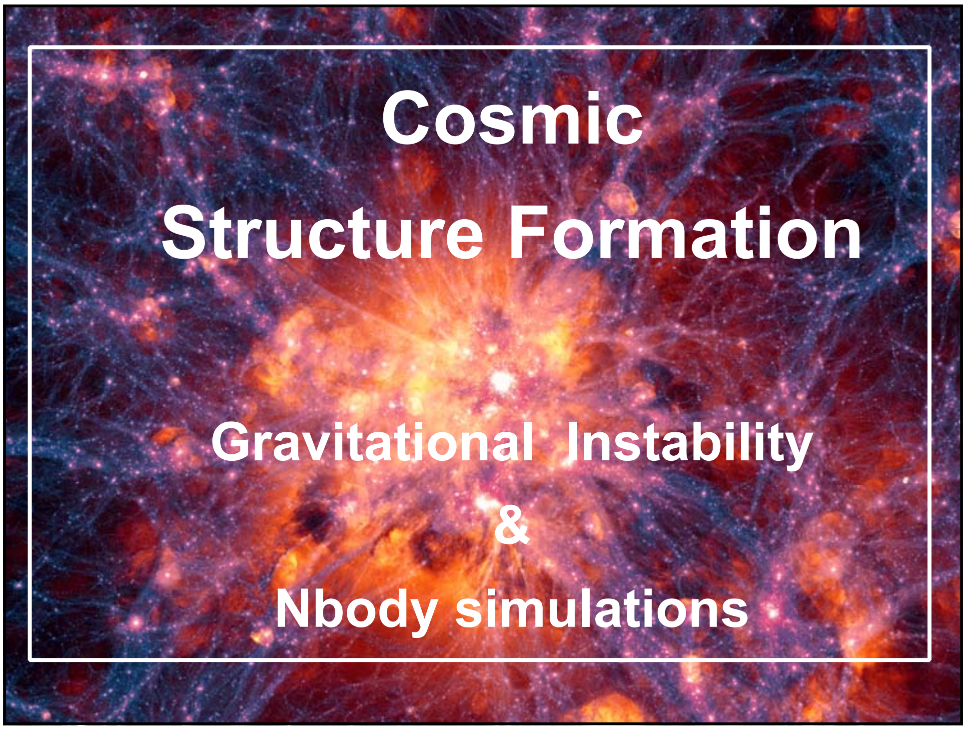
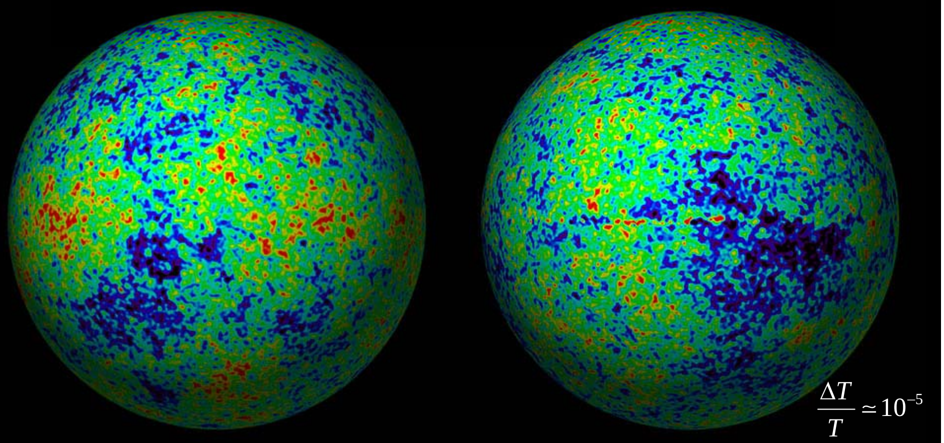
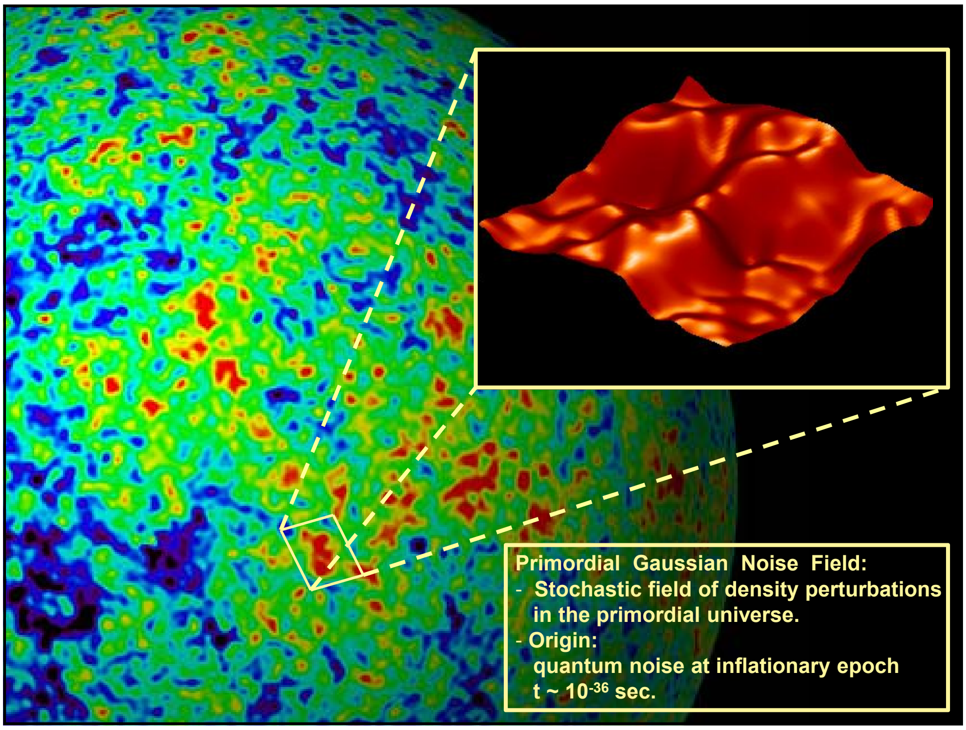


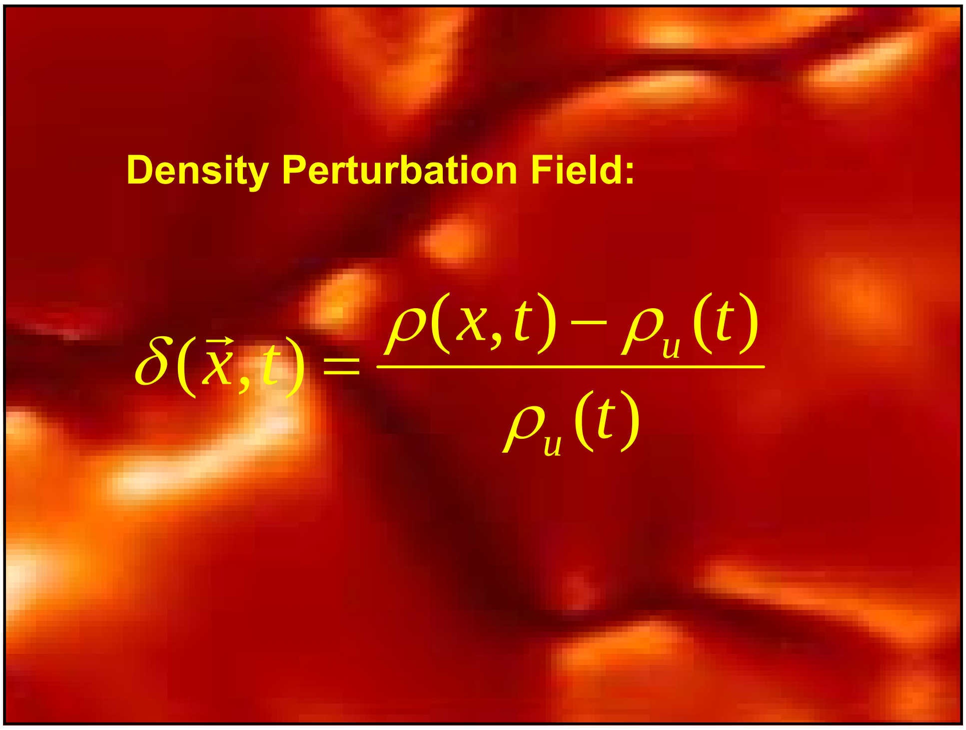
Primordial Universe

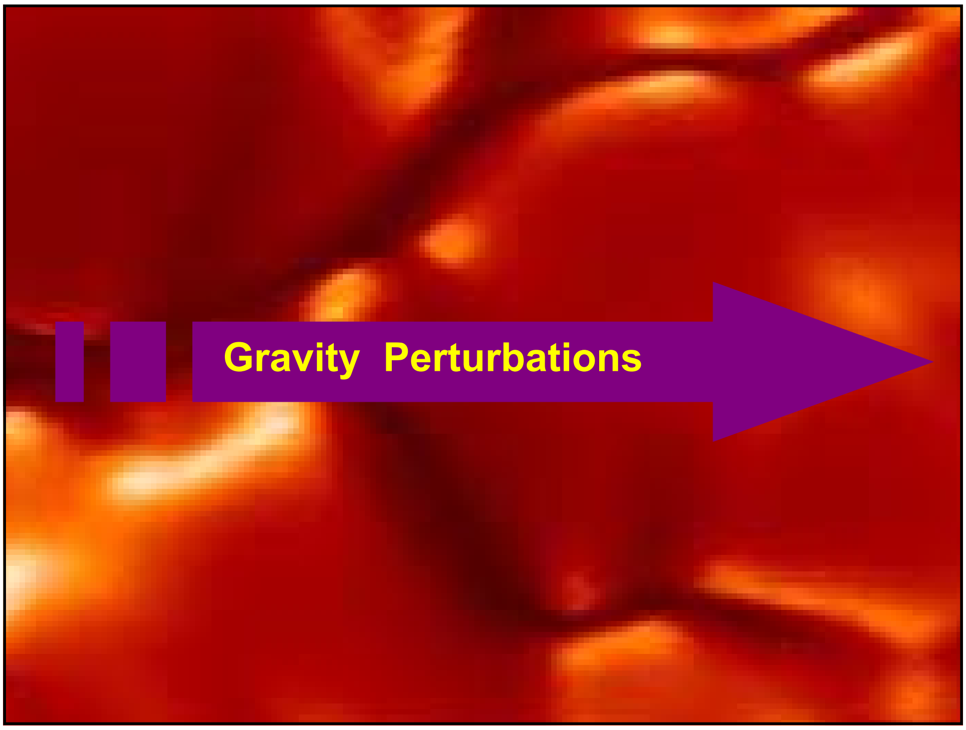


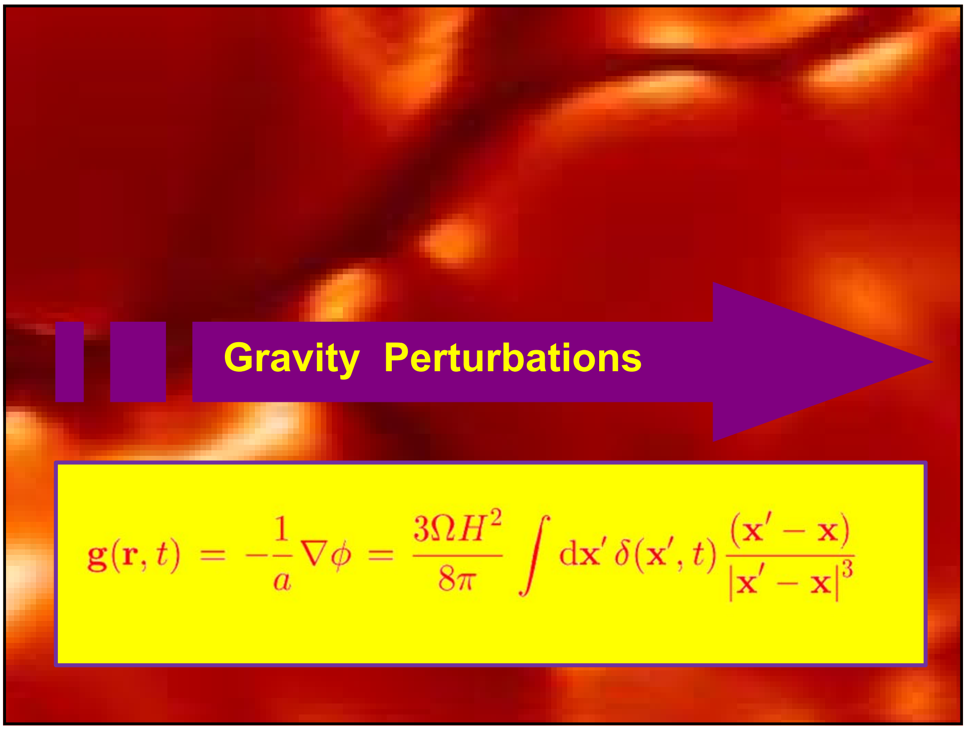
global representation cosmic surface last scattering: the world inside out

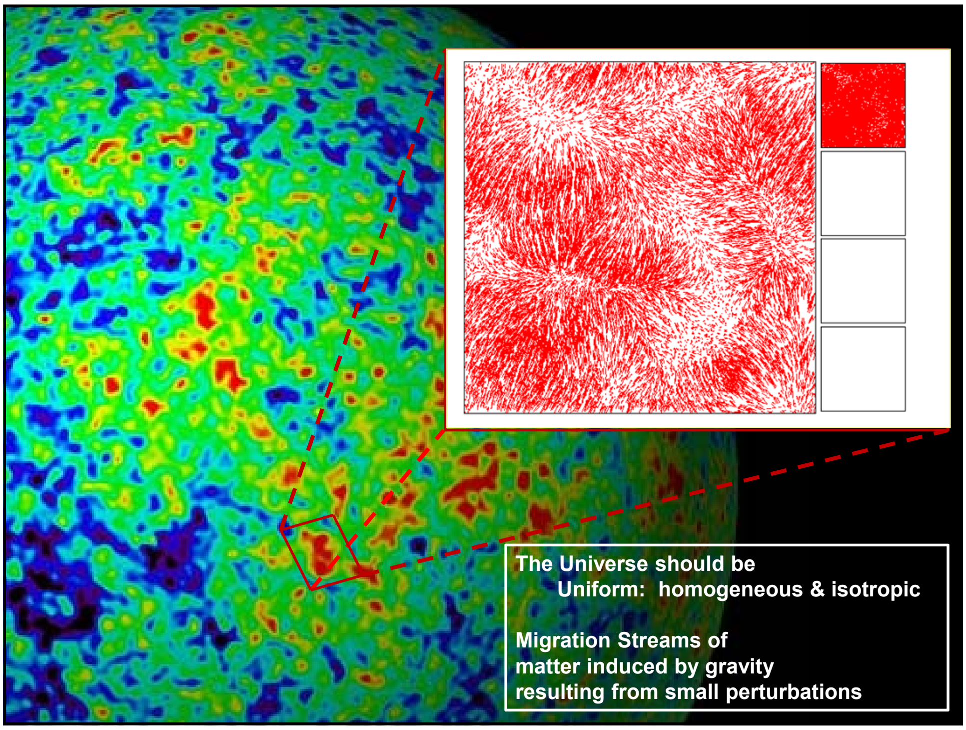
Temperature Map CMB radiation:
Tiny variations in primordial temperature, reflecting tiny inhomogeneities in energy density of Δ~10^{-5} K at recombination epoch, 379,000 yrs after Big Bang


Primordial Gaussian Noise Field:
- Stochastic field of density perturbations in the primordial universe.
- Origin:
quantum noise at inflationary epoch
t ~ 10^{-36} sec.


Density Perturbation Field:
$\delta(\vec{x},t) = \frac{\rho(x,t) - \rho_u(t)}{\rho_u(t)}$


Gravity Perturbations


Gravity Perturbations
$\mathbf{g}(\mathbf{r},t) = -\frac{1}{a}\nabla\phi = \frac{3\Omega H^2}{8\pi}\int \mathrm{d}\mathbf{x}'\,\delta(\mathbf{x}',t)\,\frac{(\mathbf{x}'-\mathbf{x})}{|\mathbf{x}'-\mathbf{x}|^3}$



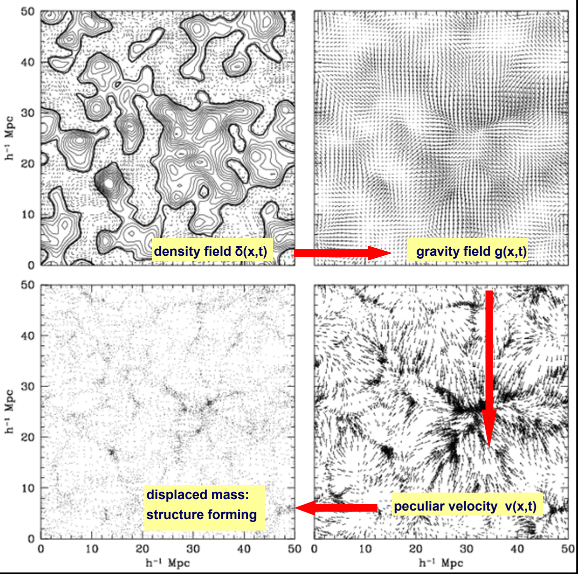
Cosmic Structure Formation

(Energy) Density Perturbations

Gravity Perturbations

(Cosmic) Flows of (Energy) & Matter:
- towards high density regions:
  - assemble more and more matter
  - their expansion comes to a halt
  - turn around and collapse
- evacuating void regions
  - low-density regions expand
  - matter moves out of region
  - turn into prominent empty voids

Emergence of cosmic structures
- Computer Simulations
  - succesfull confrontation with observational reality



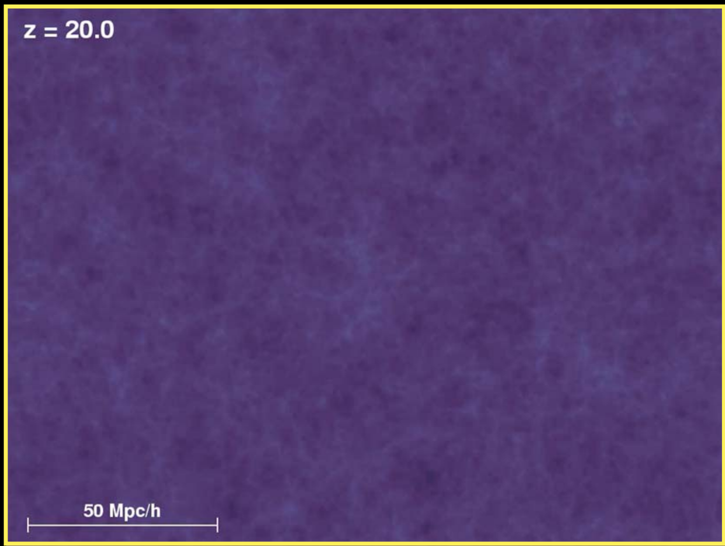
Cosmic Structure Formation

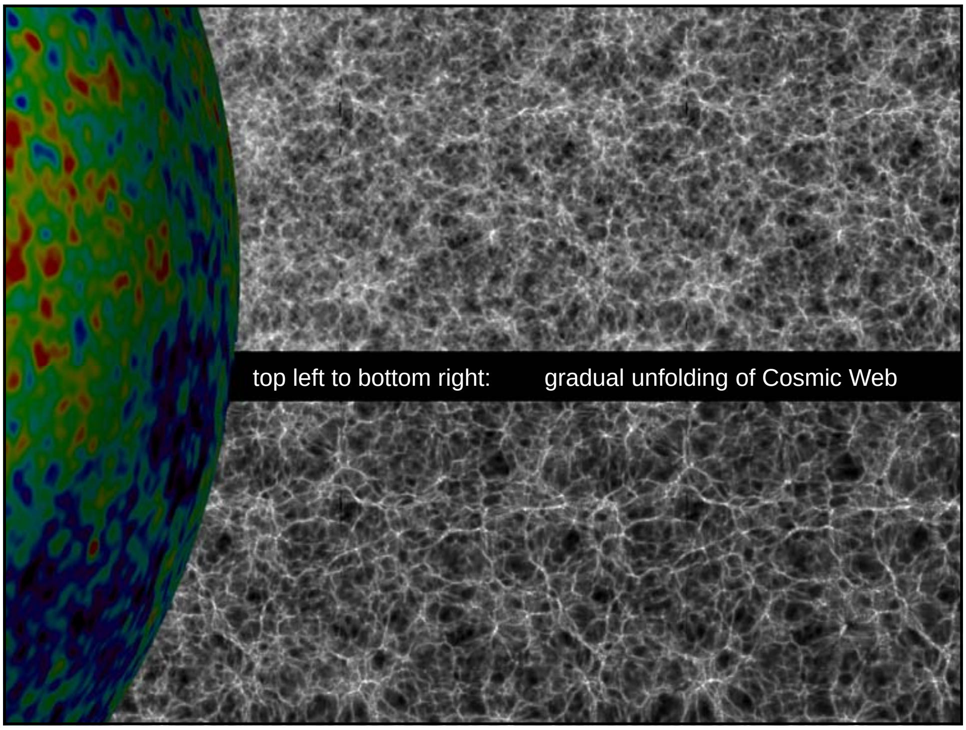
Formation
Cosmic Web:

simulation
sequence

(cold)
dark matter

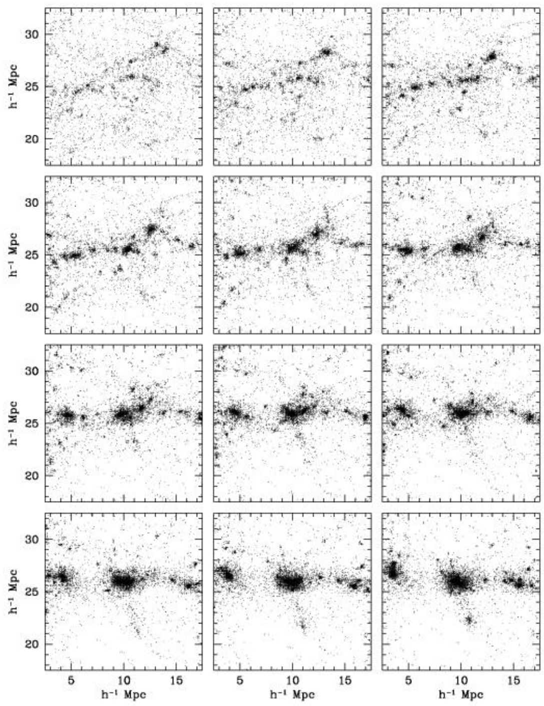
(courtesy:
Virgo/V. Springel).





