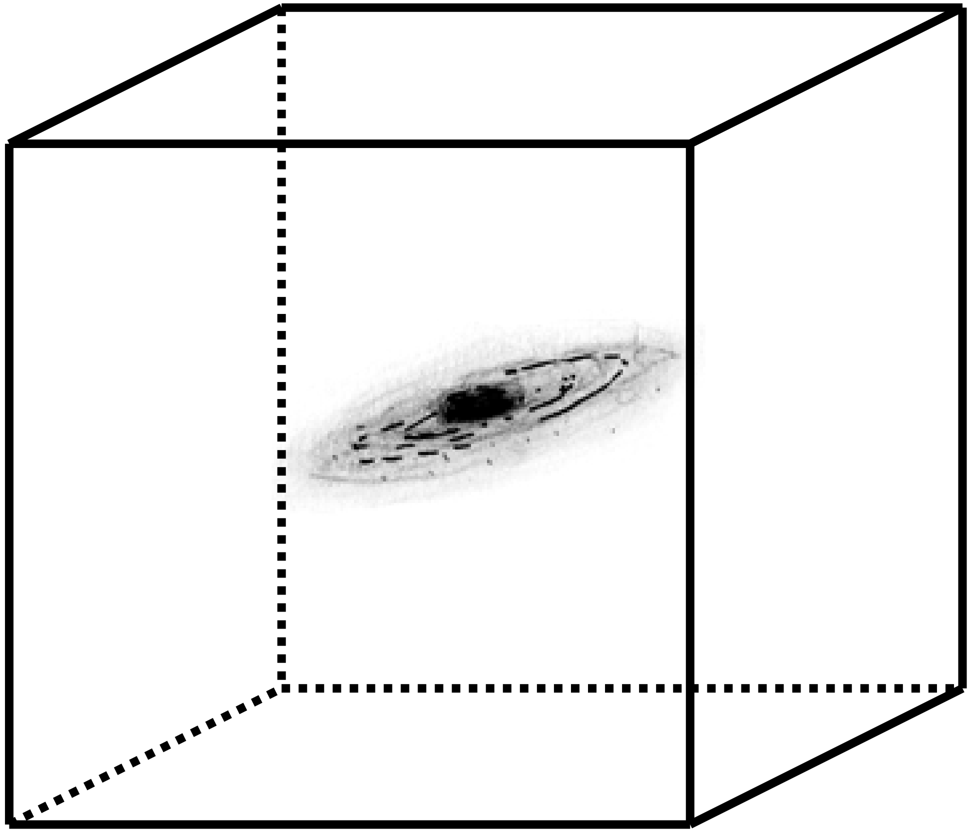
Core of the iterative process for each particle



- 1) For each cell c containing particles with position $\vec{x_i}$ and velocity $\vec{v_i}$
- 2) Interpolate density ρ in cell c depending on surrounding particles
- 3) From ρ compute the gravitational potential Φ
- 4) From Φ interpolate back the acceleration $\vec{a}$ at position $\vec{x_i}$
- 5) From $\vec{a}$ compute the new speed $\vec{v_j}$ of each particle
- 6) From $\vec{v_j}$ compute the new position $\vec{x_j}$ of each particle
- 7) Restart at 1) with uploaded position $\vec{x}$ and speed $\vec{v}$

And that's how large scale cosmic structure formation is simulated



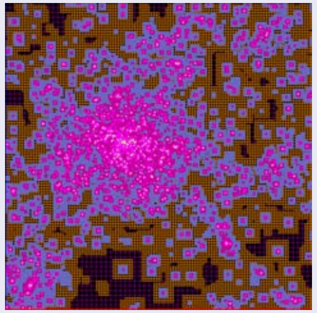


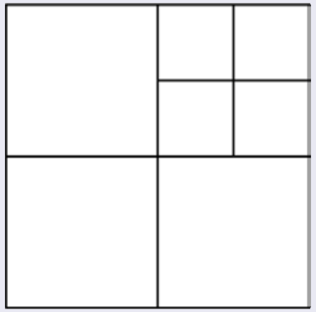
Initial conditions of the simulation (~homogeneous)

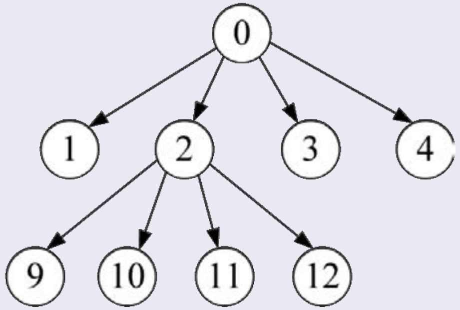
Gravitational collapse and structure formation