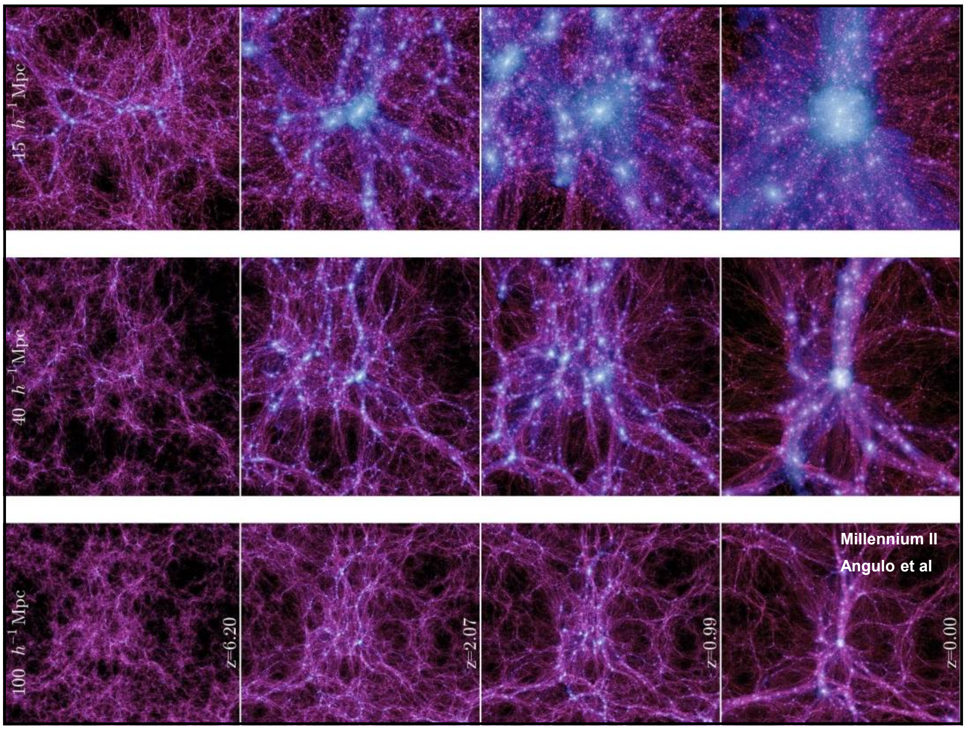
Dynamical Evolution Cosmic Web

- hierarchical structure formation
- anisotropic collapse
- void formation:
  asymmetry
  overdense vs. underdense



Structures in the Universe form
by
gradual hierarchical assembly:

- ❖ **small objects emerge & collapse first,**
- ❖ **then merge with other clumps**
- ❖ **while forming larger objects in hierarchy**



Voids:

Formation & Structure

Void Formation

Void Evolution

an illustration

cosmology:

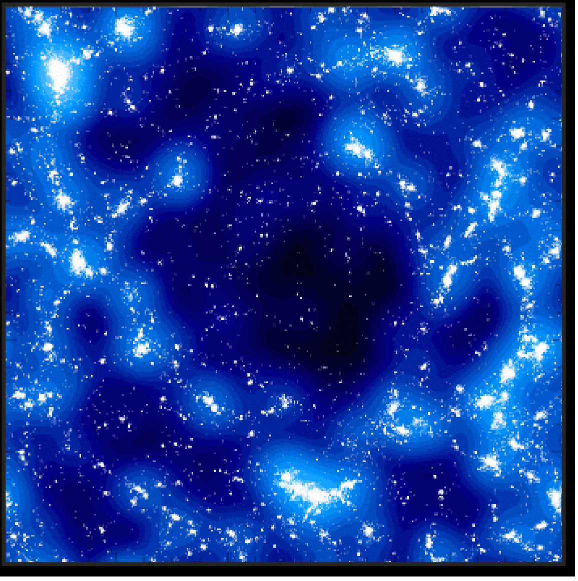
$\Omega_m = 1.0;\quad H_0 = 70\ km/s/Mpc$

initial conditions:

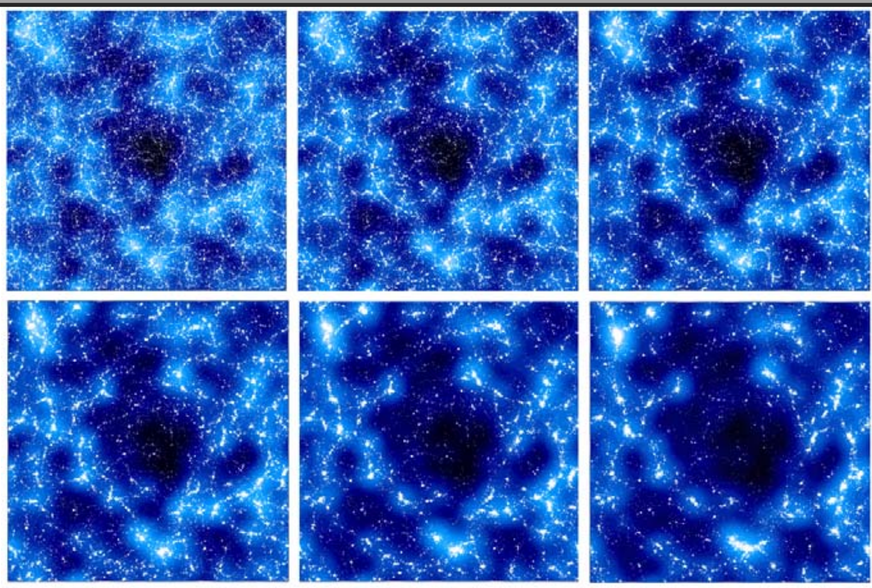
underdensity, Gaussian field

$R_G \sim 4h^{-1}Mpc$

$P(k) \propto k^{-0.5}$



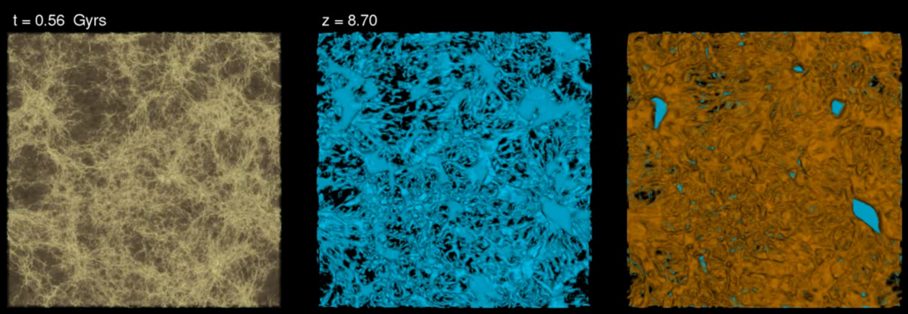
Void Formation



Multiscale Cosmic Web:

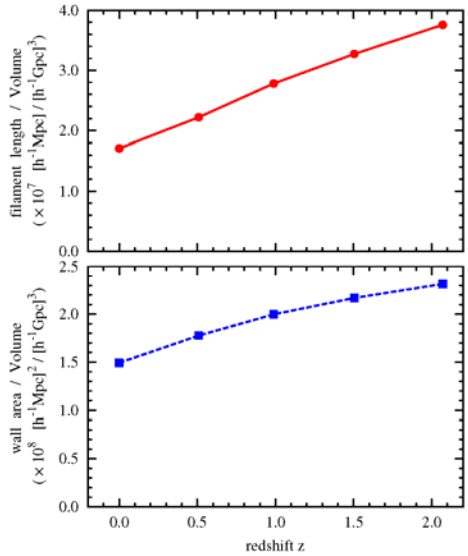
hierarchical evolution

NEXUS/MMF
Evolution Cosmic Web



Cautun et al. 2013

Evolving Filament & Wall Network



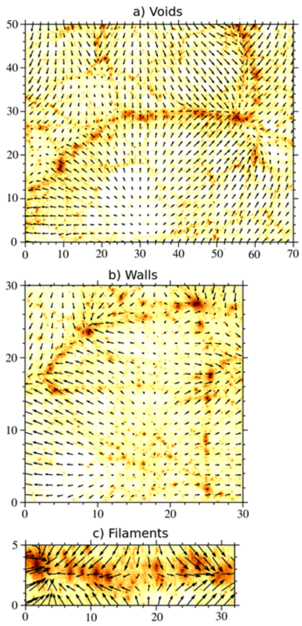
Total length of filament network :

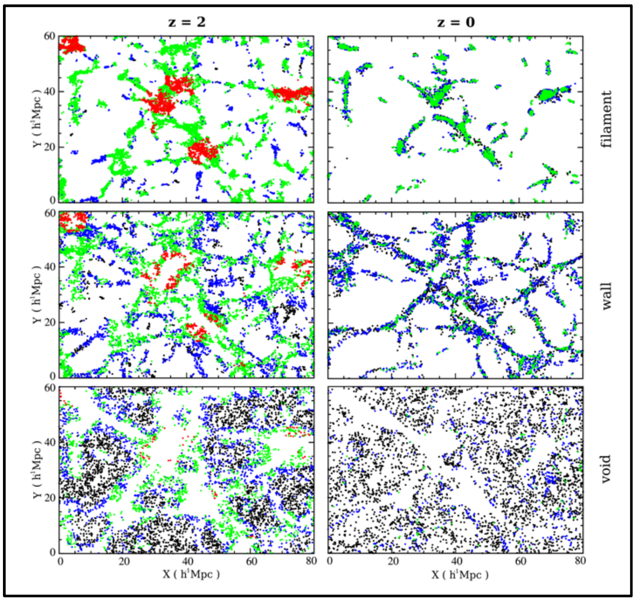
decreasing as a function of time

Total surface area of wall network :

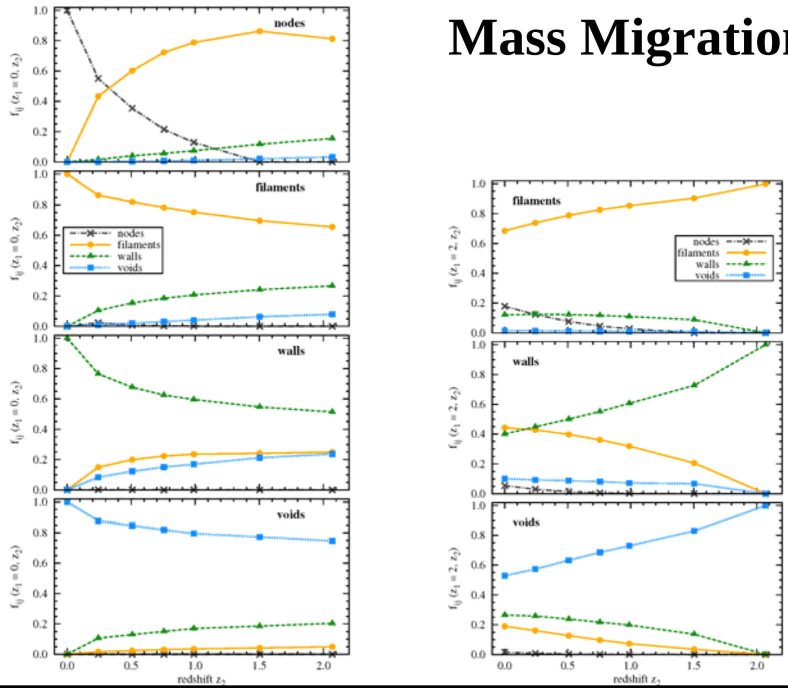
decreasing as a function of time

Web Mass Emigration





Mass Migration



CGV halo & web evolution

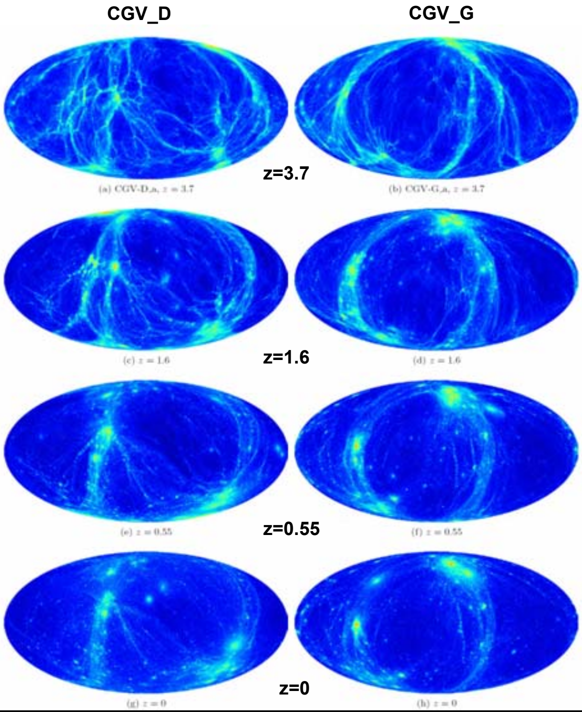
Rieder et al. 2013

CGV:

on walls & filaments

- Mollweide sky projection matter distribution around CGV halos
- CGV halos embedded in walls
- Walls dominate void infrastructure
- substantial fraction in filaments (embedded in walls)
- active dynamical evolution of wall-filament goes along with active void galaxy halo evolution

Rieder et al. 2013

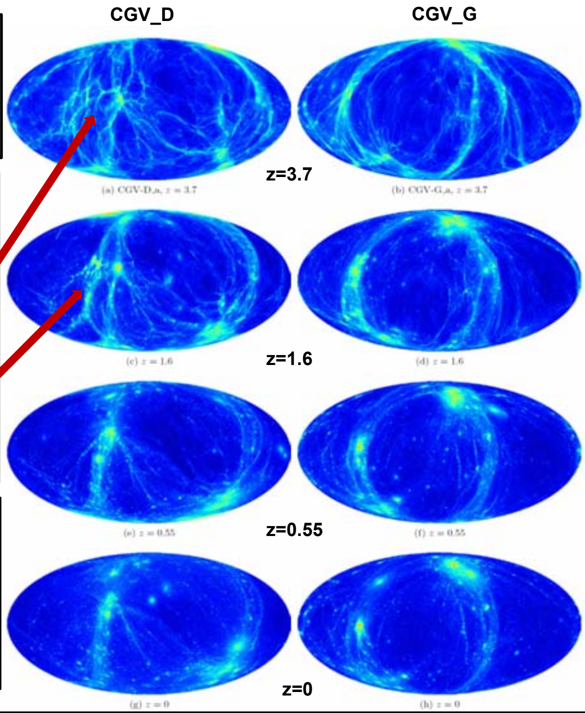


CGV:

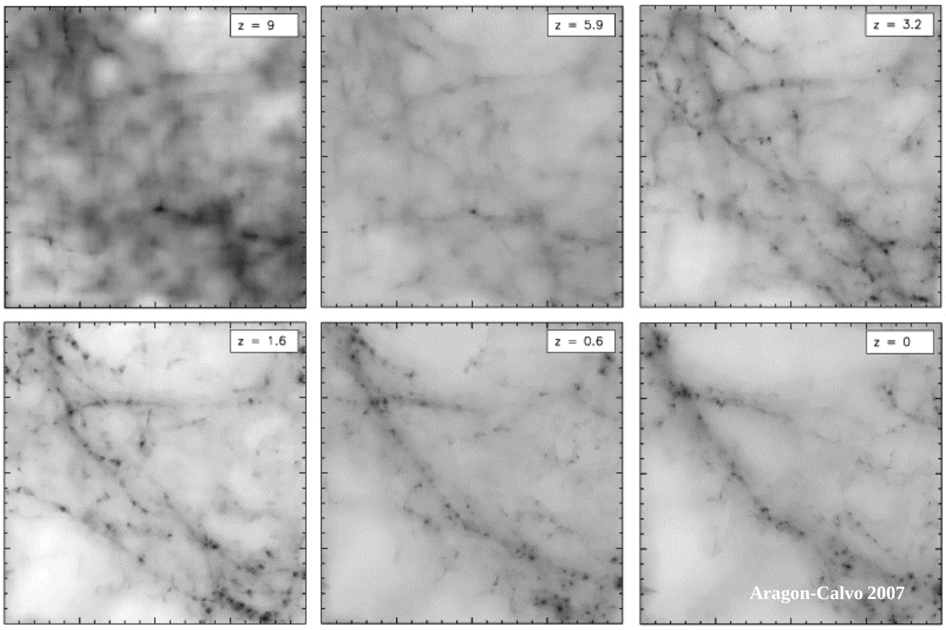
on walls & filaments

- Mollweide sky projection matter distribution around CGV halos
- CGV halos embedded in walls
- Walls dominate void infrastructure
- substantial fraction in filaments (embedded in walls)
- active dynamical evolution of wall-filament goes along with active void galaxy halo evolution

merging system of Intravoid walls



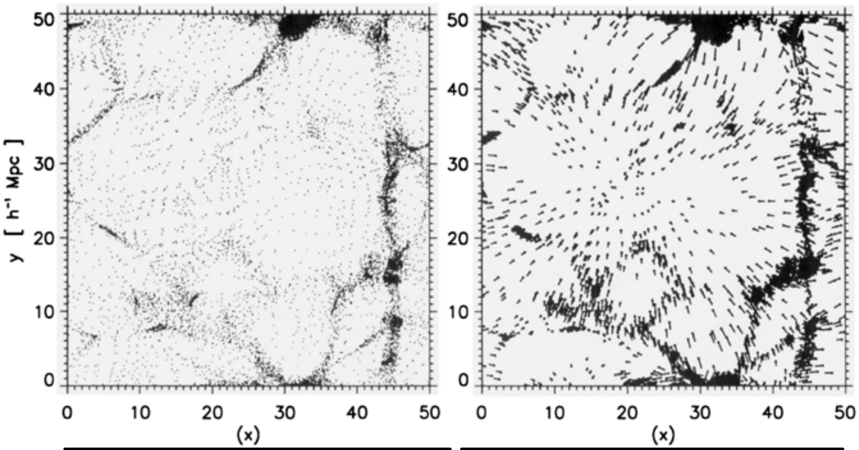
Hierarchical Filament Formation



Void Hierarchy

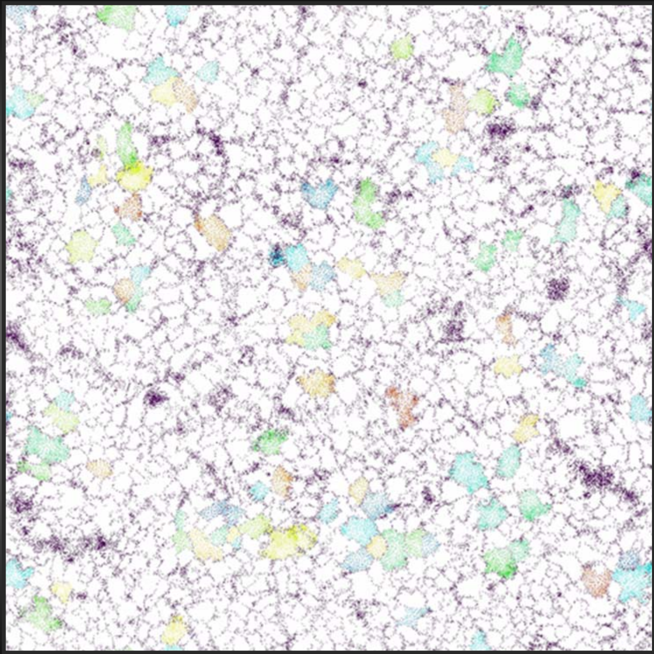
Void formation: - via gradual merging of voids
- demolition of small voids

Dubinski et al. 1993
Van de Weygaert & van Kampen 1993



Constrained Void Simulations:
Van de Weygaert 1991
vdW & van Kampen 1993

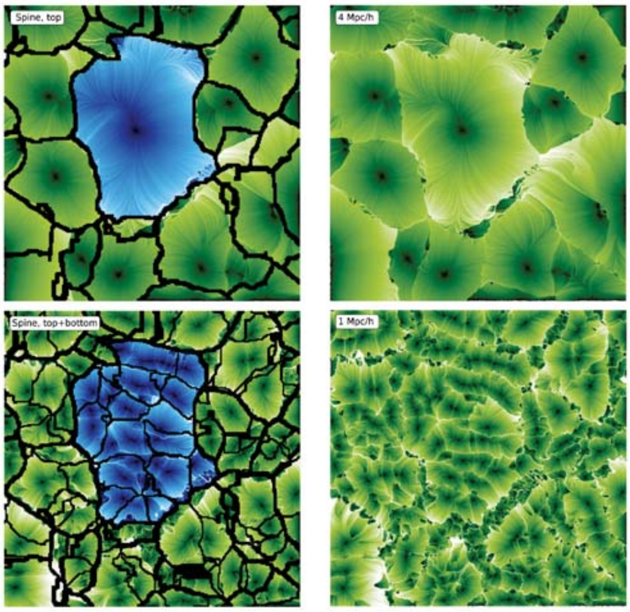
CRF formalism:
Bertschinger 1987
vdW & Bertschinger 1996



Hierarchical Web Evolution:

“Lagrangian” view: development and fate patterns LSS

Platen & vdW 2004



Hierarchical Web Evolution:

Void hierarchy expressed in multiscale structure velocity outflow

Aragon-Calvo & Szalay 2012

Nonlinear Structure Formation

Nbody Simulations

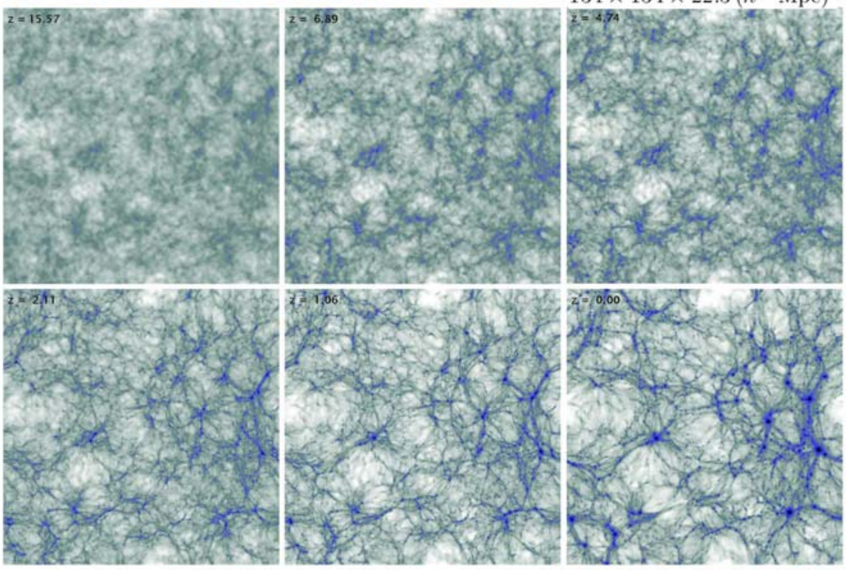
largely based on excellent Potsdam lectures on Nbody simulations (2006)
by V. Springel

Simulations on scales of order 100 Mpc are the workhorses of studies of large-scale structure formation

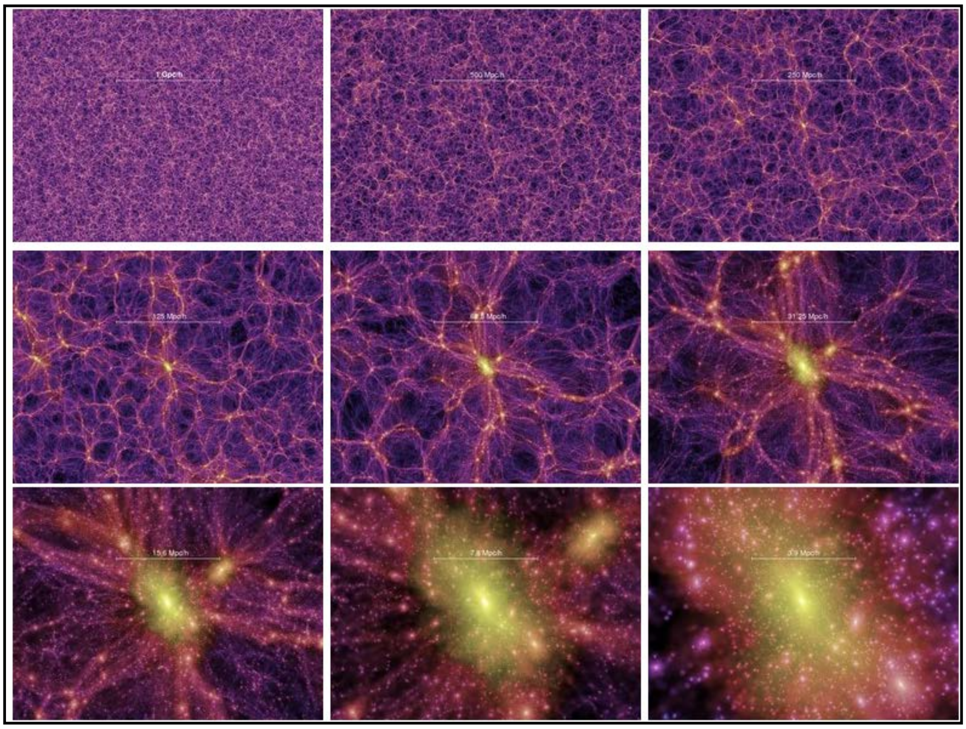
EVOLUTION OF STRUCTURE IN THE GAS DISTRIBUTION

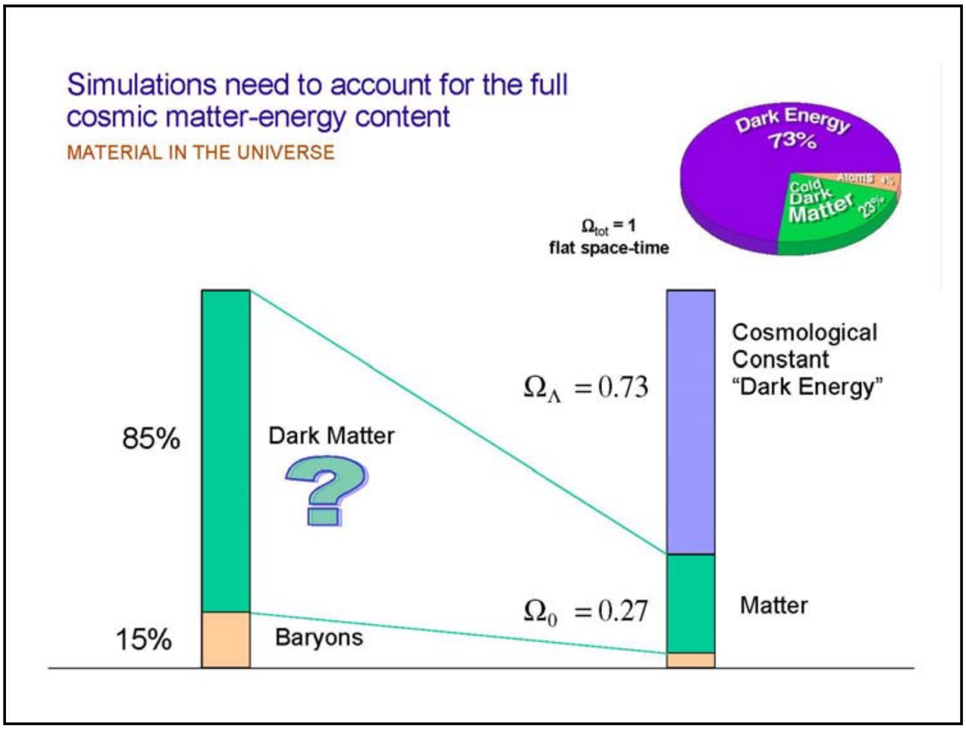
$\Lambda\mathrm{CDM},\ N = 2 \times 224^3$

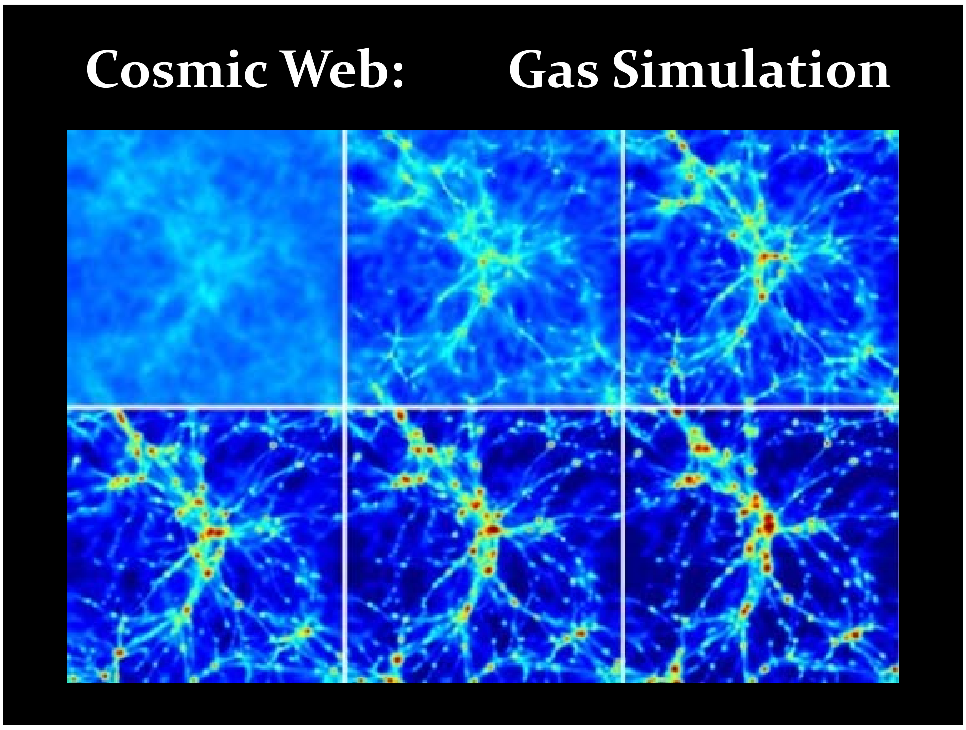
$134 \times 134 \times 22.3\,(h^{-1}\mathrm{Mpc})^3$

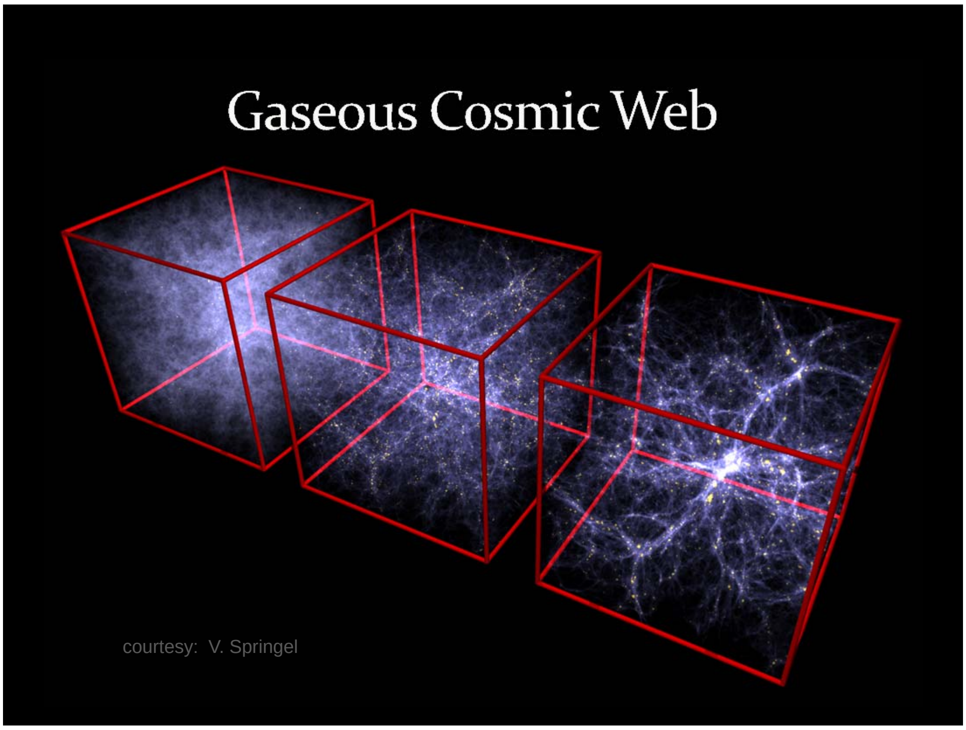


Springel, Hernquist & White (2000)




Simulations need to account for the full
cosmic matter-energy content
MATERIAL IN THE UNIVERSE
Dark Energy 73%
Atoms
Cold Dark Matter 23%
$\Omega_{tot} = 1$
flat space-time
85%
Dark Matter
15%
Baryons
$\Omega_\Lambda = 0.73$
$\Omega_0 = 0.27$
Cosmological
Constant
"Dark Energy"
Matter

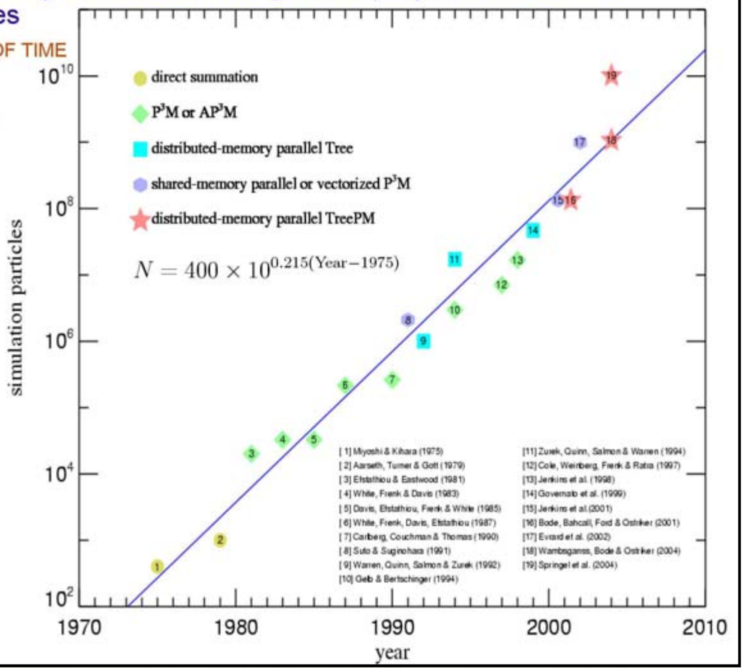

Cosmic Web: Gas Simulation


Gaseous Cosmic Web
courtesy: V. Springel

Cosmological N-body simulations have grown rapidly in size over the last three decades

"N" AS A FUNCTION OF TIME

- Computers double their speed every 18 months (Moore's law)
- N-body simulations have doubled their size every 16-17 months
- Recently, growth has accelerated further.
  The Millennium Run should have become possible in 2010 – we have done it in 2004 !



Nbody Dynamics:

Fundamentals

We assume that the only appreciable interaction of dark matter particles is **gravity**

COLLISIONLESS DYNAMICS

Because there are **so many** dark matter particles, it's best to describe the system in terms of the **single particle distribution function**

$$f = f(\mathbf{x}, \mathbf{v}, t)$$

There are so many dark matter particles that they do not scatter locally on each other, they just respond to their collective gravitational field

Collisionless Boltzmann equation

Poisson-Vlasov System

$$\frac{\mathrm{d}f}{\mathrm{d}t} = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial \mathbf{x}} \cdot \mathbf{v} + \frac{\partial f}{\partial \mathbf{v}} \cdot \left(-\frac{\partial \Phi}{\partial \mathbf{x}}\right) = 0$$

$$\nabla^2 \Phi(\mathbf{x}, t) = 4\pi G \int f(\mathbf{x}, \mathbf{v}, t)\, \mathrm{d}\mathbf{v}$$

Phase-space is conserved along each characteristic (i.e. particle orbit).

The number of stars in galaxies is so large that the two-body relaxation time by far exceeds the Hubble time. Stars in galaxies are therefore also described by the above system.

This system of partial differential equations is very difficult (impossible) to solve directly in non-trivial cases.

The N-body method uses a finite set of particles to sample the underlying distribution function

"MONTE-CARLO" APPROACH TO COLLISIONLESS DYNAMICS

We discretize in terms of N particles, which approximately move along characteristics of the underlying system.

$$\ddot{\mathbf{x}}_i = -\nabla_i\, \Phi(\mathbf{x}_i)$$

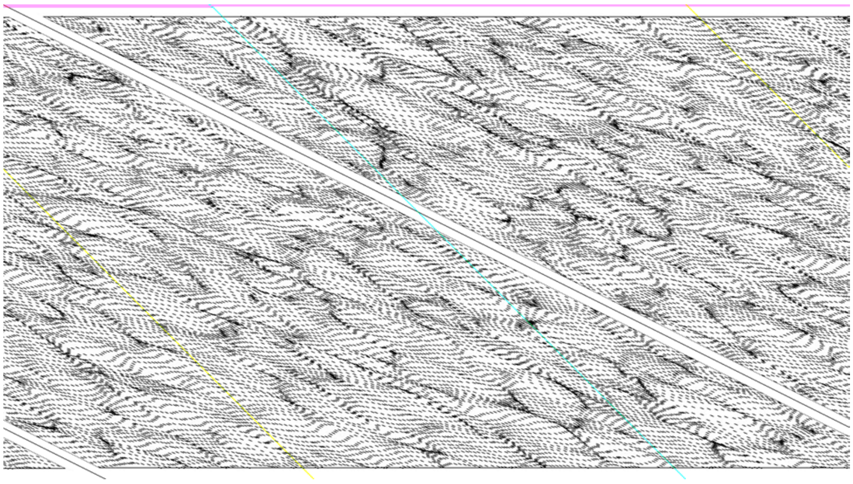
$$\Phi(\mathbf{x}) = -G \sum_{j=1}^{N} \frac{m_j}{[(\mathbf{x} - \mathbf{x}_j)^2 + \epsilon^2]}$$

The need for **gravitational softening**:

- Prevent large-angle particle scatterings and the formation of bound particle pairs.
- Ensure that the two-body relaxation time is sufficiently large.
- Allows the system to be integrated with low-order intergations schemes.