PM-AMR: Adaptive Mesh Refinement

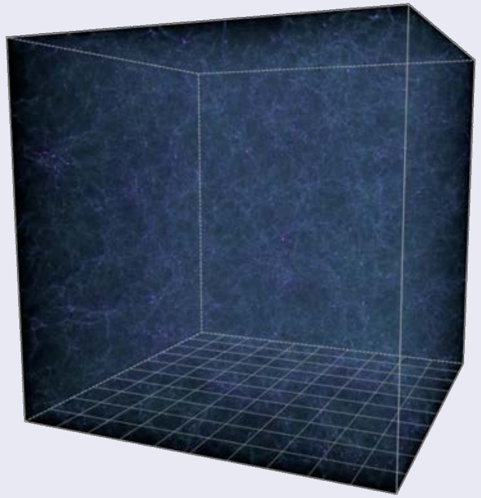
Quadtrees in 2D

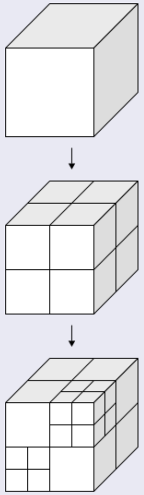


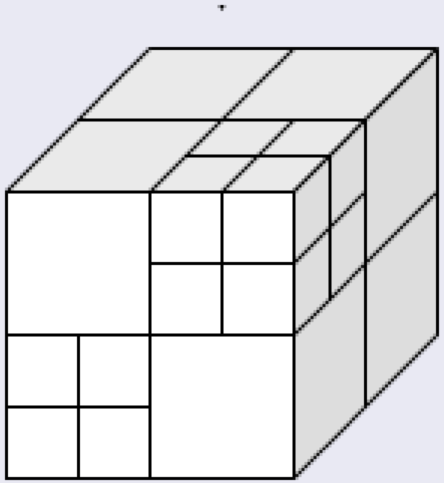


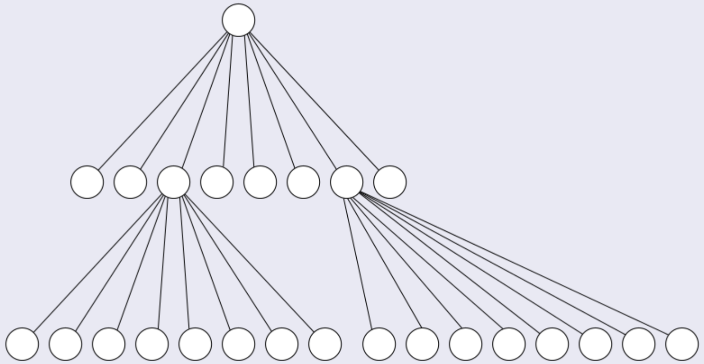


Octrees in 3D



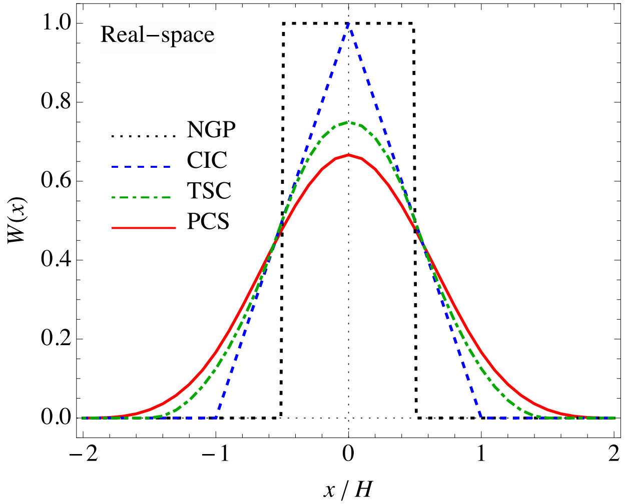






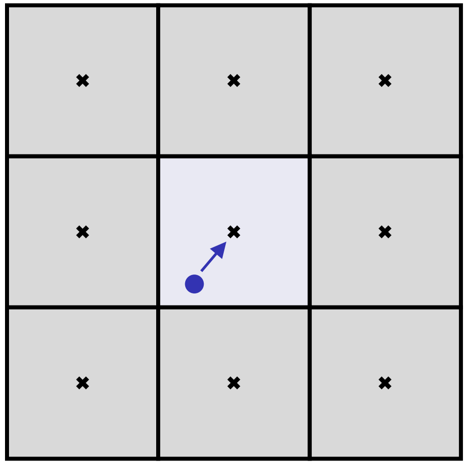
Interpolation schemes: NGP, CIC, TSC…

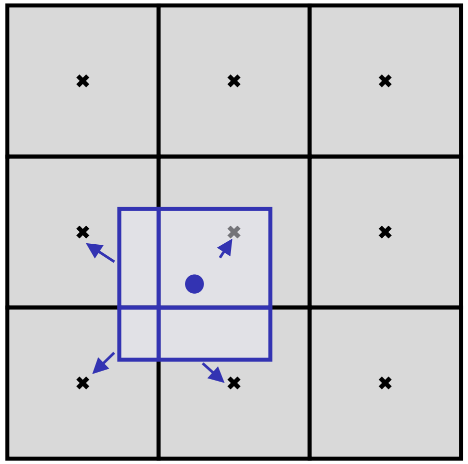
1D



NGP	CIC	TSC
Nearest Grid Point	Cloud-in-Cell	Triangular Shaped Cloud