Needed for faithful collisionless behaviour

Particle Simulation



Courtesy: J. Hidding

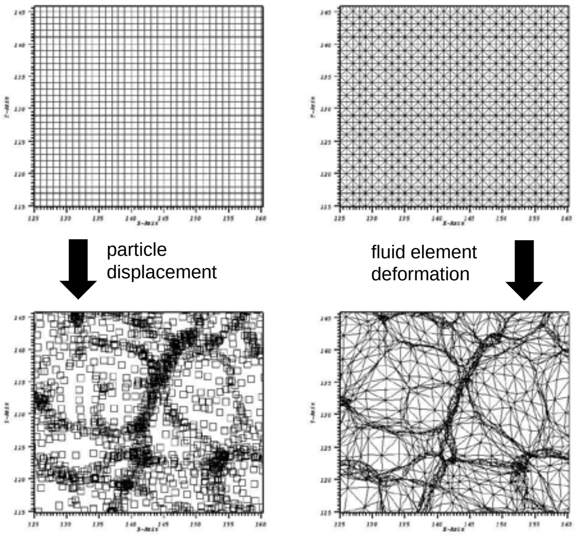
Mass Element Evolution

Courtesy: J. Hidding

Tessellation Deformation & Phase Space Projection

Translation towards Multi-D space:

- Look at deformation of initial tessellation
- each tessellation cell represents matter cell
- evolution deforms cell
- once cells start to overlap, manifestation of different phase-space matter streams



The dynamics of structure formation is driven by gravity

Gravity
general relativity, but Newtonian approximation in expanding space usually sufficient

Hydrodynamics
shock waves
radiation processes
star formation
supernovae,
black holes, etc...

↓

dark matter is collisionless

↓

Monte-Carlo integration as an **N-body system**

↓

3N **coupled**, non-linear differential equations of second order →

Problems:
- N is very large
- All equations are coupled to each other

Two conflicting requirements complicate the study of **hierarchical** structure formation

DYNAMIC RANGE PROBLEM FACED BY COSMOLOGICAL SIMULATIONS

Want **small particle mass** to resolve internal structure of halos

Want **large volume** to obtain respresentative sample of universe

need large N

where N is the particle number

Problems due to a small box size:

- Fundamental mode goes non-linear soon after the first halos form. ⇨ Simulation cannot be meaningfully continued beyond this point.
- No rare objects (the first halo, **rich** galaxy clusters, etc.)

Problems due to a large particle mass:

- Physics cannot be resolved.
- Small galaxies are missed.

At any given time, halos exist on a large range of mass-scales !

Several questions come up when we try to use the N-body approach for cosmological simulations

- How do we compute the gravitational forces efficiently and accurately?
- How do we integrate the orbital equations in time?
- How do we generate appropriate initial conditions?

$$\ddot{\mathbf{x}}_i = -\nabla_i \, \Phi(\mathbf{x}_i)$$

$$\Phi(\mathbf{x}) = -G \sum_{j=1}^{N} \frac{m_j}{[(\mathbf{x}-\mathbf{x}_j)^2 + \epsilon^2]}$$

Note: The naïve computation of the forces is an N^2 - task.

Time Integration Issues

Time integration methods

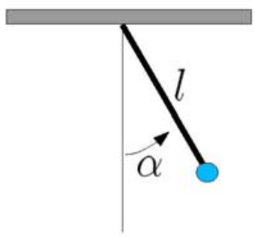
Want to numerically integrate an **ordinary differential equation** (ODE)

$$\dot{\mathbf{y}} = f(\mathbf{y})$$

Note: y can be a vector

Example: Simple pendulum

$$\ddot{\alpha} = -\frac{g}{l}\sin\alpha$$



$$y_0 \equiv \alpha \qquad y_1 \equiv \dot{\alpha}$$

$$\rightarrow \dot{\mathbf{y}} = f(y) = \begin{pmatrix} y_1 \\ -\frac{g}{l}\sin y_0 \end{pmatrix}$$

A numerical approximation to the ODE is a set of values $\{\mathbf{y}_0, \mathbf{y}_1, \mathbf{y}_2, \ldots\}$ at times $\{t_0, t_1, t_2, \ldots\}$

There are many different ways for obtaining this.

Explicit Euler method

$$y_{n+1} = y_n + f(y_n)\Delta t$$

- Simplest of all
- Right hand-side depends only on things already non, **explicit method**
- The error in a single step is $O(\Delta t^2)$, but for the N steps needed for a finite time interval, the total error scales as $O(\Delta t)$!
- Never use this method, it's only **first order accurate**.

Implicit Euler method

$$y_{n+1} = y_n + f(y_{n+1})\Delta t$$

- **Excellent** stability properties
- Suitable for very stiff ODE
- Requires implicit solver for y_{n+1}

Implicit mid-point rule

$$y_{n+1} = y_n + f\left(\frac{y_n + y_{n+1}}{2}\right)\Delta t$$

- **2nd order accurate**
- Time-symmetric, in fact **symplectic**
- But still implicit...

Runge-Kutta methods

whole class of integration methods

2nd order accurate

$$\begin{aligned} k_1 &= f(y_n) \\ k_2 &= f(y_n + k_1\Delta t) \\ y_{n+1} &= y_n + \left(\frac{k_1 + k_2}{2}\right)\Delta t \end{aligned}$$

4th order accurate.

$$\begin{aligned} k_1 &= f(y_n, t_n) \\ k_2 &= f(y_n + k_1\Delta t/2, t_n + \Delta t/2) \\ k_3 &= f(y_n + k_2\Delta t/2, t_n + \Delta t/2) \\ k_4 &= f(y_n + k_3\Delta t/2, t_n + \Delta t) \\ y_{n+1} &= y_n + \left(\frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6}\right)\Delta t \end{aligned}$$

The Leapfrog

For a second order ODE: $\ddot{\mathbf{x}} = f(\mathbf{x})$

"Drift-Kick-Drift" version

$$\begin{aligned} x_{n+\frac{1}{2}} &= x_n + v_n \frac{\Delta t}{2} \\ v_{n+1} &= v_n + f(x_{n+\frac{1}{2}})\Delta t \\ x_{n+1} &= x_{n+\frac{1}{2}} + v_{n+1}\frac{\Delta t}{2} \end{aligned}$$

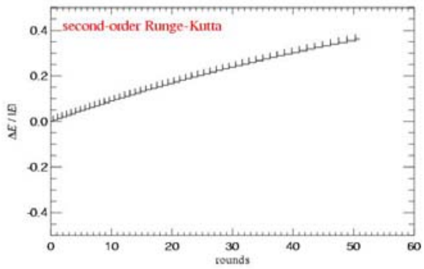
"Kick-Drift-Kick" version

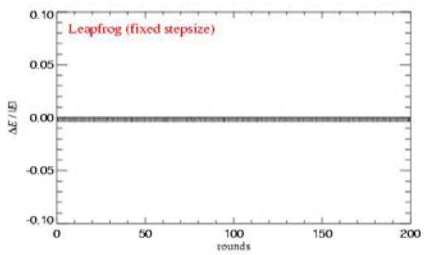
$$\begin{aligned} v_{n+\frac{1}{2}} &= v_n + f(x_n)\frac{\Delta t}{2} \\ x_{n+1} &= x_n + v_{n+\frac{1}{2}}\frac{\Delta t}{2} \\ v_{n+1} &= v_{n+\frac{1}{2}} + f(x_{n+1})\frac{\Delta t}{2} \end{aligned}$$

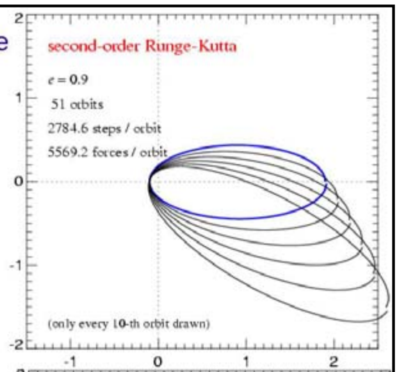
- **2^{nd} order accurate**
- **symplectic**
- can be rewritten into time-centred formulation

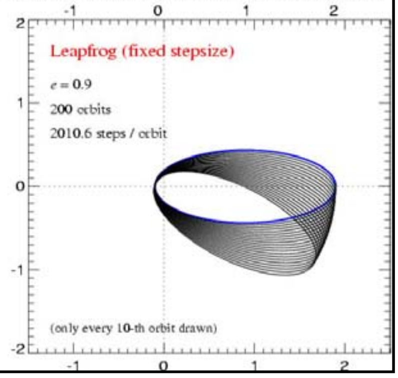
When compared with an integrator of the same order, the leapfrog is highly superior

INTEGRATING THE KEPLER PROBLEM









Nbody Dynamics:

Historical

THE ASTROPHYSICAL JOURNAL

AN INTERNATIONAL REVIEW OF SPECTROSCOPY AND ASTRONOMICAL PHYSICS

VOLUME 94 NOVEMBER 1941 NUMBER 3

ON THE CLUSTERING TENDENCIES AMONG THE NEBULAE

II. A STUDY OF ENCOUNTERS BETWEEN LABORATORY MODELS OF STELLAR SYSTEMS BY A NEW INTEGRATION PROCEDURE

ERIK HOLMBERG

ABSTRACT

In a previous paper[1] the writer discussed the possibility of explaining the observed clustering effects among extragalactic nebulae as a result of captures. The present investigation deals with the important problem of whether the loss of energy resulting from the tidal disturbances at a close encounter between two nebulae is large enough to effect a capture. The tidal deformations of two models of stellar systems, passing each other at a small distance, are studied by reconstructing, piece by piece, the orbits described by the individual mass elements. The difficulty of integrating the total gravitational force acting upon a certain element at a certain point of time is solved by replacing gravitation by light. The mass elements are represented by light-bulbs, the candle power being proportional to mass, and the total light is measured by a photocell (Fig. 1). The nebulae are assumed to have a flattened shape, and each is represented by 37 light-bulbs. It is found that the tidal deformations cause an increase in the attraction between the two objects, the increase reaching its maximum value when the nebulae are separating, i.e., after the passage. The resulting loss of energy (Fig. 6) is comparatively large and may, in favorable cases, effect a capture. The spiral arms developing during the encounter (Figs. 4) represent an interesting by-product of the investigation. The direction of the arms depends on the direction of rotation of the nebulae with respect to the direction of their space motions.

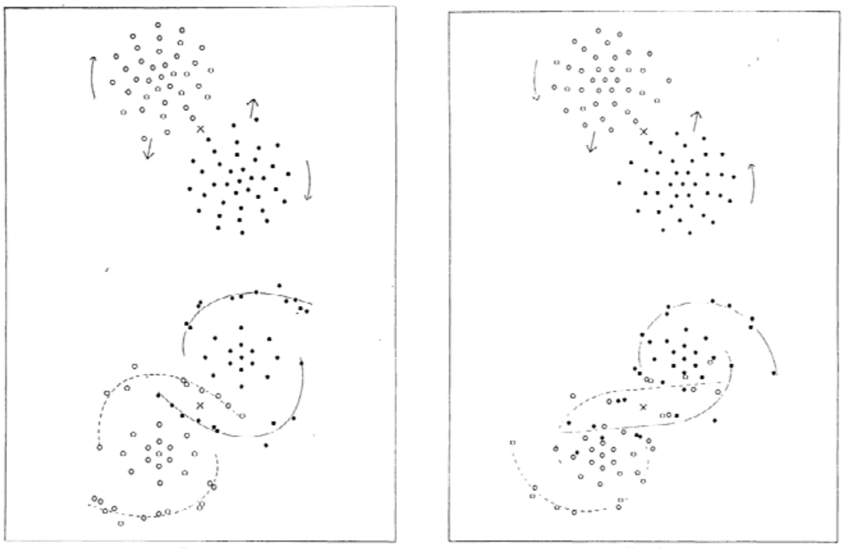


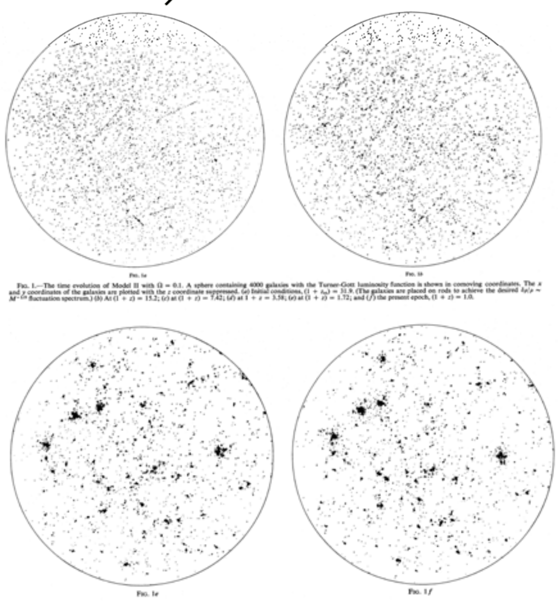
FIG. 4a FIG. 4b

FIG. 4a.—Tidal deformations corresponding to parabolic motions, clockwise rotations, and a distance of closest approach equal to the diameters of the nebulae. The spiral arms point in the direction of the rotation.

FIG. 4b.—Same as above, with the exception of counterclockwise rotations. The spiral arms point in the direction opposite to the rotation.

1979: Aarseth, Turner & Gott

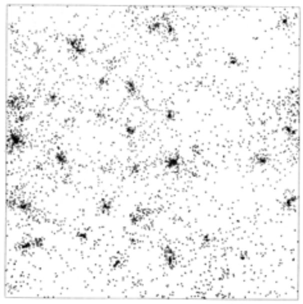
- 1000 particles in spherical volume
- initial conditions: random/semi-random

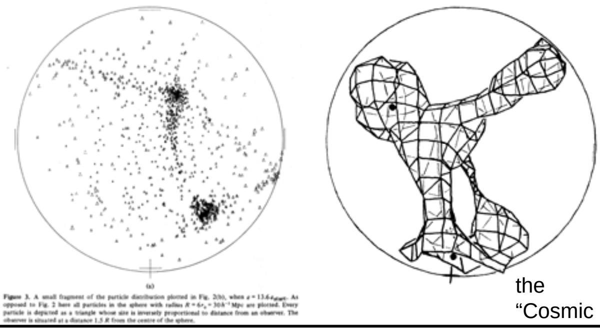


1983: Klypin & Shandarin

- 32^3 particle in cubic volume
- initial conditions: 1st time proper cosmological Gaussian conditions:

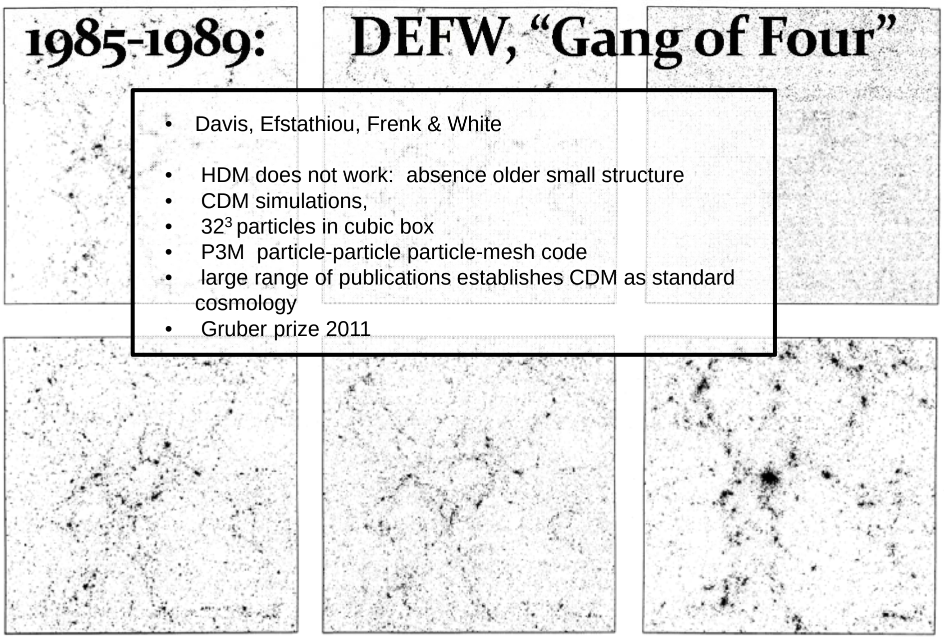
  Zeldovich approximation
- PM particle-mesh simulation





the "Cosmic Chicken"

1985-1989: DEFW, "Gang of Four"



- Davis, Efstathiou, Frenk & White
- HDM does not work: absence older small structure
- CDM simulations,
- 32^3 particles in cubic box
- P3M particle-particle particle-mesh code
- large range of publications establishes CDM as standard cosmology
- Gruber prize 2011

1985-1989: DEFW, "Gang of Four"

- Davis, Efstathiou, Frenk & White

- HDM does not work: absence older small structure
- CDM simulations,
- 32^3 particles in cubic box
- P3M particle-particle particle-mesh code
- large range of publications establishes CDM as standard cosmology
- Gruber prize 2011

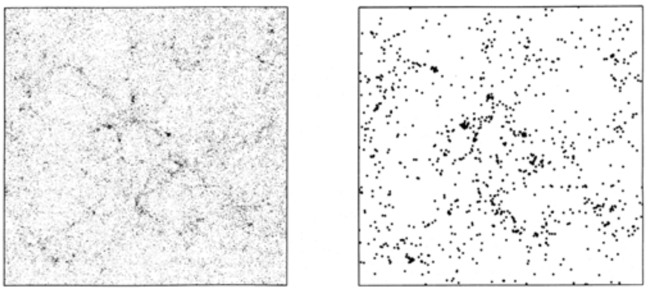


FIG. 16.—The projected distribution of all particles (*left*) and of the "galaxies" (*right*) in EdS1 at $a = 1.4$. The side of the box is $32.5h^{-1}$ Mpc. "Galaxies" are assumed to form only at the 2.5 σ peaks of the linear density distribution.

1985-1989: DEFW, "Gang of Four"

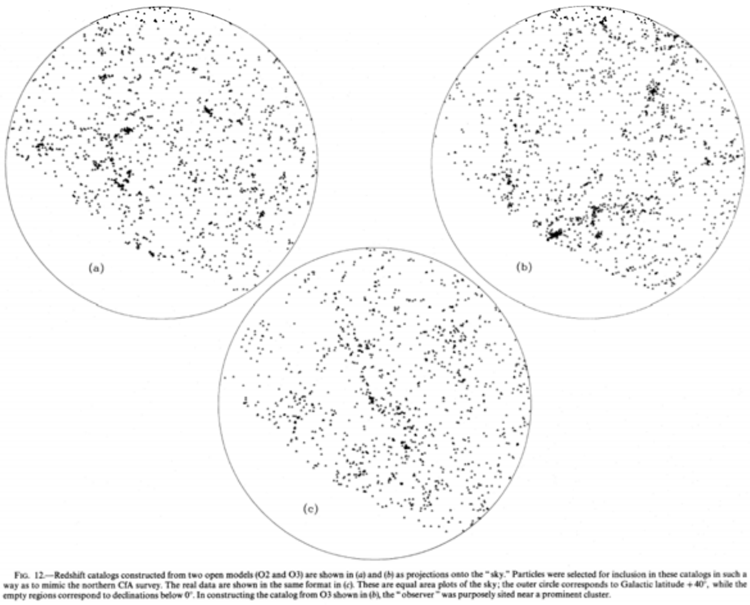
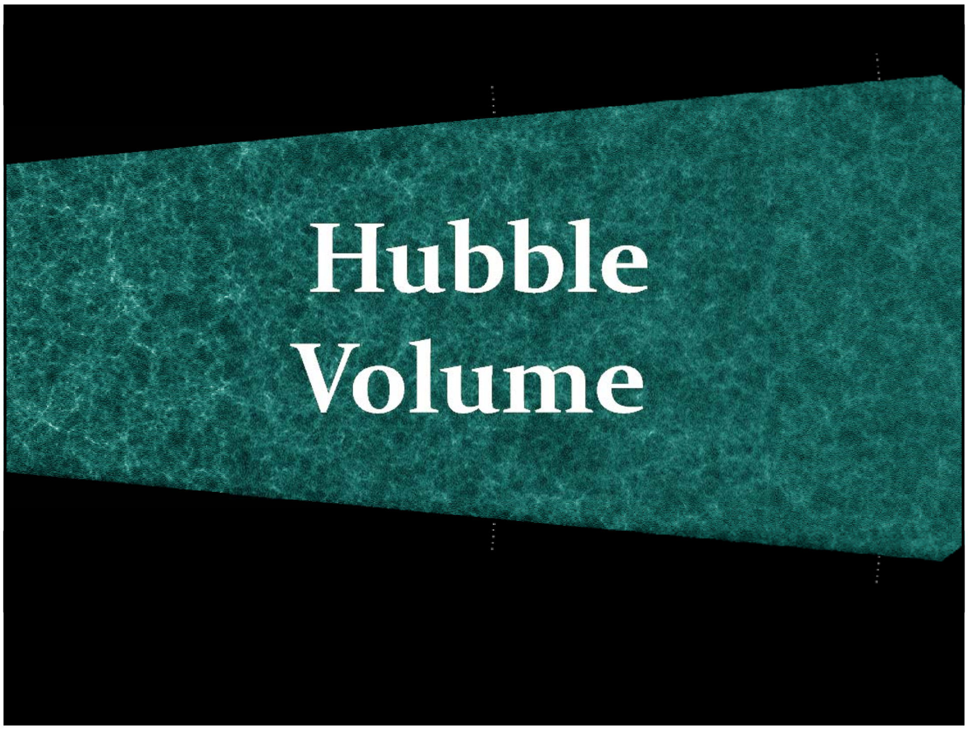
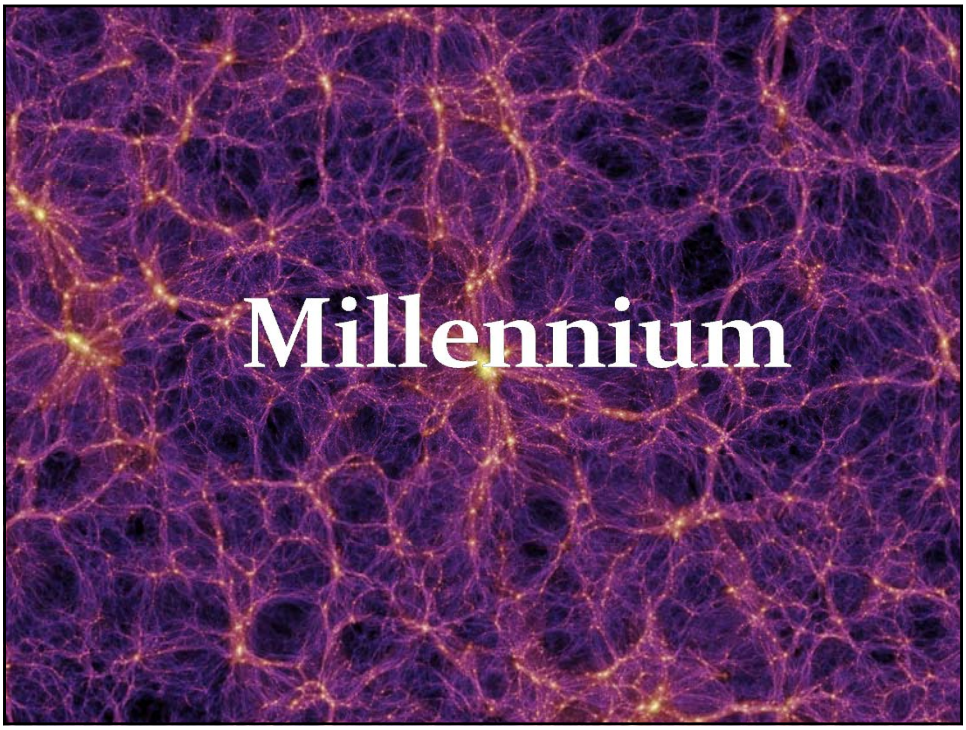
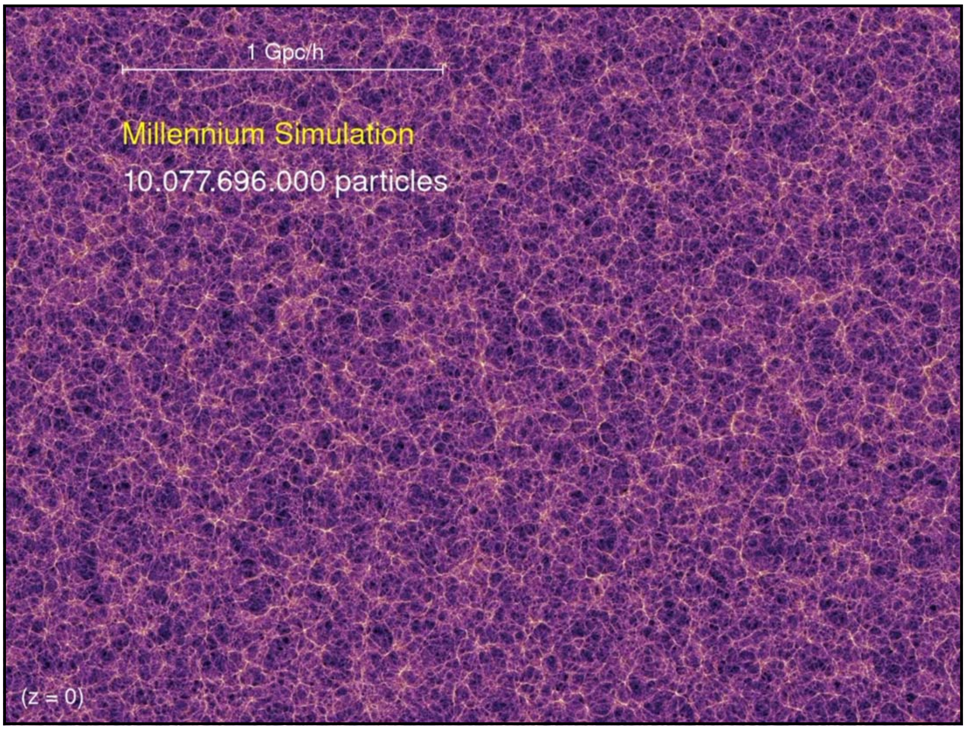


FIG. 12.—Redshift catalogs constructed from two open models (O2 and O3) are shown in (*a*) and (*b*) as projections onto the "sky." Particles were selected for inclusion in these catalogs in such a way as to mimic the northern CfA survey. The real data are shown in the same format in (*c*). These are equal area plots of the sky; the outer circle corresponds to Galactic latitude +40°, while the empty regions correspond to declinations below 0°. In constructing the catalog from O3 shown in (*b*), the "observer" was purposely sited near a prominent cluster.


Hubble
Volume


Millennium



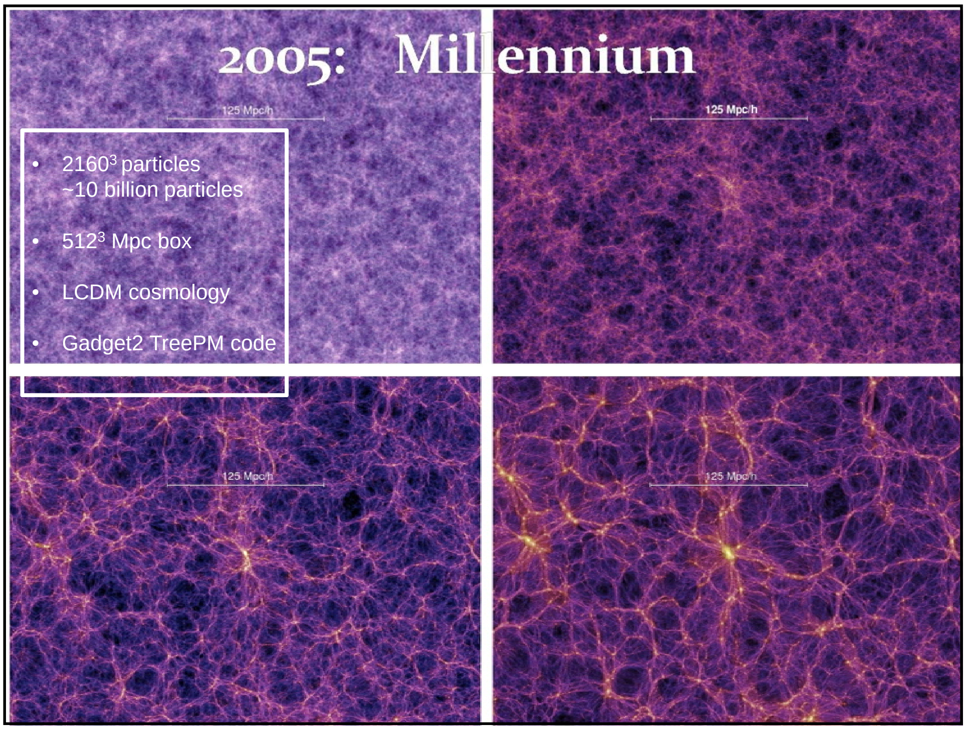
Millennium Simulation

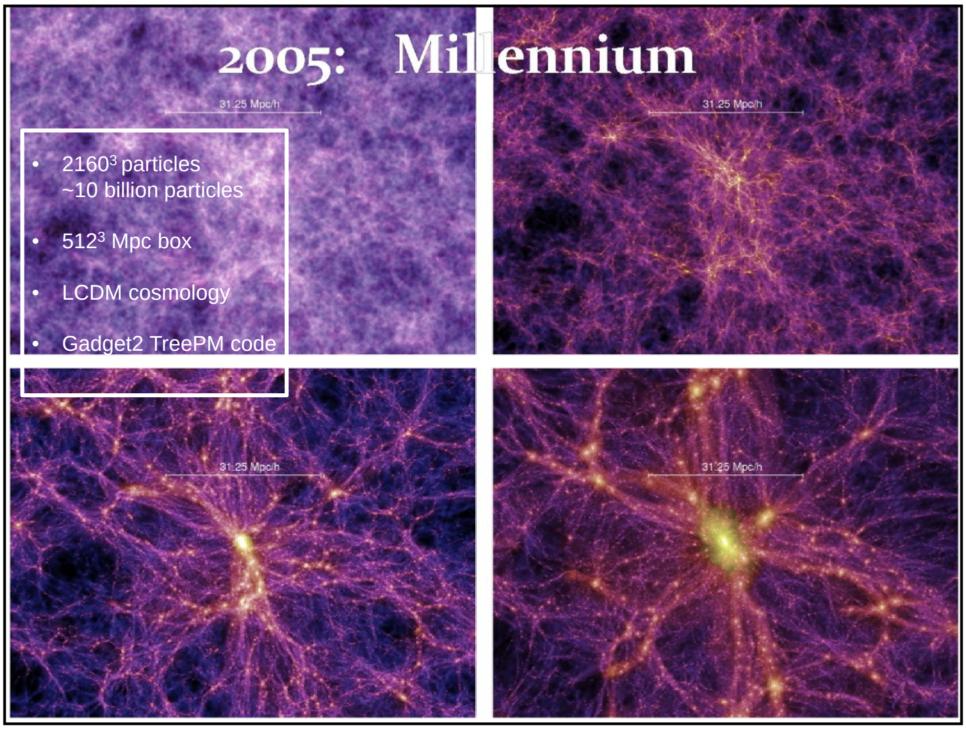
- Run: 2005
- Virgo consortium: UK-Germany centered European consortium
- 2005-2014: 650 publications based on Millennium

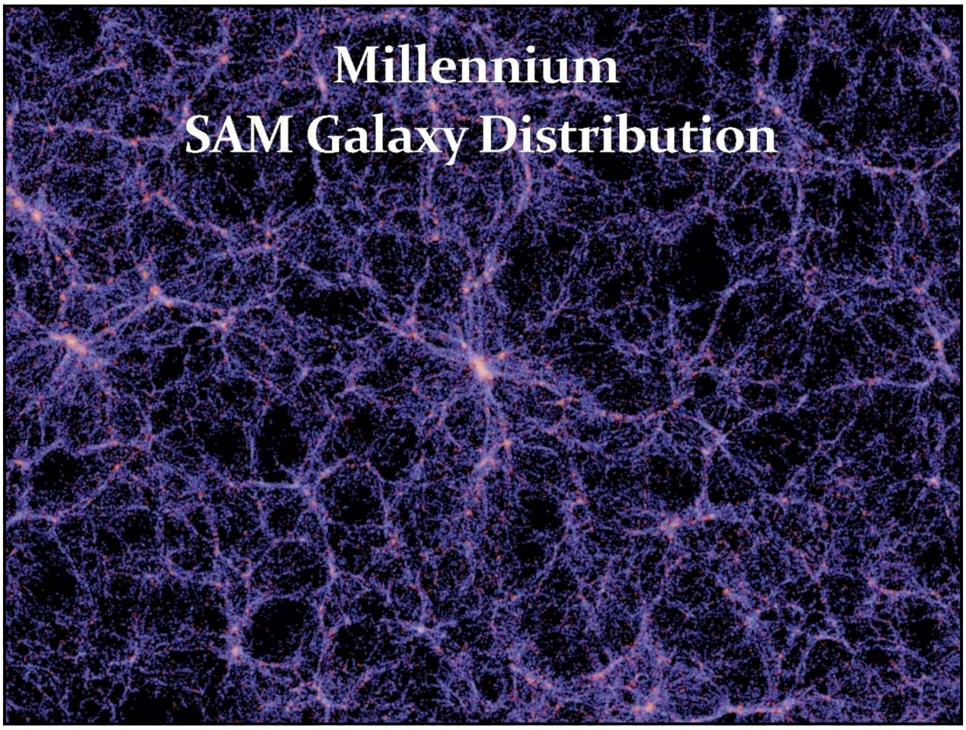
- Nbody simulation, TreePM (Gadget2)
- LCDM cosmology
- 2160^3 particles ~ 10×10^{10} particles
- Volume: cubic region $500h^{-1}$ Mpc
- Resolution: $5h^{-1}$ kpc
- Dynamic range: 10^5 per dimension
- Data: 25 Terabyte

- Galaxy modelling by semi-analytical modelling
- 2×10^7 galaxies with mass > LMC

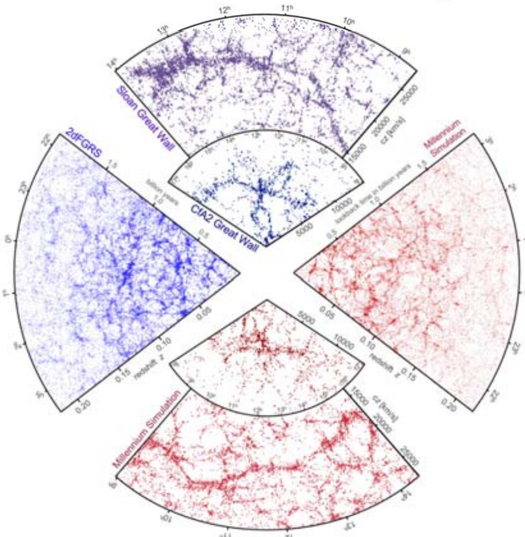
- Public Database:
  VO-oriented and SQL-queryable database (G. Lemson)


2005: Millennium
125 Mpc/h
• 2160^3 particles
~10 billion particles
• 512^3 Mpc box
• LCDM cosmology
• Gadget2 TreePM code


2005: Millennium
31.25 Mpc/h
• 2160^3 particles
~10 billion particles
• 512^3 Mpc box
• LCDM cosmology
• Gadget2 TreePM code


Millennium
SAM Galaxy Distribution

Millennium
SAM Galaxy Distribution



Semi-analytical models (SAM)
of galaxy formation:

DM particle distribution

⇩

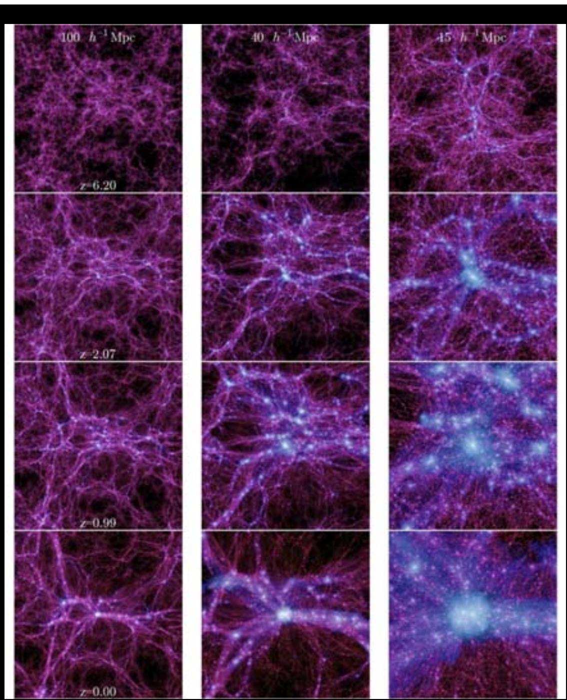
DM halos (clumps):
collapsed & virialized clumps

⇩

Complex (> 18 parameter) SAM models use halos to populate these with galaxies

- Morphology, Luminosity, Colour etc. of model galaxies dependent on
  - mass halos
  - merger rate and history
  - and other factors

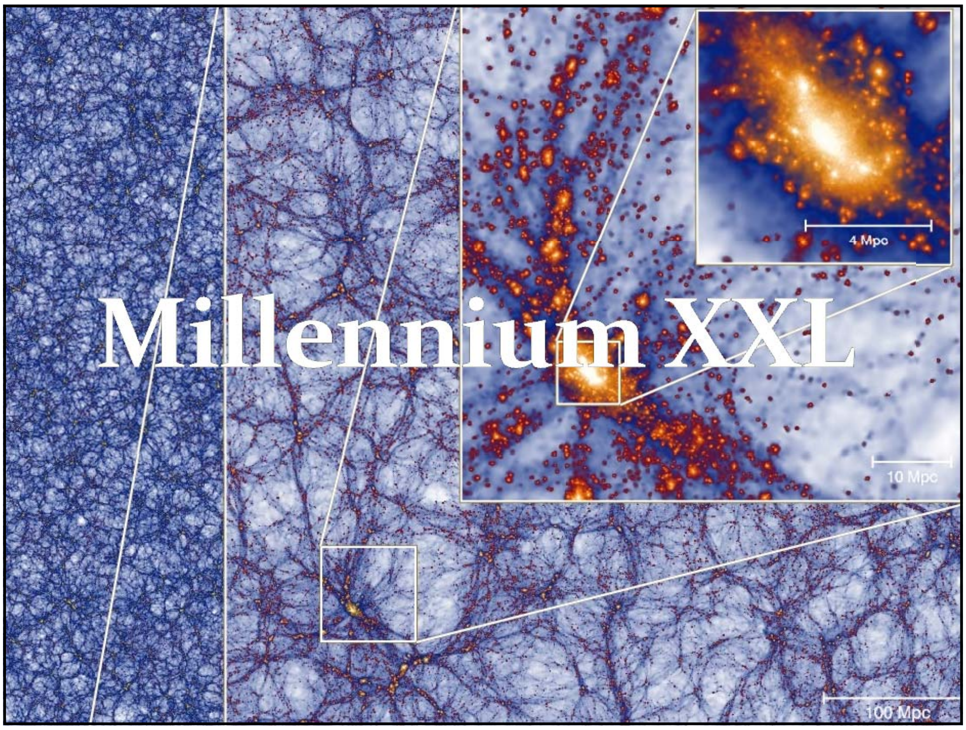
Figure:
Comparison Millennium simulation SDSS, 2dFGRS, CfA galaxy surveys



Millennium-II

- Run: 2009
- Virgo consortium:

- Nbody simulation, TreePM (Gadget)
- LCDM cosmology
- 2160^3 particles ~ 10 x 10^{10} particles
- Volume: cubic region $100h^{-1}$ Mpc
- Resolution: $1h^{-1}$ kpc
- Dynamic range: 10^5 per dimension
- Particle mass: 6.89 x 10^6 $M_\odot/h$

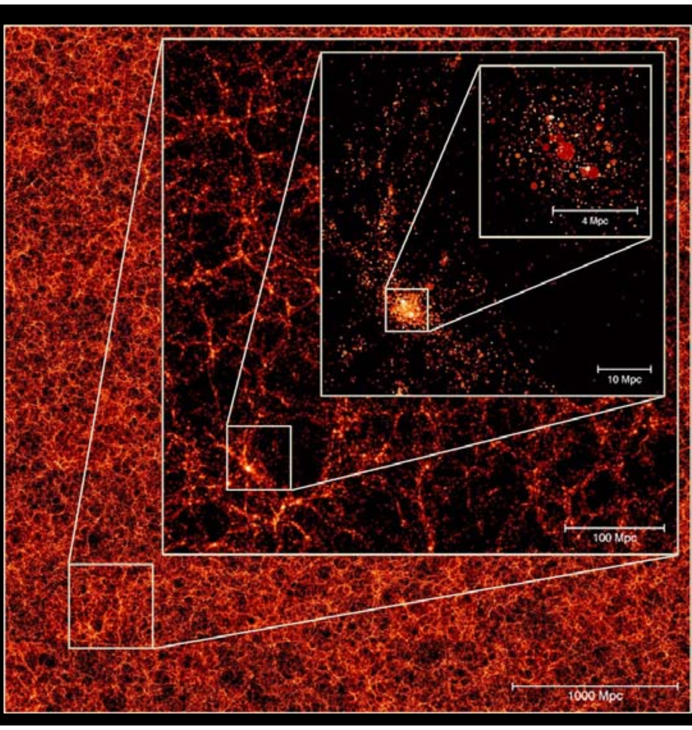


Millennium XXL Simulation

- Run: 2010
- Virgo consortium: UK-Germany centered European consortium
- First multi – 100 billion particle simulation

- Nbody simulation, TreePM (Gadget2)
- LCDM cosmology
- 6720^3 particles ~ 300 x 10^{10} particles
- Volume: cubic region 4.1 Gpc
  216 volume Millennium
  27000 volumes Millennium-II
- Run on JUROPA supercomputer,
  12,000 cores, 300 years CPU time, 30 Terabyte RAM
- Data: 100 Terabyte

- Galaxy modelling by semi-analytical modelling
- 700 x 10^6 galaxies at z=0



PM:

Particle-Mesh

The particle-mesh method

Poisson's equation can be solved in real-space by a convolution of the density field with a Green's function.

$$\Phi(\mathbf{x}) = \int g(\mathbf{x} - \mathbf{x}')\, \rho(\mathbf{x})\, \mathrm{d}\mathbf{x}'$$

Example for vacuum boundaries:

$$\Phi(\mathbf{x}) = -G \int \frac{\rho(\mathbf{x})}{|\mathbf{x} - \mathbf{x}'|}\, \mathrm{d}\mathbf{x}' \qquad g(\mathbf{x}) = -\frac{G}{|\mathbf{x}|}$$

In Fourier-space, the convolution becomes a simple multiplication!