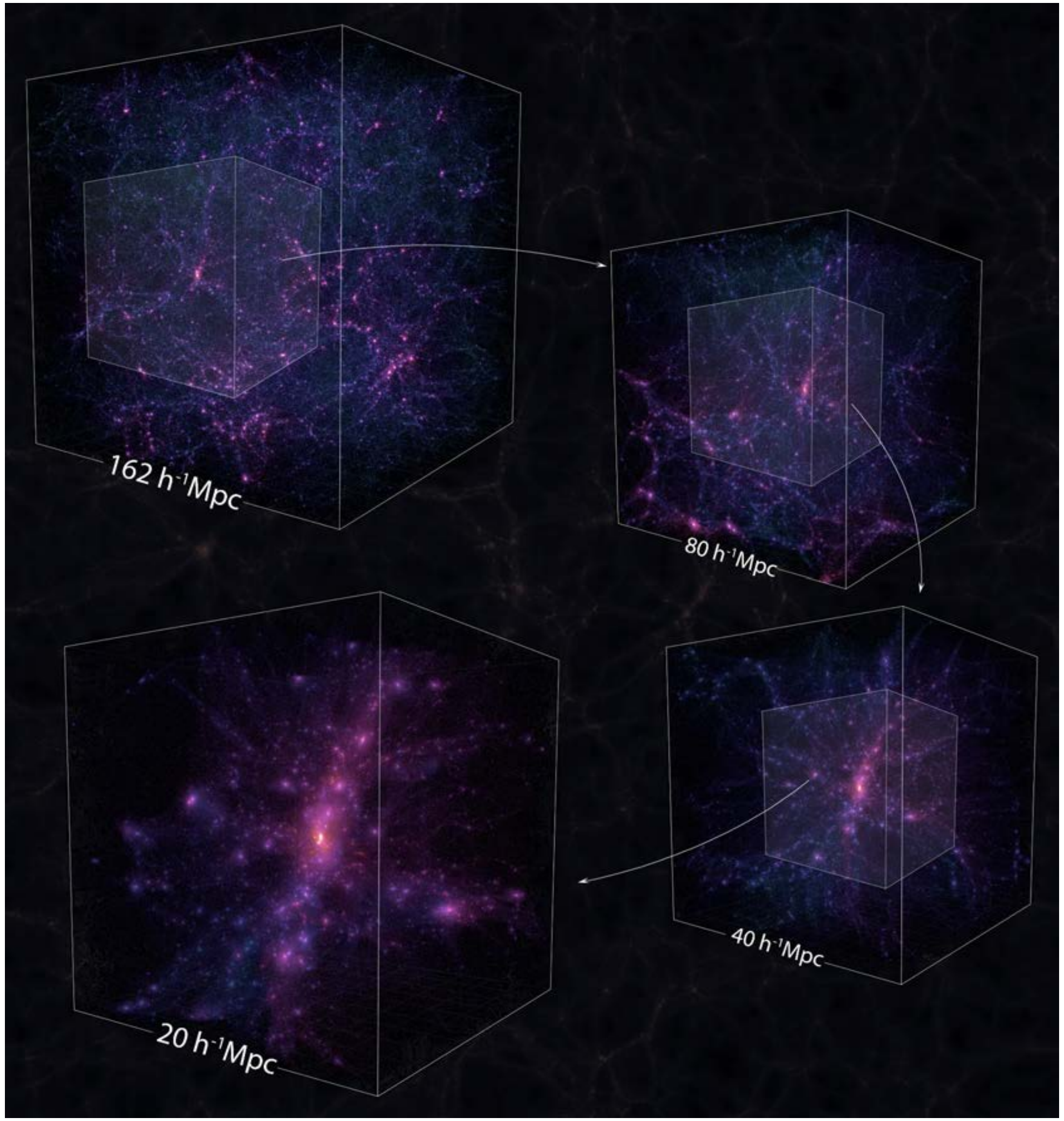
2D





Cells	1D	2D	3D
NGP	1	1	1
CIC	2	4	8
TSC	3	9	27

Properties of pure dark matter simulations



Main parameters

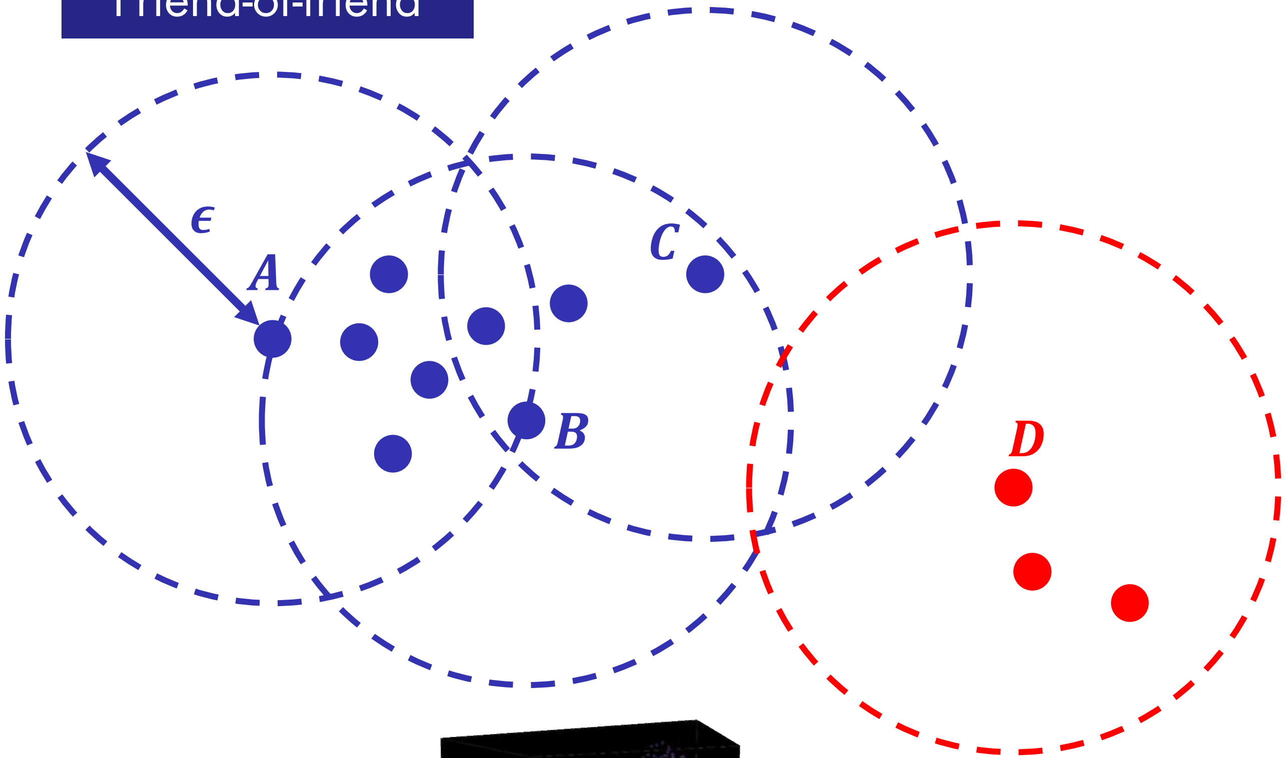
- Initial positions and speed of particles
- Cosmological model
- Box size
- Number of particles
- Resolution in mass (particle mass)
- Resolution in size (minimum cell size)
- Resolution in time (time step)

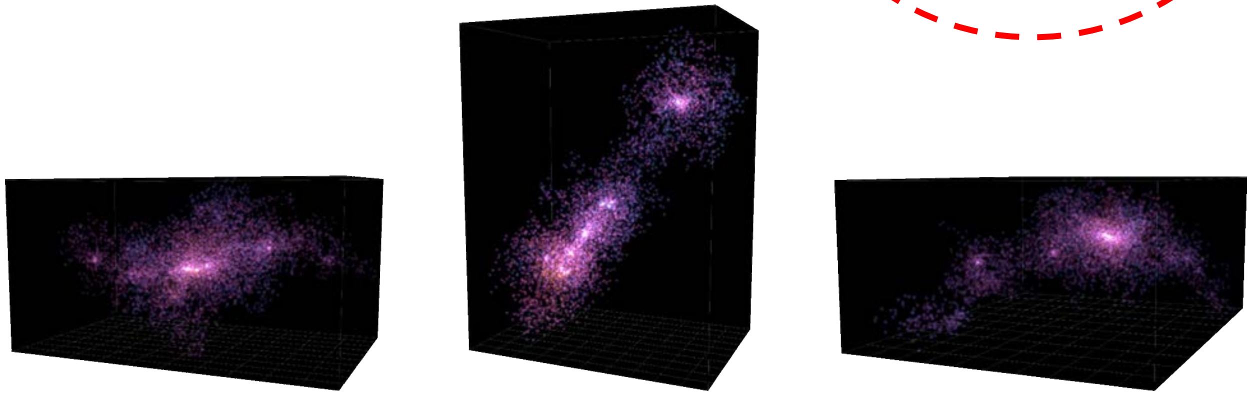
Solver parameters (examples)