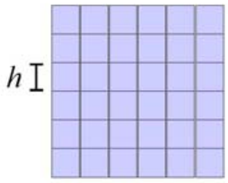
$$\hat{\Phi}(\mathbf{k}) = \hat{g}(\mathbf{k}) \cdot \hat{\rho}(\mathbf{k})$$

→ **Solve the potential in these steps:**

(1) FFT forward of the density field
(2) Multiplication with the Green's function
(3) FFT backwards to obtain potential

The four steps of the PM algorithm
(a) Density assignment
(b) Computation of the potential
(c) Determination of the force field
(d) Assignment of forces to particles

Density assignment



$\{\mathbf{x}_m\}$ set of discrete mesh centres

Give particles a "shape" S(x). Then to each mesh cell, we assign the fraction of mass that falls into this cell. The overlap for a cell is given by:

$$W(\mathbf{x_m} - \mathbf{x}_i) = \int_{\mathbf{x_m}-\frac{h}{2}}^{\mathbf{x_m}+\frac{h}{2}} S(\mathbf{x}' - \mathbf{x}_i)\,\mathrm{d}\mathbf{x}' = \int \Pi\left(\frac{\mathbf{x}' - \mathbf{x_m}}{h}\right) S(\mathbf{x}' - \mathbf{x}_i)\,\mathrm{d}\mathbf{x}'$$

The assignment function is hence the convolution:

$$W(\mathbf{x}) = \Pi\left(\frac{\mathbf{x}}{h}\right) \star S(\mathbf{x}) \quad \text{where} \quad \Pi(x) = \begin{cases} 1 & \text{for } |x| \leq \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}$$

The density on the mesh is then a sum over the contributions of each particle as given by the assignment function:

$$\rho(\mathbf{x_m}) = \frac{1}{h^3} \sum_{i=1}^{N} m_i\, W(\mathbf{x}_i - \mathbf{x_m})$$

Commenly used particle shape functions and assignment schemes

Name	Shape function S(x)	# of cells involved	Properties of force
NGP Nearest grid point	$\delta(\mathbf{x})$	$1^3 = 1$	piecewise constant in cells
CIC Clouds in cells	 $\frac{1}{h^3}\Pi\left(\frac{\mathbf{x}}{h}\right) \star \delta(\mathbf{x})$	$2^3 = 8$	piecewise linear, continuous
TSC Triangular shaped clouds	 $\frac{1}{h^3}\Pi\left(\frac{\mathbf{x}}{h}\right) \star \frac{1}{h^3}\Pi\left(\frac{\mathbf{x}}{h}\right)$	$3^3 = 27$	continuous first derivative

Note: For interpolation of the grid to obtain the forces, the same assignment function needs to be used to ensure momentum conservation. (In the CIC case, this is identical to tri-linear interpolation.)

Advantages and disadvantages of the PM-scheme

Pros: **SPEED** and simplicity

Cons:

- Spatial force resolution limited to mesh size. → *serious problem:*
- Force errors somewhat anisotropic on the scale of the cell size

cosmological simulations cluster strongly and have a very large dynamic range

cannot make the PM-mesh fine enough and resolve internal structure of halos as well as large cosmological scales

→ we need a method to increase the **dynamic range** available in the force calculation

P3M:

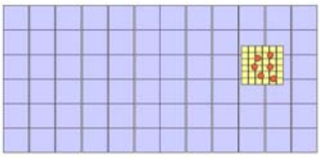
Particle-Particle Particle-Mesh

Particle-Particle PM schemes (P^3M)

Idea: Supplement the PM force with a direct summation short-range force at the scale of the mesh cells. The particles in cells are linked together by a chaining list.

Offers much higher dynamic range, but becomes slow when clustering sets in.

In AP^3M, mesh-refinements are placed on clustered regions



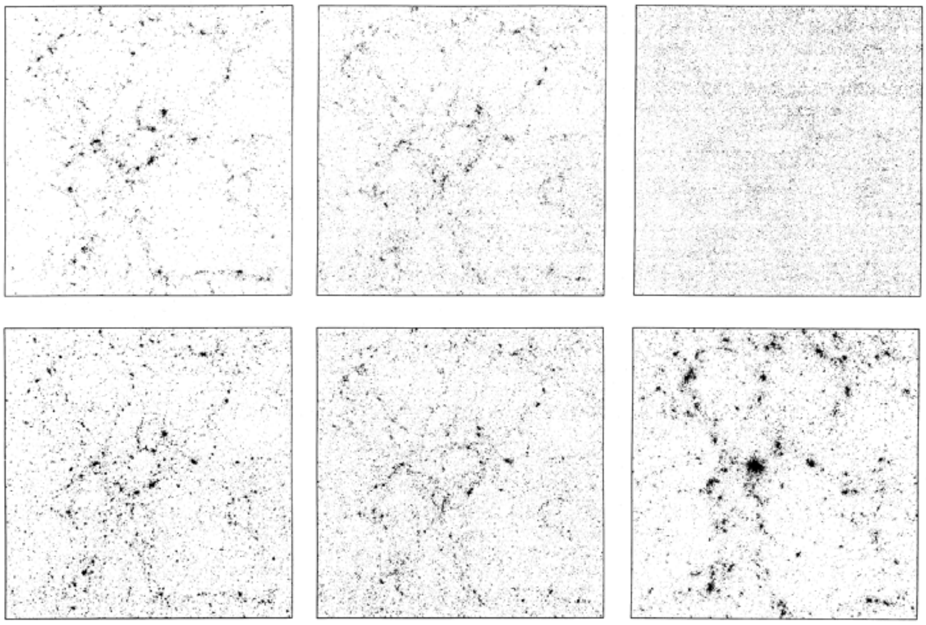
Can avoid clustering slow-down, but has higher complexity and ambiguities in mesh placement

Codes that use AP^3M: **HYDRA** (Couchman)

DEFW:

Davis –Efstathiou-Frenk-White

‘the gang of four’



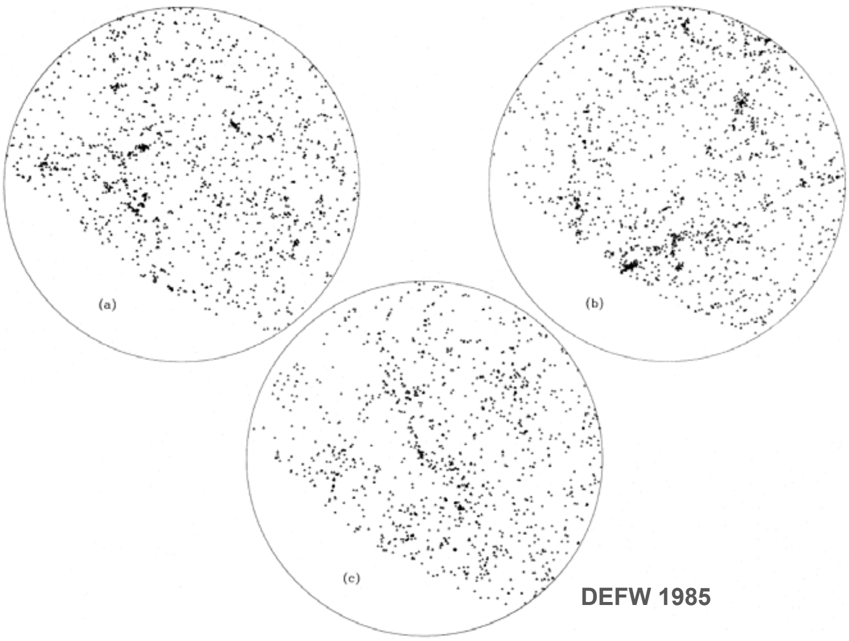


FIG. 12.—Redshift catalogs constructed from two open models (O2 and O3) are shown in (*a*) and (*b*) as projections onto the "sky." Particles were selected for inclusion in these catalogs in such a way as to mimic the northern CfA survey. The real data are shown in the same format in (*c*). These are equal area plots of the sky; the outer circle corresponds to Galactic latitude +40°, while the empty regions correspond to declinations below 0°. In constructing the catalog from O3 shown in (*b*), the "observer" was purposely sited near a prominent cluster.

DEFW 1985

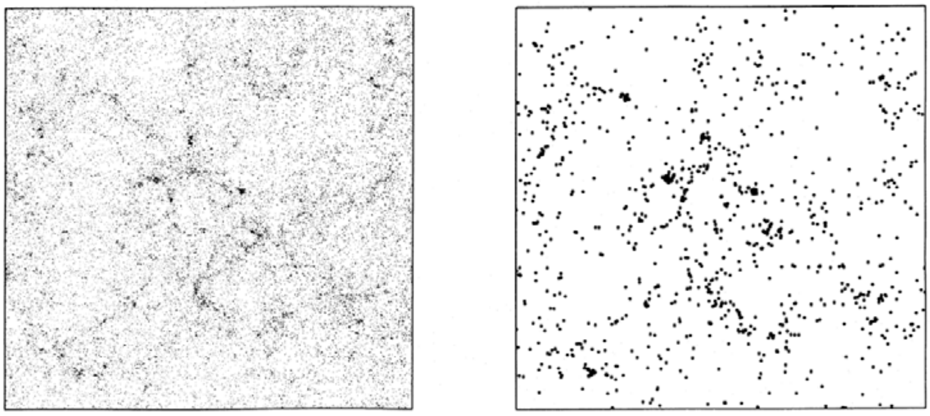


FIG. 16.—The projected distribution of all particles (*left*) and of the "galaxies" (*right*) in EdS1 at $a = 1.4$. The side of the box is $32.5h^{-1}$ Mpc. "Galaxies" are assumed to form only at the $2.5\,\sigma$ peaks of the linear density distribution.

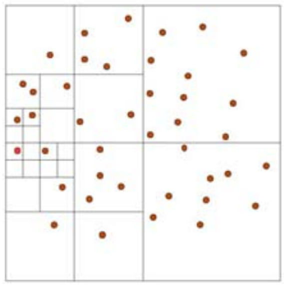
Tree Algorithms

Gravity is the driving force for structure formation in the universe

HIERARCHICAL TREE ALGORITHMS

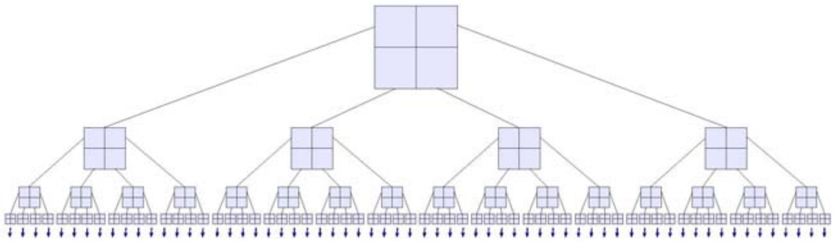
The N^2 - scaling of direct summation puts serious limitations on N...

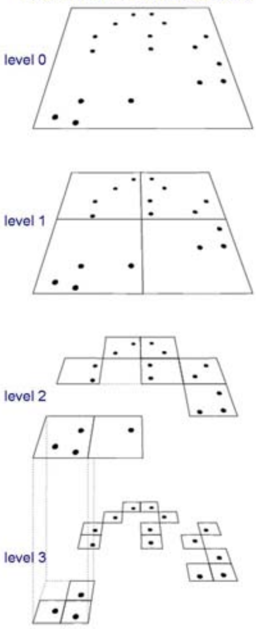
But we want N ~ 10^6-10^{10} for collisionless dynamics of dark matter !



Idea: Group distant particles together, and use their multipole expansion.

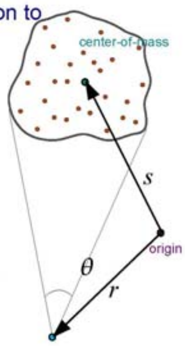
→ Only ~ log(N) force terms per particle.





Tree algorithms

Idea: Use hierarchical multipole expansion to account for distant particle groups



$$\Phi(\mathbf{r}) = -G \sum_i \frac{m_i}{|\mathbf{r} - \mathbf{x}_i|}$$

We expand:

$$\frac{1}{|\mathbf{r} - \mathbf{x}_i|} = \frac{1}{|(\mathbf{r} - \mathbf{s}) - (\mathbf{x}_i - \mathbf{s})|}$$

for $\quad |\mathbf{x}_i - \mathbf{s}| \ll |\mathbf{r} - \mathbf{s}| \qquad \mathbf{y} \equiv \mathbf{r} - \mathbf{s}$

and obtain:

$$\frac{1}{|\mathbf{y} + \mathbf{s} - \mathbf{x}_i|} = \frac{1}{|\mathbf{y}|} - \frac{\mathbf{y} \cdot (\mathbf{s} - \mathbf{x}_i)}{|\mathbf{y}|^3} + \frac{1}{2} \frac{\mathbf{y}^T \left[3(\mathbf{s} - \mathbf{x}_i)(\mathbf{s} - \mathbf{x}_i)^T - \mathbf{I}(\mathbf{s} - \mathbf{x}_i)^2\right] \mathbf{y}}{|\mathbf{y}|^5} + \ldots$$

the dipole term vanishes when summed over all particles in the group

The multipole moments are computed for each node of the tree

Monpole moment:

$$M = \sum_i m_i$$

Quadrupole tensor:

$$Q_{ij} = \sum_k m_k \left[3(\mathbf{x}_k - \mathbf{s})_i(\mathbf{x}_k - \mathbf{s})_j - \delta_{ij}(\mathbf{x}_k - \mathbf{s})^2\right]$$

Resulting potential/force approximation:

$$\Phi(\mathbf{r}) = -G\left[\frac{M}{|\mathbf{y}|} + \frac{1}{2}\frac{\mathbf{y}^T\mathbf{Q}\,\mathbf{y}}{|\mathbf{y}|^5}\right]$$

For a single force evaluation, not *N* single-particle forces need to be computed, but **only of order *log(N)* multipoles**, depending on opening angle.

- The tree algorithm has no intrinsic restrictions for its dynamic range
- force accuracy can be conveniently adjusted to desired level
- the speed does depend only very weakly on clustering state
- geometrically flexible, allowing arbitrary geometries

TreePM

mixing Treecode with PM

Particularly at high redshift, it is expensive to obtain accurate forces with the tree-algorithm

THE TREE-PM FORCE SPLIT

Periodic peculiar potential

$$\nabla^2\phi(\mathbf{x}) = 4\pi G[\rho(\mathbf{x}) - \overline{\rho}] = 4\pi G \sum_{\mathbf{n}} \sum_i m_i \left[\tilde{\delta}(\mathbf{x} - \mathbf{x}_i - \mathbf{n}L) - \frac{1}{L^3}\right]$$

Idea: Split the potential (of a single particle) in Fourier space into a long-range and a short-range part, and compute them separately with PM and TREE algorithms, respectively.

Poisson equation in Fourier space:

$$\phi_{\mathbf{k}} = -\frac{4\pi G}{\mathbf{k}^2}\rho_{\mathbf{k}} \quad (\mathbf{k} \neq 0)$$

$$\phi_{\mathbf{k}}^{\mathrm{long}} = \phi_{\mathbf{k}} \exp(-\mathbf{k}^2 r_s^2)$$

Solve with PM-method
- CIC mass assignment
- FFT
- multiply with kernel
- FFT backwards
- Compute force with 4-point finite difference operator
- Interpolate forces to particle positions

$$\phi_{\mathbf{k}}^{\mathrm{short}} = \phi_{\mathbf{k}} \left[1 - \exp(-\mathbf{k}^2 r_s^2)\right]$$

FFT to real space $\phi(r) = -\frac{Gm}{r}\,\mathrm{erfc}\left(\frac{r}{2r_s}\right)$

Solve in real space with TREE

The maximum size of a TreePM simulation with *Lean*-GADGET-II is essentially memory bound

A HIGHLY MEMORY EFFICIENT VERSION OF GADGET-II

Particle Data	Tree storage	FFT workspace
44 bytes / particle	40 bytes / particle	24 bytes / mesh-cell

Not needed concurently!

Special code version *Lean*-GADGET-II needs: **84 bytes / particle** (Assuming 1.5 mesh-cells/particle)

Simulation Set-up:

- Particle number: 2160^3 = 10.077.696.000 = ~ 10^{10} particles
- Boxsize: L = 500 h^{-1} Mpc
- Particle mass: $m_p = 8.6 \times 10^8\ h^{-1}\ M_\odot$
- Spatial resolution: 5 h^{-1} kpc
- Size of FFT: 2560^3 = 16.777.216.000 = ~ 17 billion cells

Minimum memory requirement of simulation code **~840 GByte**

Compared to Hubble-Volume simulation:
- > 2000 times better mass resolution
- 10 times larger particle number
- 13 better spatial resolution

Moving Mesh:

Arepo & TESS

A finite volume discretization of the Euler equations on a moving mesh can be readily defined

THE EULER EQUATIONS AS HYPERBOLIC SYSTEM OF CONSERVATION LAWS

Euler equations

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F} = 0$$

State vector

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho \mathbf{v} \\ \rho e \end{pmatrix}$$

Flux vector

$$\mathbf{F}(\mathbf{U}) = \begin{pmatrix} \rho \mathbf{v} \\ \rho \mathbf{v}\mathbf{v}^T + P \\ (\rho e + P)\mathbf{v} \end{pmatrix}$$

$$e = u + \mathbf{v}^2/2$$

Equation of state: $P = (\gamma - 1)\rho u$

Discretization in terms of a number of finite volume cells:

Cell averages

$$\mathbf{Q}_i = \begin{pmatrix} M_i \\ \mathbf{p}_i \\ E_i \end{pmatrix} = \int_{V_i} \mathbf{U}\, \mathrm{d}V$$

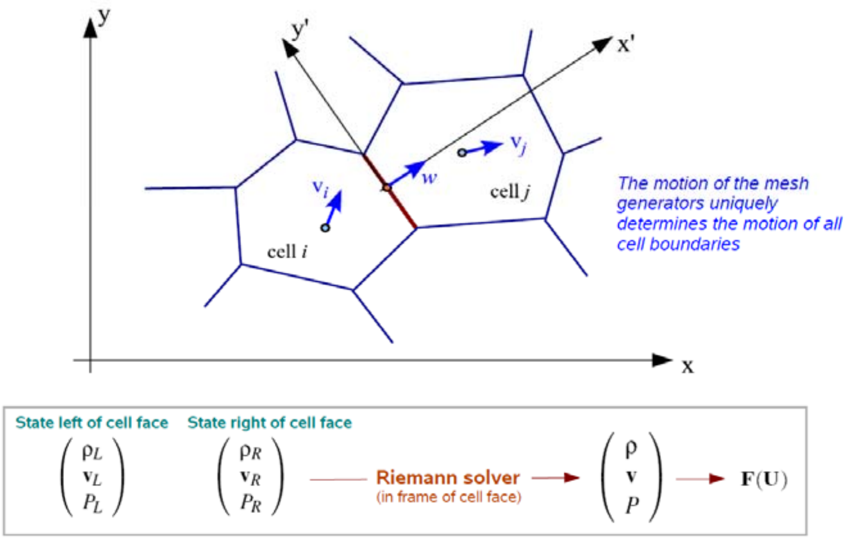
Evolution equation

$$\frac{\mathrm{d}\mathbf{Q}_i}{\mathrm{d}t} = -\int_{\partial V_i} \left[\mathbf{F}(\mathbf{U}) - \mathbf{U}\mathbf{w}^T\right] \mathrm{d}\mathbf{n}$$

But how to compute the fluxes through cell surfaces?