- Algorithm
- Discretization strategy
- Refinement strategy
- Floating-point precision
- Interpolation scheme
- Parallelization strategy

Halo finders and populating haloes with galaxies

Friend-of-friend





Populating haloes with galaxies

- Semi-analytical models
- E.g.: Halo Occupation Distribution (HOD)

The following operations may NOT commute

- Cluster detection
- Populating with galaxies

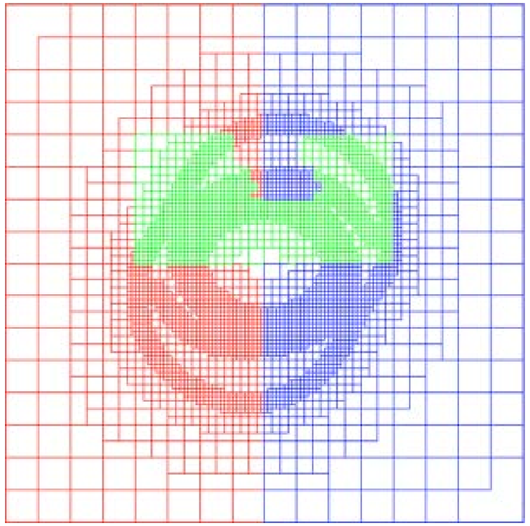
Metaparameters dependency

- May depend on non-physical metaparameters

There is no "TRUE" clusters in simulations

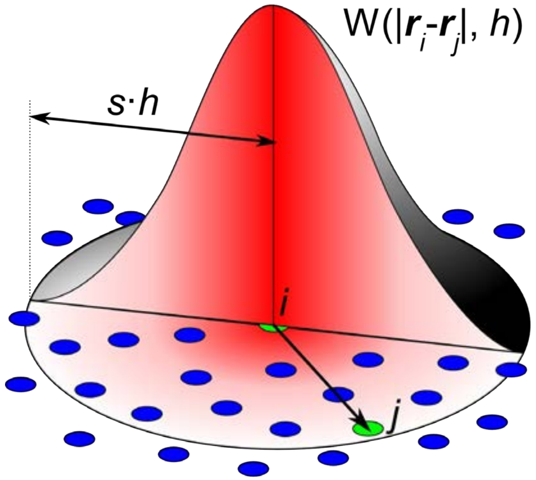
- Depend on cluster detection algorithms
- Depend on models to populate with galaxies