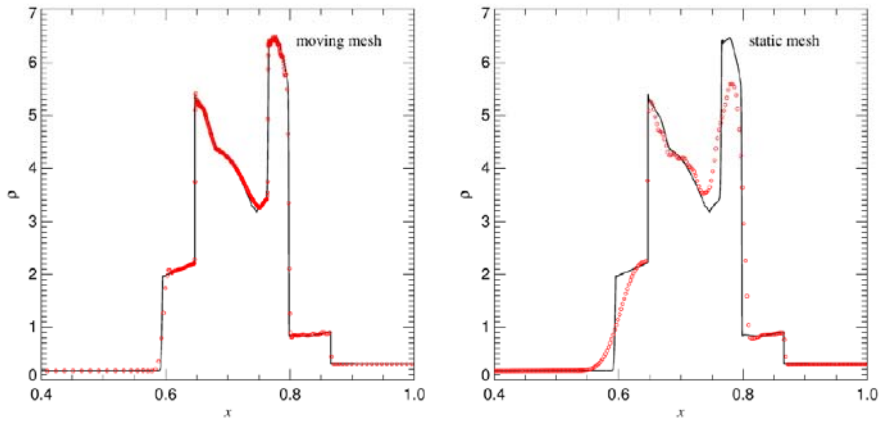
The fluxes are calculated with an exact Riemann solver in the frame of the moving cell boundary

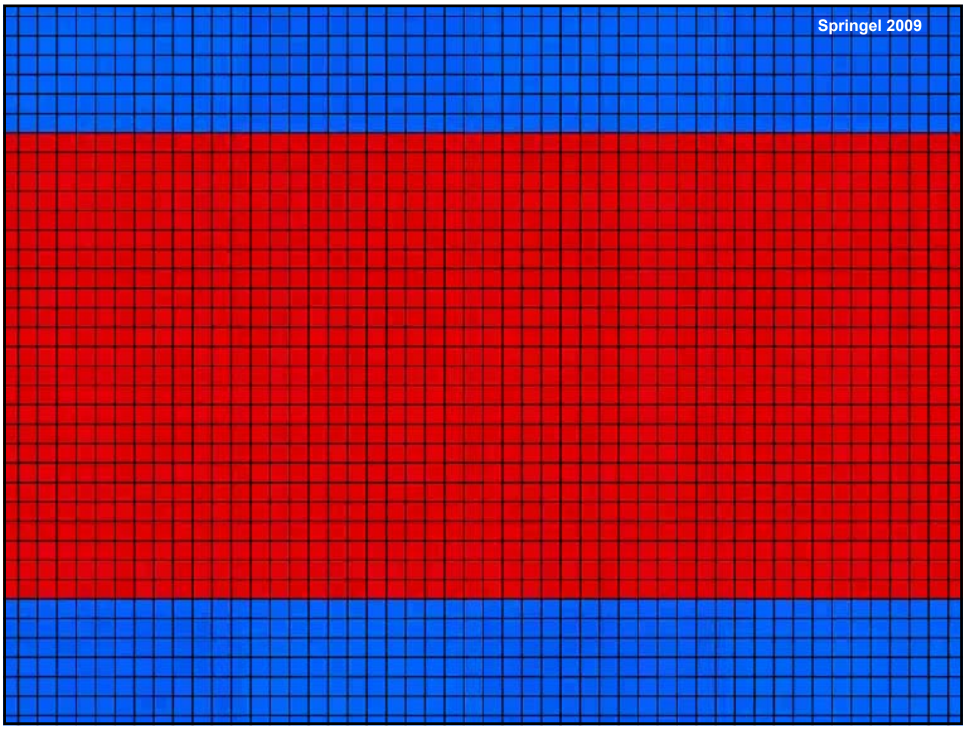
SKETCH OF THE FLUX CALCULATION



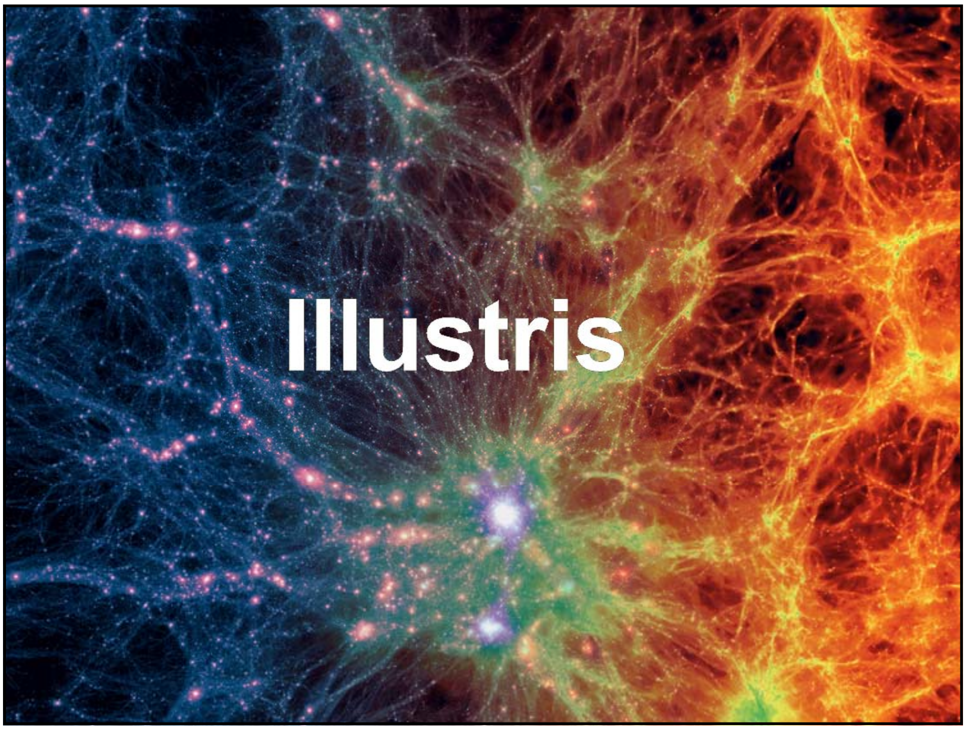
The moving-mesh code deals well will problems that involve complicated shock interactions

WOODWARD & COLELLA'S INTERACTING DOUBLE BLAST PROBLEM




Springel 2009


Springel 2009

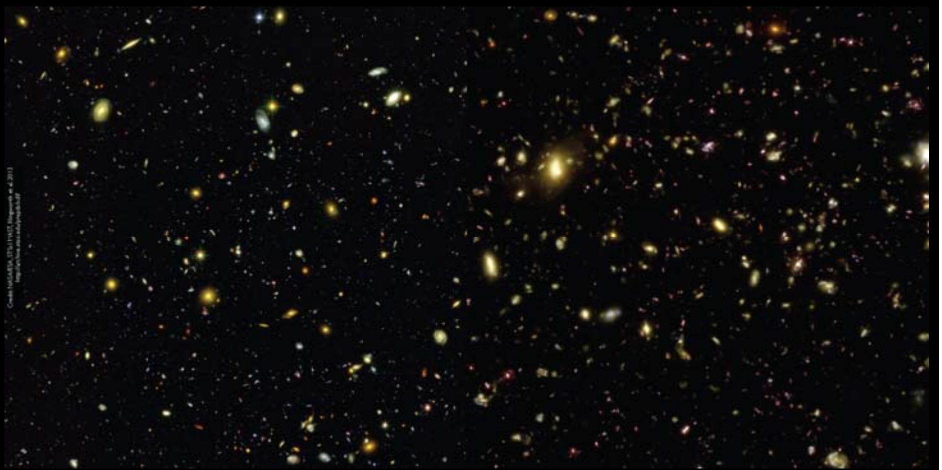


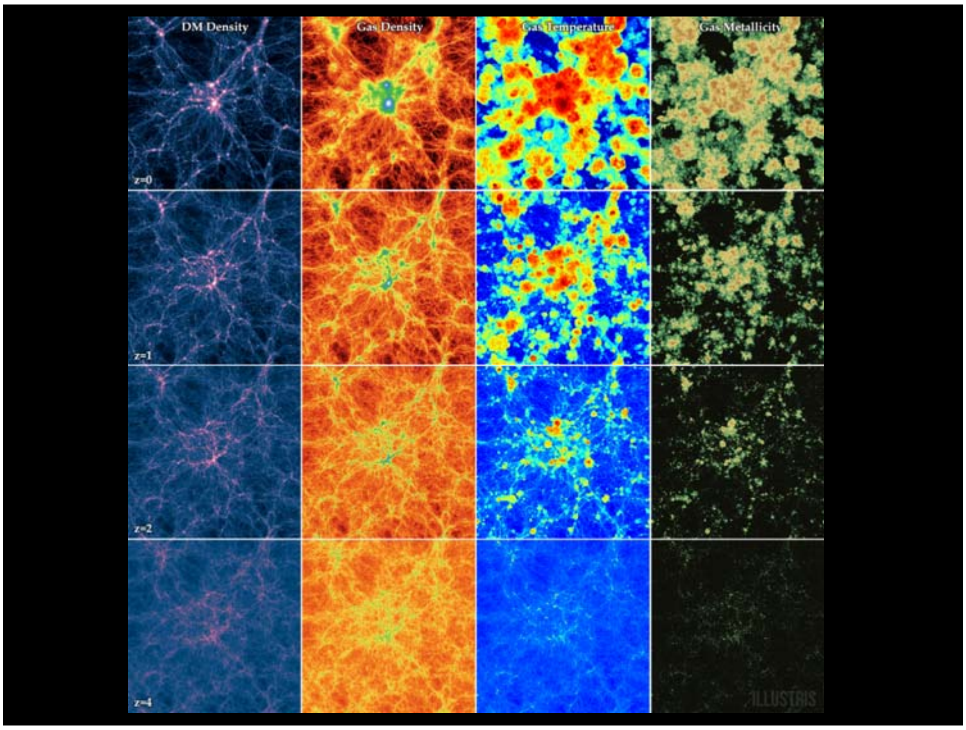
Illustris Simulation

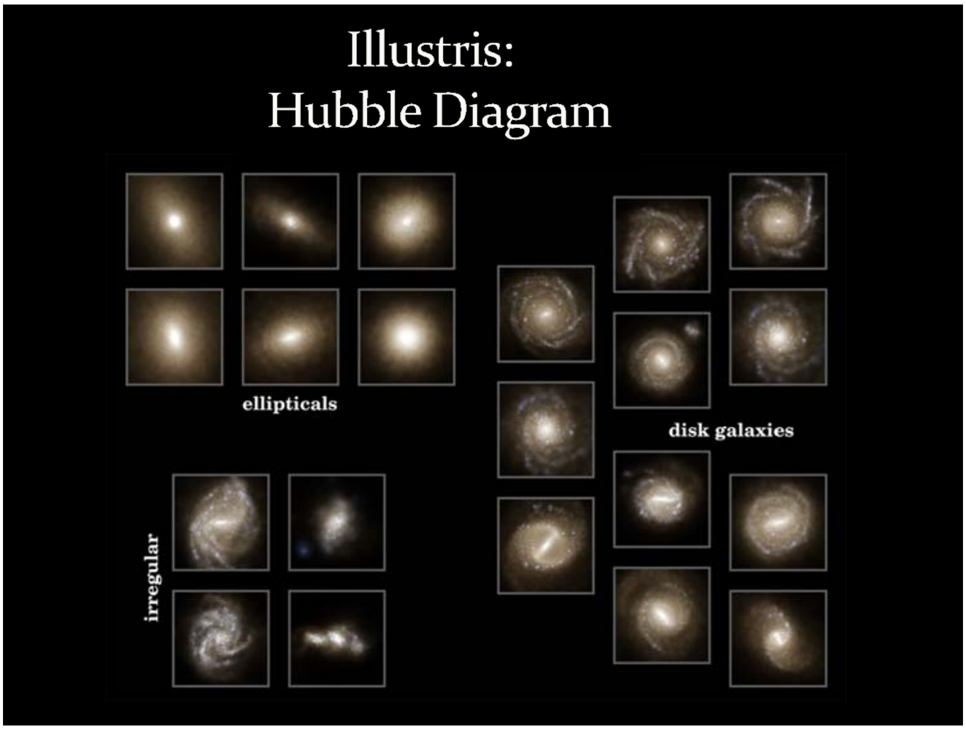
- Run: 2013-2014
- First multi – 100 billion particle simulation

- Arepo adaptive-grid (Voronoi) Nbody+hydro simulations
- LCDM cosmology
- 6720^3 particles ~ 300×10^{10} particles
- Run on various supercomputers (France, Germany, USA), 8192 cores, 19 million CPU hours, 25 Terabyte RAM
- Data: 230 Terabyte

Illustris:
HUDF, real vs. simulated




DM Density
Gas Density
Gas Temperature
Gas Metallicity
z=0
z=1
z=2
z=4
ILLUSTRIS


Illustris:
Hubble Diagram
ellipticals
disk galaxies
irregular

Phase-Space Dynamics

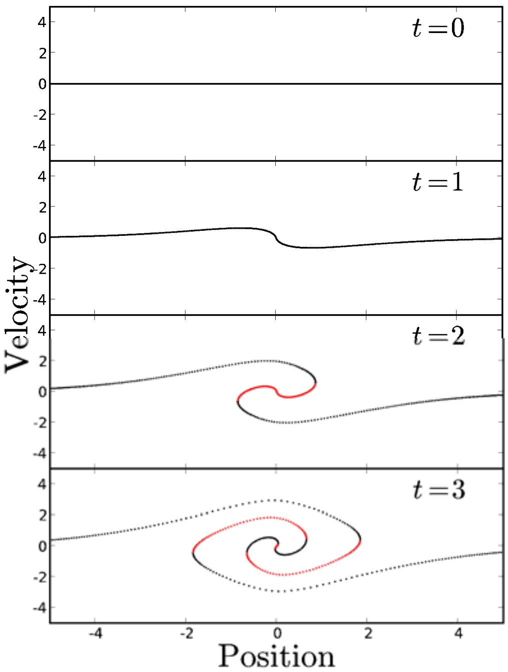
Phase Space Evolution

Dark Matter Phase Space sheet:

3-D structure projection of a folding DM phase space sheet In 6-D phase space

- Shandarin 2010, 2011
- Neyrinck et al. 2011, 2012 Origami
- Abel et al. 2011

Evolving matter distribution in position-velocity space – 1D



Phase Space Evolution

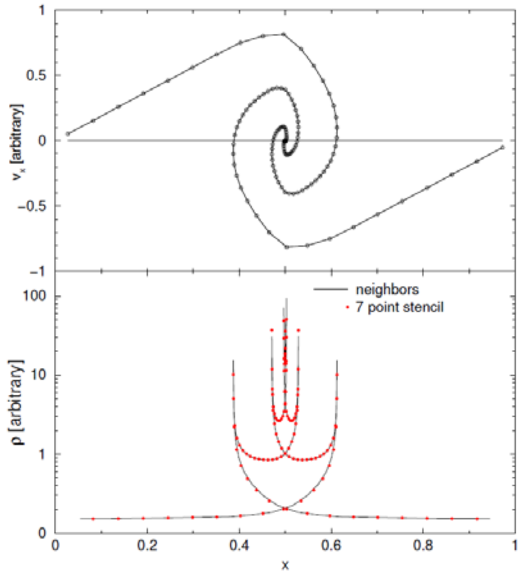
Phase space:

Velocity vs. Position

⬇

Density:

$$\rho(\vec{x},t) = \int f(\vec{x},\vec{v},t)\, d\vec{v}$$



Lagrangian-Eulerian Phase Space

To follow evolving phase-space of cosmic structure, it is sometimes insightful to consider a coordinate transformation of 6D phase-space:

Eulerian coordinates $\vec{x}$ and Eulerian coordinates $\vec{q}$ of a mass element:

$$f(\vec{x},\vec{q})$$

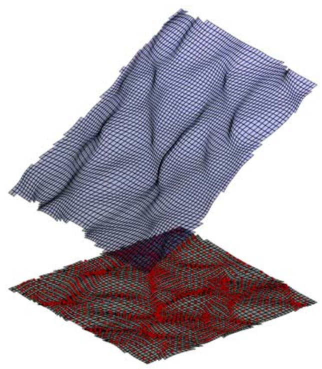
Note that in Zeldovich approximation, the velocity of a mass element is:

$$\vec{v}(\vec{q},t) = -a(t)D(t)f(\Omega)\ \vec{\nabla}\Phi(\vec{q})$$

Tessellation Deformation & Phase Space Projection

Translation towards Multi-D space:

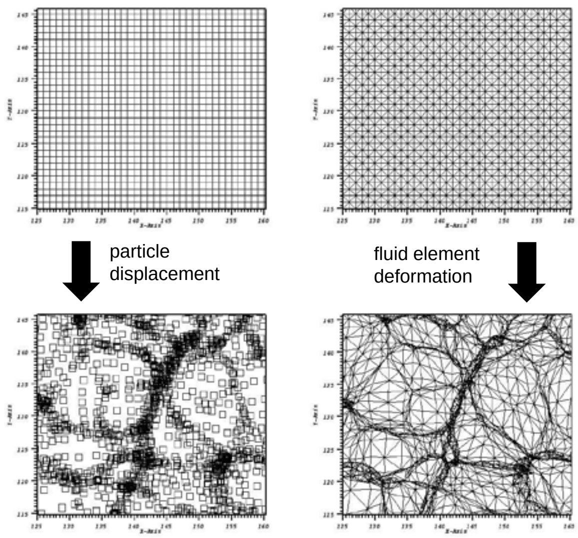
- Look at deformation of initial tessellation
- each tessellation cell represents matter cell
- evolution deforms cell
- once cells start to overlap, manifestation of different phase-space matter streams



Tessellation Deformation & Phase Space Projection

Translation towards Multi-D space:

- Look at deformation of initial tessellation
- each tessellation cell represents matter cell
- evolution deforms cell
- once cells start to overlap, manifestation of different phase-space matter streams

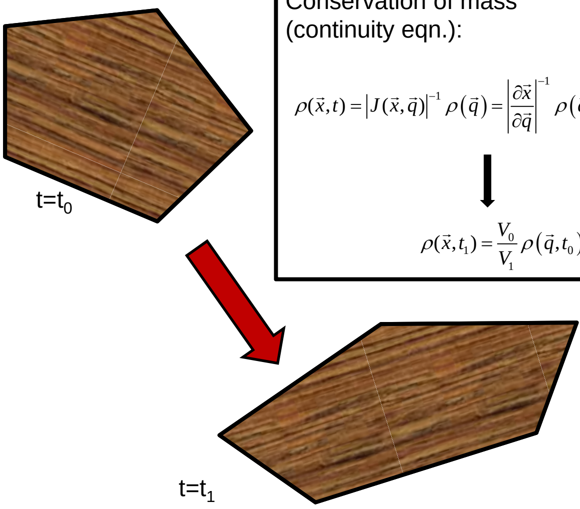


Tessellation Deformation & Phase Space Projection

Translation towards Multi-D space:

- Look at deformation of initial tessellation
- each tessellation cell represents matter cell
- evolution deforms cell