Alternative approach: hydrodynamics simulations



Eulerian approach

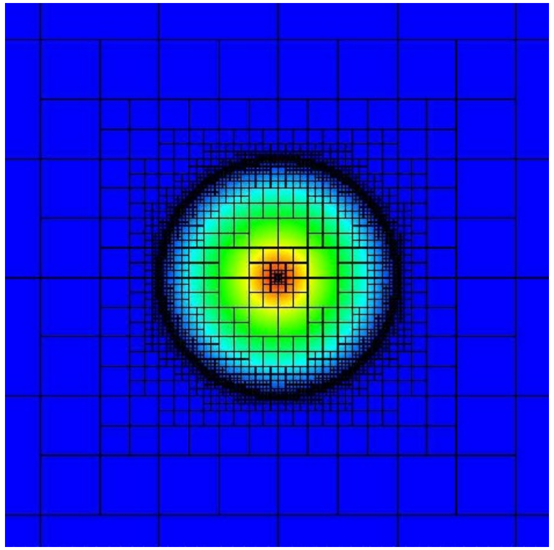
Lagrangian approach

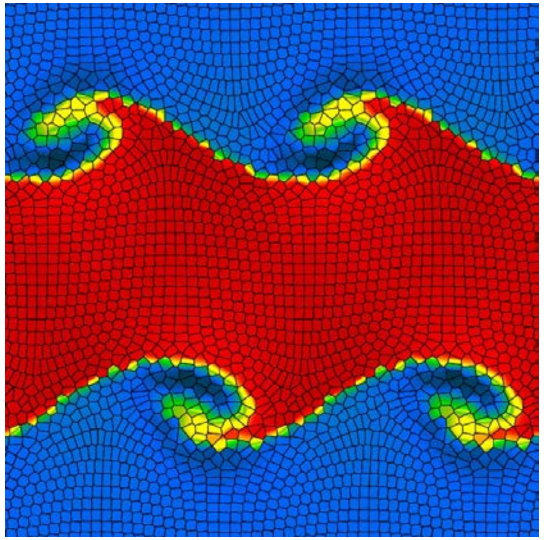


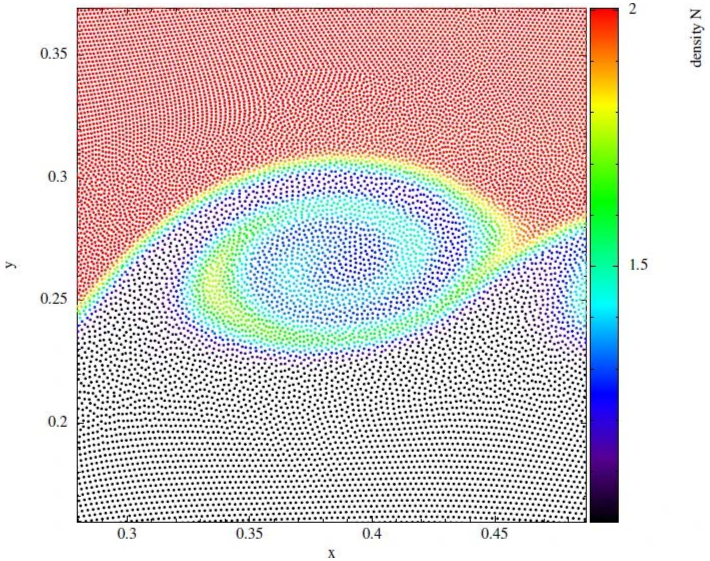
AMR codes

Moving mesh / Tesselation

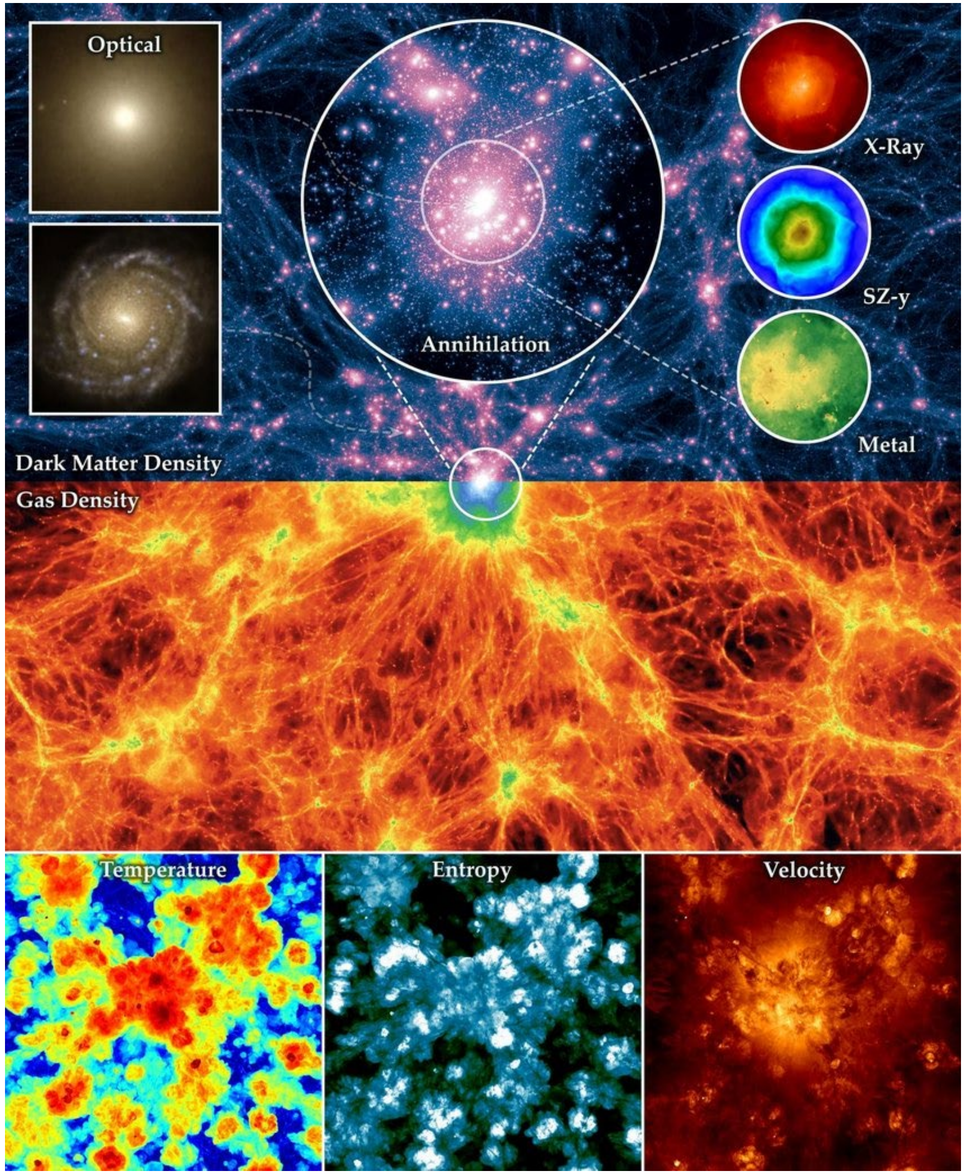
SPH codes







Hydrodynamical cosmological simulations



Principle

- Simulate baryons on top of dark matter

Pros

- No need of semi-analytical populating algorithms
- Remove associated metaparameters

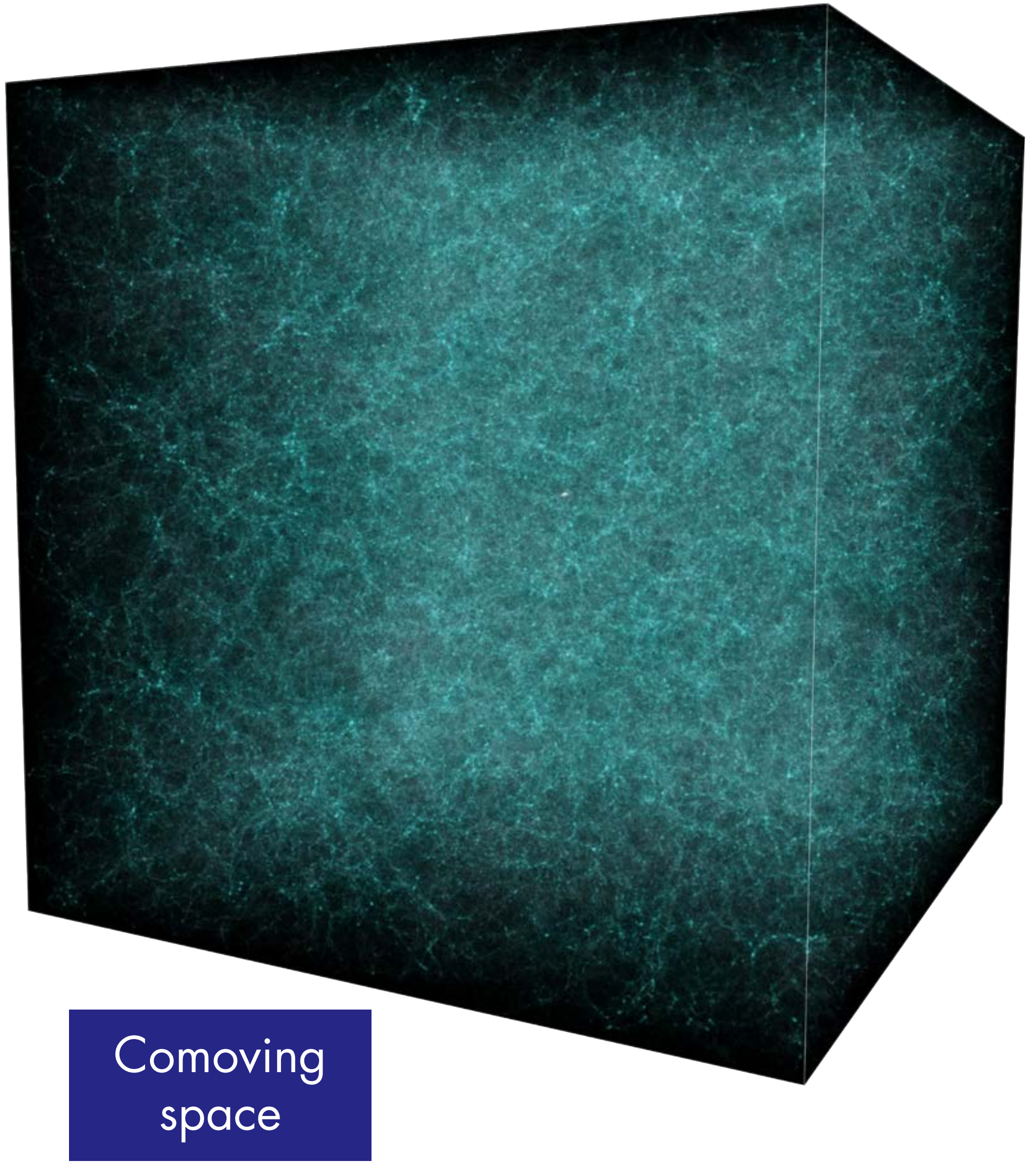
Cons

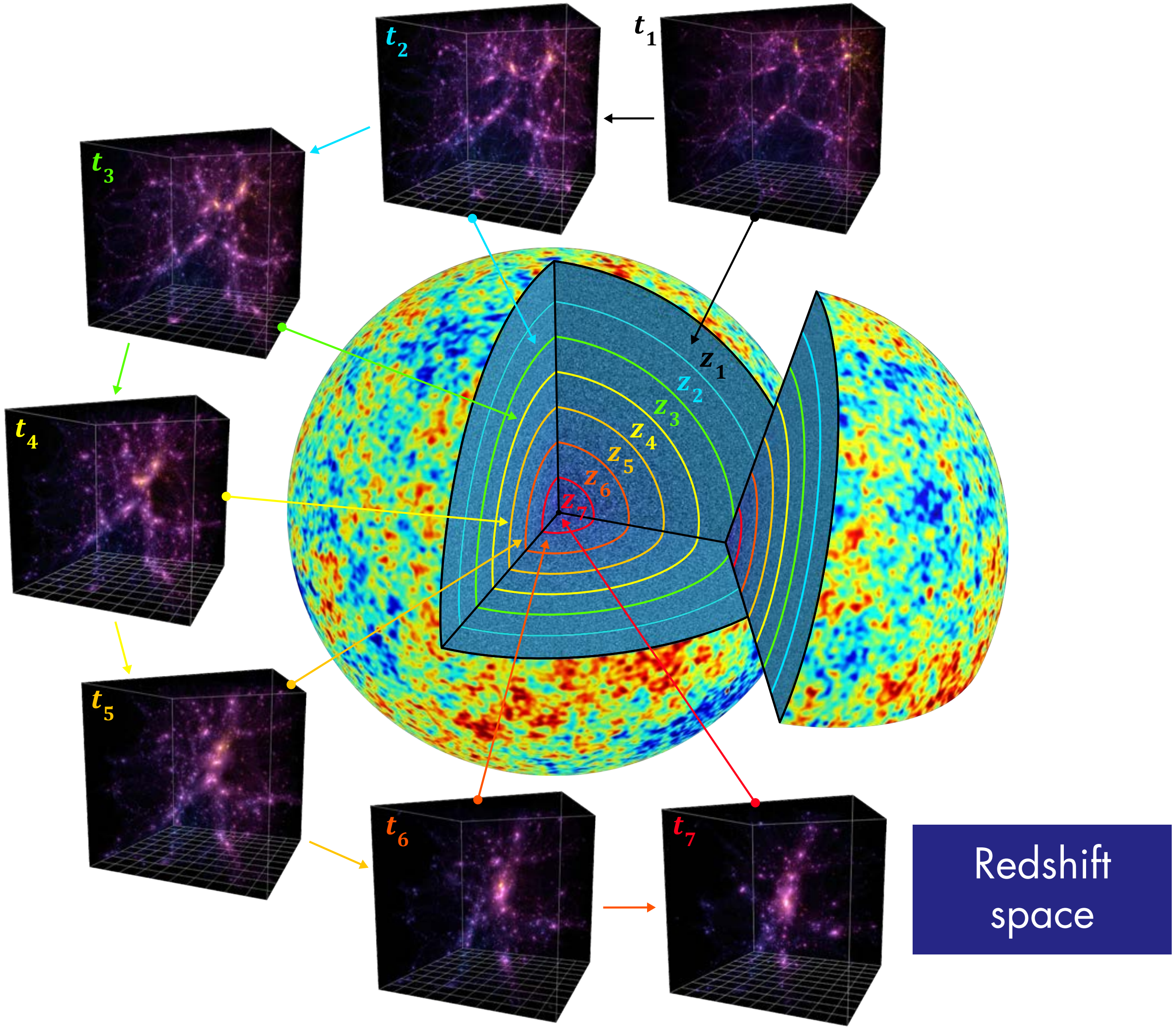
- Gastrophysics (lots of metaparameters)
- Subgrid semi-analytical models

Subgrid models

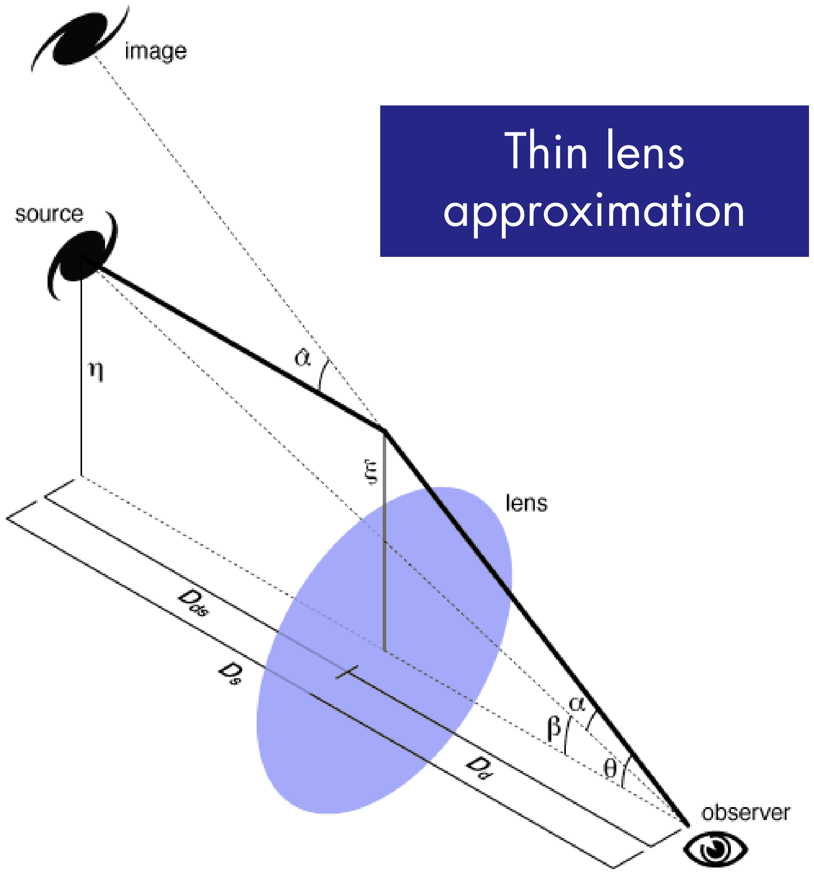
- Subgrid models are (generally) BAD

Simulation coordinates



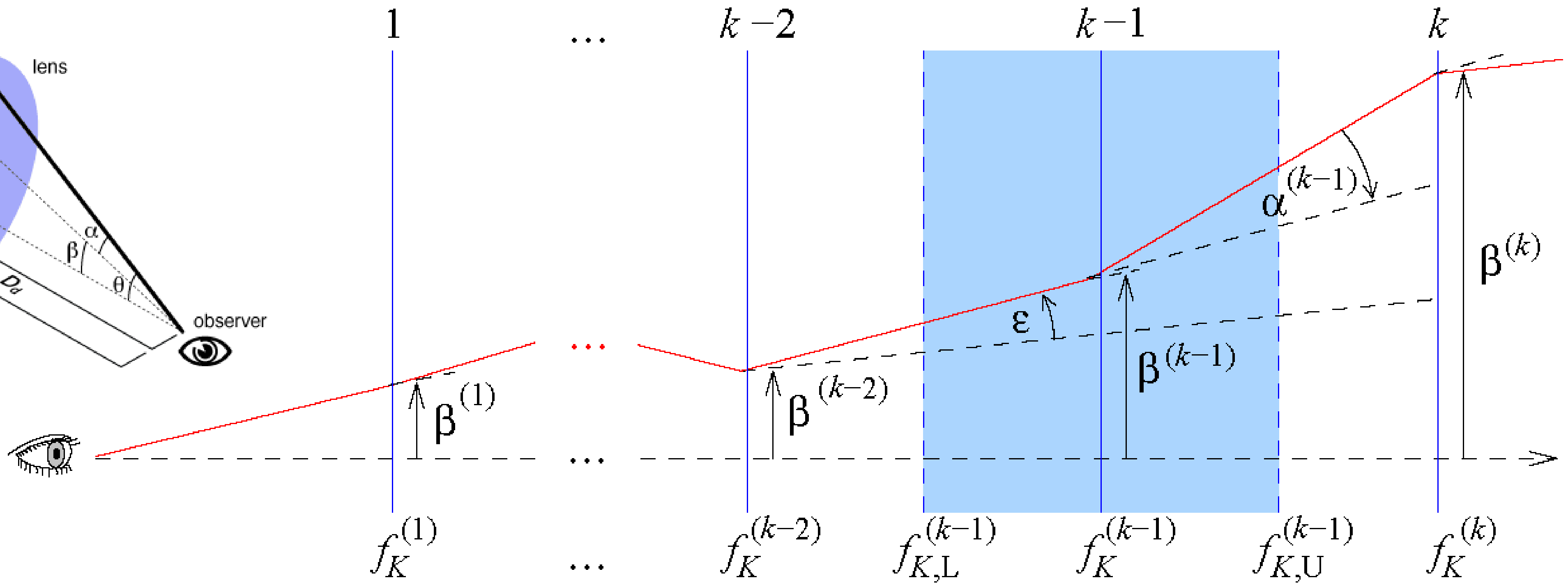


Lensing simulations: usual approximations

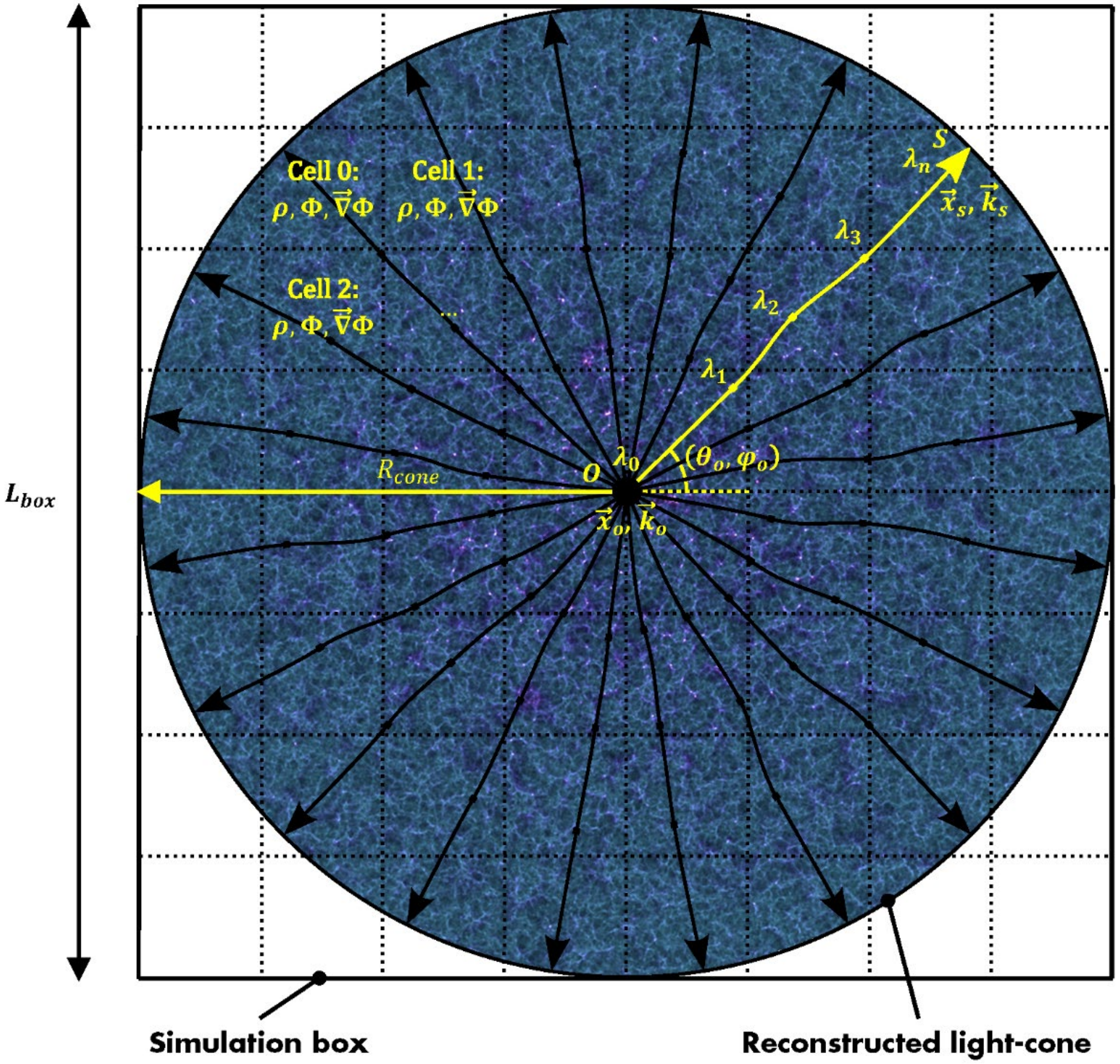


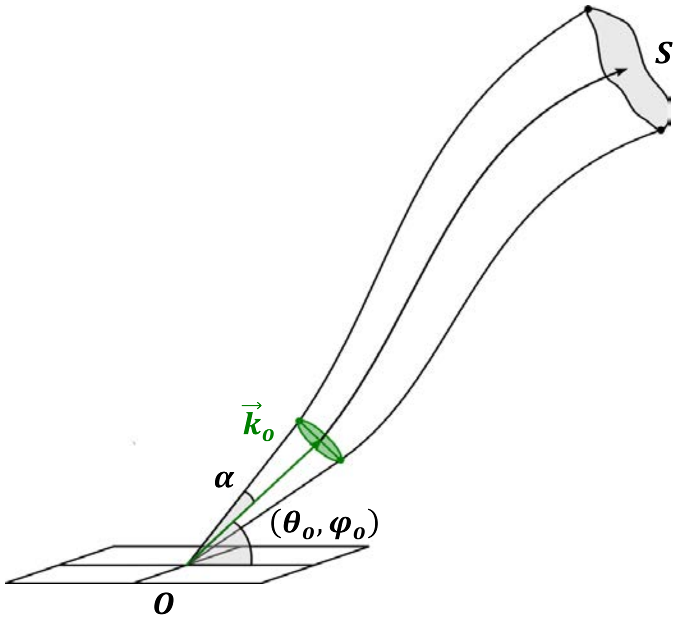
Thin lens approximation

Multi-lens plane approximation



Relativistic raytracing: solving geodesics equations directly





Geodesics equations at 1st order

$$\frac{d^2\eta}{d\lambda^2} \approx -\frac{2a'}{a}\frac{d\eta}{d\lambda}\frac{d\eta}{d\lambda} - \frac{2}{c^2}\frac{d\Phi}{d\lambda}\frac{d\eta}{d\lambda} + 2\frac{\partial\Phi}{\partial\eta}\left(\frac{d\eta}{d\lambda}\right)^2$$

$$\frac{d^2x}{d\lambda^2} \approx -\frac{2a'}{a}\frac{d\eta}{d\lambda}\frac{dx}{d\lambda} + \frac{2}{c^2}\frac{d\Phi}{d\lambda}\frac{dx}{d\lambda} - 2\frac{\partial\Phi}{\partial x}\left(\frac{d\eta}{d\lambda}\right)^2$$

$$\frac{d^2y}{d\lambda^2} \approx -\frac{2a'}{a}\frac{d\eta}{d\lambda}\frac{dy}{d\lambda} + \frac{2}{c^2}\frac{d\Phi}{d\lambda}\frac{dy}{d\lambda} - 2\frac{\partial\Phi}{\partial y}\left(\frac{d\eta}{d\lambda}\right)^2$$

$$\frac{d^2z}{d\lambda^2} \approx -\frac{2a'}{a}\frac{d\eta}{d\lambda}\frac{dz}{d\lambda} + \frac{2}{c^2}\frac{d\Phi}{d\lambda}\frac{dz}{d\lambda} - 2\frac{\partial\Phi}{\partial z}\left(\frac{d\eta}{d\lambda}\right)^2$$

⇒ see Magrathea-Pathfinder (Reverdy 2014, Breton & Reverdy 2022)

Being careful with post-processing of numerical simulations

Two types of effects

- Physics effects
- Numerical effects (~ biases and systematics of experiments)

Example of machine-learning epic failures

- Learning from the encoding of floating-point numbers
- Learning the inverse of a semi-analytical models
- Learning preferred directions on cartesian grids

General rules

- The higher the number of metaparameters, the harder it is to distinguish the numerical and physics effects
- The higher the number of post-processing phases, the harder it is to correctly interpret the results
- Always keep in mind that an algorithm may be measuring the result of a numerical effect
- Black box simulations + black box post-processing algorithms ≈ black box results

Conclusions and questions

Simulations are numerical experiments

- As for every experiments there are biases and systematics

General Relativity

- Not fully relativistic ⇒ Newtonian gravity in a precomputed FLRW expansion

Non-linear regime

- Powerful tool to study the non-linear collapse of matter

Take-home messages

- Cosmological simulations are **not perfect**
- Full of subtelties, technical details, **numerical approximations** that can impact results
- The more you know about a simulation, the better the **interpretation** will be
- In some cases, machine learning algorithms can **reverse-engineer semi-analytical models**
- In some cases, machine learning algorithms can **learn numerical effects**