Monostream Density Evolution



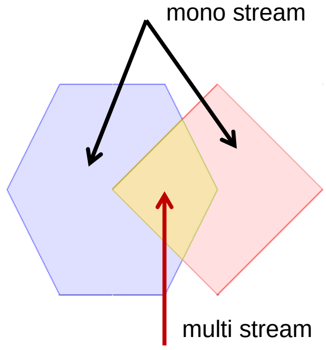
Conservation of mass (continuity eqn.):

$$\rho(\vec{x},t) = \left|J(\vec{x},\vec{q})\right|^{-1} \rho(\vec{q}) = \left|\frac{\partial \vec{x}}{\partial \vec{q}}\right|^{-1} \rho(\vec{q})$$

$$\downarrow$$

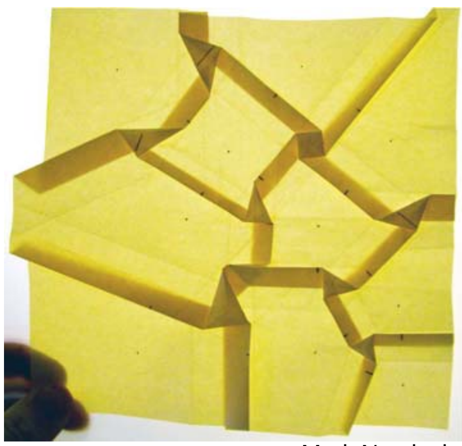
$$\rho(\vec{x},t_1) = \frac{V_0}{V_1} \rho(\vec{q},t_0)$$

(Cosmic) ORIGAMI



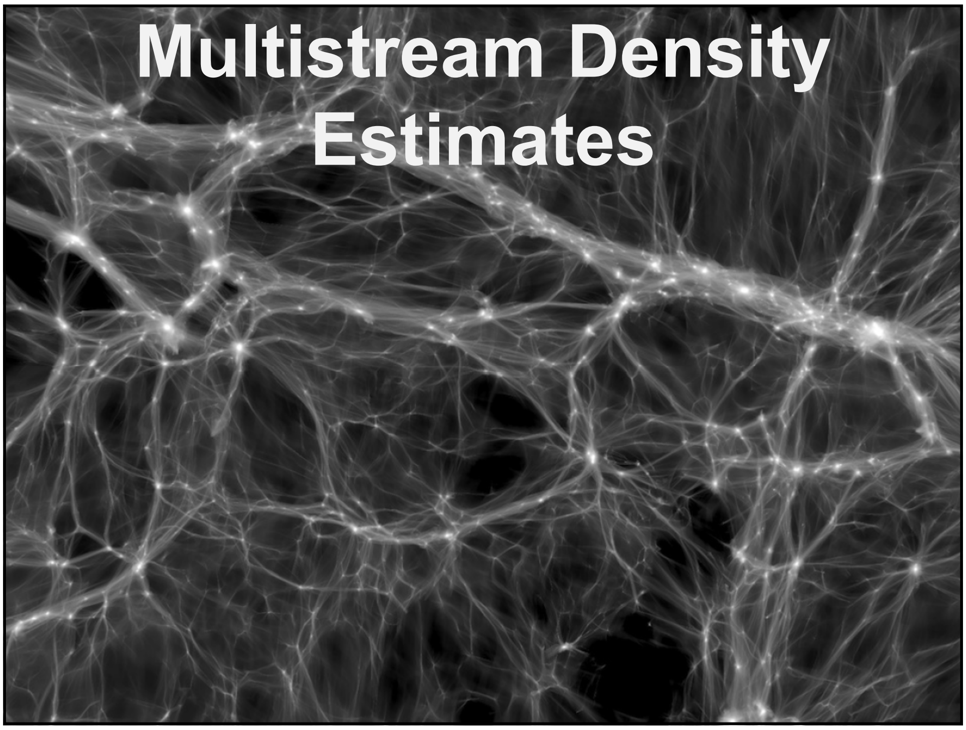
$$\rho_{total}(\vec{x},t_1) = \sum_i \frac{V_{0i}}{V_{1i}} \rho(\vec{q}_i,t_0)$$

Evolution of dynamical system:
Phase-space folding – Cosmic Origami

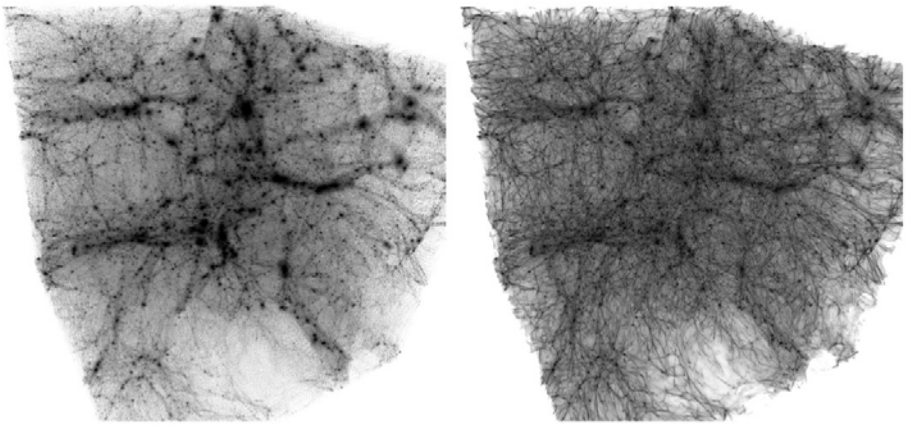


Mark Neyrinck

Multistream Density Estimates



Multistream Density Estimates



Cosmic Web Stream Density

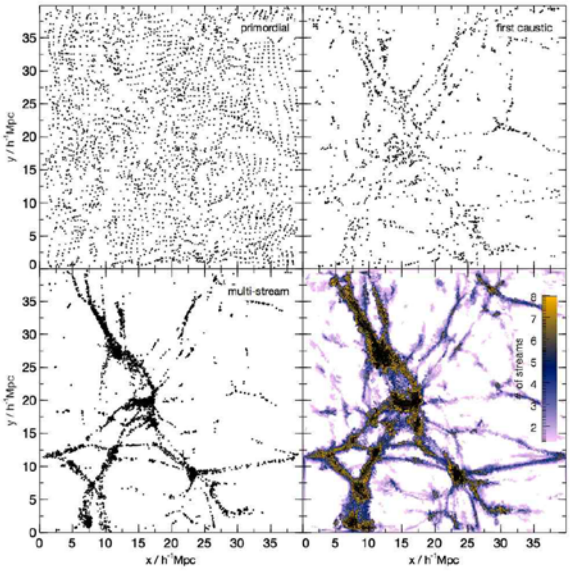
Translation towards Multi-D space:

Density of
dark matter streams:

- # phase space folds

=

locally overlapping tessellation cells



SPH

Smooth Particle Hydrodynamics

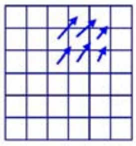
What is smoothed particle hydrodynamics?

DIFFERENT METHODS TO DISCRETIZE A FLUID

Eulerian

discretize space

representation on a mesh
(volume elements)



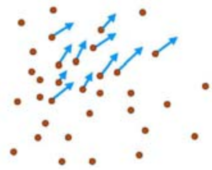
principle advantage:

high accuracy (shock capturing), low numerical viscosity

Lagrangian

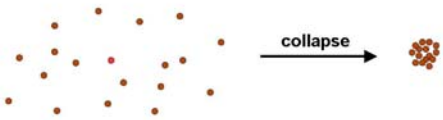
discretize mass

representation by fluid elements
(particles)



principle advantage:

resolutions adjusts automatically to the flow



The baryons in the universe can be modelled as an *ideal* gas

BASIC HYDRODYNAMICAL EQUATIONS

Euler equation:

$$\frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t} = -\frac{\nabla P}{\rho} - \nabla\Phi$$

Continuity equation:

$$\frac{\mathrm{d}\rho}{\mathrm{d}t} + \rho\nabla\cdot\mathbf{v} = 0$$

First law of thermodynamics:

$$\frac{\mathrm{d}u}{\mathrm{d}t} = -\frac{P}{\rho}\nabla\cdot\mathbf{v} - \frac{\Lambda(u,\rho)}{\rho}$$

Equation of state of ideal monoatomic gas:

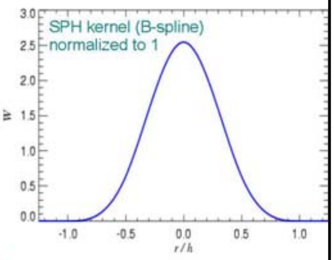
$$P = (\gamma - 1)\rho u\,, \qquad \gamma = 5/3$$

Kernel interpolation is used in smoothed particle hydrodynamics (SPH) to build continous fluid quantities from discrete tracer particles

DENSITY ESTIMATION IN SPH BY MEANS OF ADAPTIVE KERNEL ESTIMATION

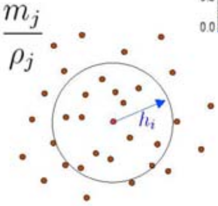
Kernel interpolant of an arbitrary function:

$$\langle A(\mathbf{r})\rangle = \int W(\mathbf{r}-\mathbf{r}', h)\, A(\mathbf{r}')\, \mathrm{d}^3 r'$$



If the function is only known at a set of discrete points, we approximate the integral as a sum, using the replacement:

$$\mathrm{d}^3 r' \mapsto \frac{m_j}{\rho_j}$$



$$\langle A_i\rangle = \sum_{j=1}^{N} \frac{m_j}{\rho_j} A_j\, W(\mathbf{r}_{ij}; h_i)$$

This leads to the SPH density estimate, for $A_i = \rho_i$

$$\rho_i = \sum_{j=1}^{N} m_j W(|\mathbf{r}_{ij}|, h_i)$$

→ **This can be differentiated !**

Kernel interpolants allow the construction of derivatives from a set of discrete tracer points

EXAMPLES FOR ESTIMATING THE VELOCITY DIVERGENCE

Smoothed estimate for the velocity field:

$$\langle \mathbf{v}_i\rangle = \sum_j \frac{m_j}{\rho_j} \mathbf{v}_j\, W(\mathbf{r}_i - \mathbf{r}_j)$$

Velocity divergence can now be readily estimated:

$$\nabla\cdot\mathbf{v} = \nabla\cdot\langle\mathbf{v}_i\rangle = \sum_j \frac{m_j}{\rho_j} \mathbf{v}_j\, \nabla_i W(\mathbf{r}_i - \mathbf{r}_j)$$

But alternative (and better) estimates are possible also:

Invoking the identity

$$\rho\nabla\cdot\mathbf{v} = \nabla\cdot(\rho\mathbf{v}) - \mathbf{v}\cdot\nabla\rho$$

one gets a "pair-wise" formula:

$$\rho_i(\nabla\cdot\mathbf{v})_i = \sum_j m_j(\mathbf{v}_j - \mathbf{v}_i)\, \nabla_i W(\mathbf{r}_i - \mathbf{r}_j)$$

What is smoothed particle hydrodynamics?

BASIC EQUATIONS OF SMOOTHED PARTICLE HYDRODYNAMICS

Each particle carries either the energy or the entropy per unit mass as independent variable

Density estimate $\rho_i = \sum_{j=1}^{N} m_j W(|\mathbf{r}_{ij}|, h_i)$ → **Continuity equation automatically fulfilled.**

→ $P_i = (\gamma - 1)\rho_i u_i$

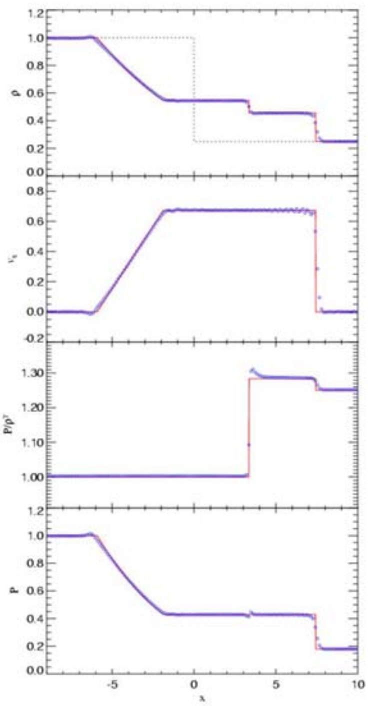
Euler equation

$$\frac{\mathrm{d}\mathbf{v}_i}{\mathrm{d}t} = -\sum_{j=1}^{N} m_j \left(\frac{P_i}{\rho_i^2} + \frac{P_j}{\rho_j^2} + \Pi_{ij} \right) \nabla_i \overline{W}_{ij}$$

$+\Pi_{ij}$: Artificial viscosity

First law of thermodynamics

$$\frac{\mathrm{d}u_i}{\mathrm{d}t} = \frac{1}{2}\sum_{j=1}^{N} m_j \left(\frac{P_i}{\rho_i^2} + \frac{P_j}{\rho_j^2} + \Pi_{ij} \right) \mathbf{v}_{ij} \cdot \nabla_i \overline{W}_{ij}$$



An artificial viscosity needs to be introduced to capture shocks

SHOCK TUBE PROBLEM AND VISCOSITY

viscous force:

$$\left.\frac{\mathrm{d}\mathbf{v}_i}{\mathrm{d}t}\right|_{\mathrm{visc}} = -\sum_{j=1}^{N} m_j \Pi_{ij} \nabla_i \overline{W}_{ij}$$

parameterization of the artificial viscosity:

$$\Pi_{ij} = \begin{cases} -\frac{\alpha}{2}\frac{[c_i + c_j - 3w_{ij}]w_{ij}}{\rho_{ij}} & \text{if } \mathbf{v}_{ij}\cdot\mathbf{r}_{ij} < 0 \\ 0 & \text{otherwise} \end{cases}$$

$$v_{ij}^{\mathrm{sig}} = c_i + c_j - 3w_{ij},$$

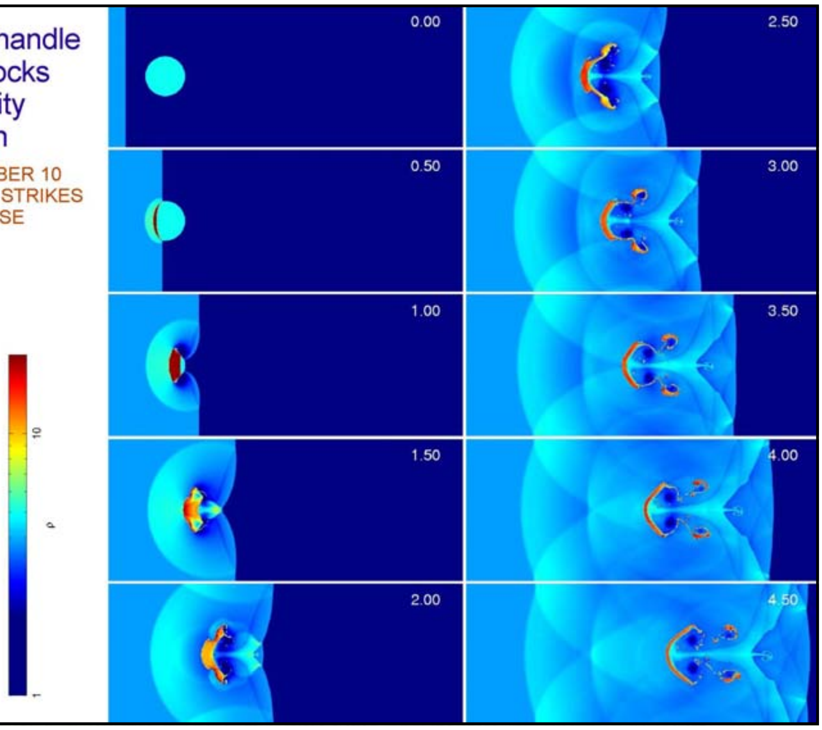
$$w_{ij} = \mathbf{v}_{ij}\cdot\mathbf{r}_{ij}/|\mathbf{r}_{ij}|$$

heat production rate:

$$\frac{\mathrm{d}u_i}{\mathrm{d}t} = \frac{1}{2}\sum_{j=1}^{N} m_j \Pi_{ij} \mathbf{v}_{ij} \cdot \nabla_i \overline{W}_{ij}$$

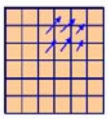
SPH can handle strong shocks and vorticity generation

A MACH NUMBER 10 SHOCK THAT STRIKES AN OVERDENSE CLOUD



There are principal differences between SPH and Eulerian schemes

SOME FUNDAMENTAL DIFFERENCE BETWEENS SPH AND MESH-HYDRODYNAMICS



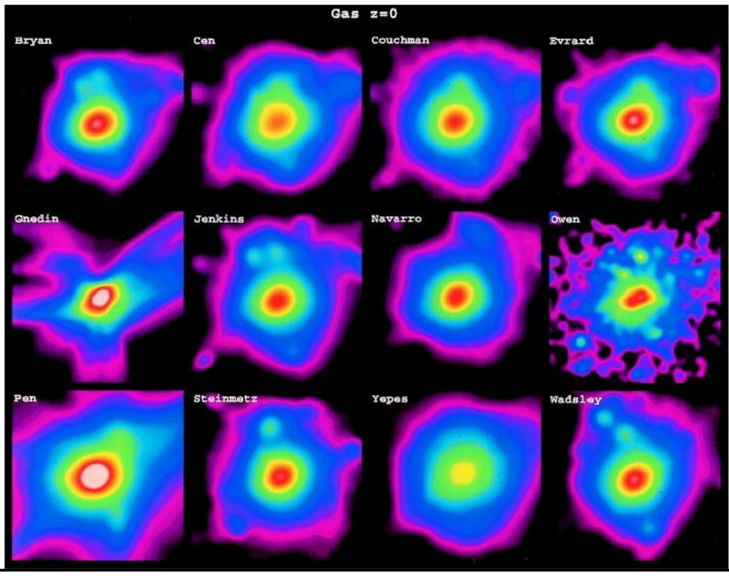
Eulerian	Lagrangian
sharp shocks, somewhat less sharp contact discontinuities (best schemes resolve fluid discontinuities it in one cell)	**shocks broadened over roughly 2-3 smoothing lengths** (post-shock properties are correct though)
mixing happens implicitly at the cell level (but advection adds numerical diffusivity and may provide a source of spurious entropy)	**mixing entirely suppressed at the particle-level** (no spurious entropy production, but fluid instabilities may be suppressed)
no need for artificial viscosity (in Godunov schemes)	**requires artificial viscosity**
Truncation error not Galilean invariant *("high Mach number problem")*	**Galilean invariant**
self-gravity of the gas done on a mesh (but dark matter must still be represented by particles) **no explicit conservation of total energy when self-gravity is included**	**self-gravity of the gas naturally treated with the same accuracy as the dark matter, total energy conserved**

Springel 2009

Different hydrodynamical simulation codes are broadly in agreement, albeit with substantial scatter and differences in detail

THE SANTA BARBARA CLUSTER COMPARISON PROJECT

Frenk, White & 23 co-authors (1999)

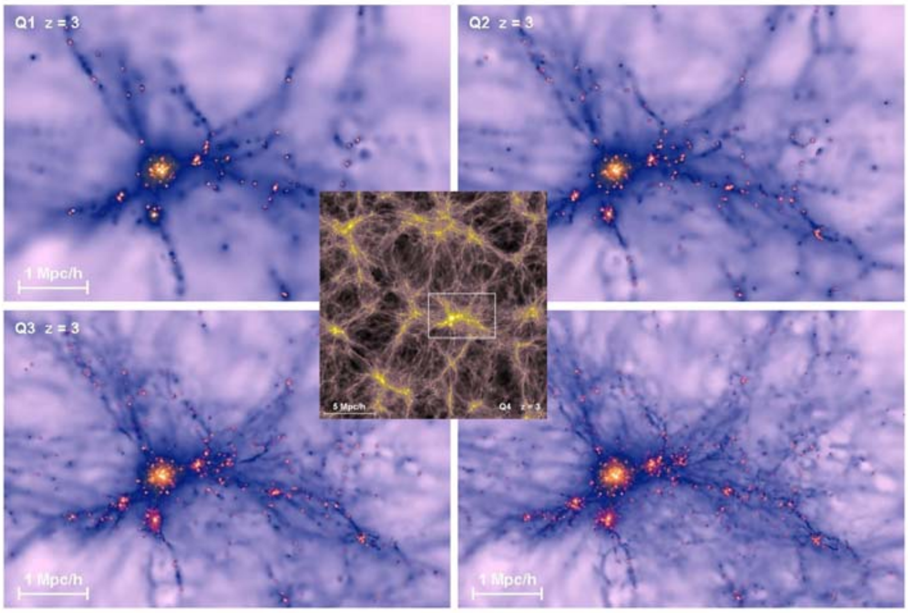


Nbody Simulations

a select number of results

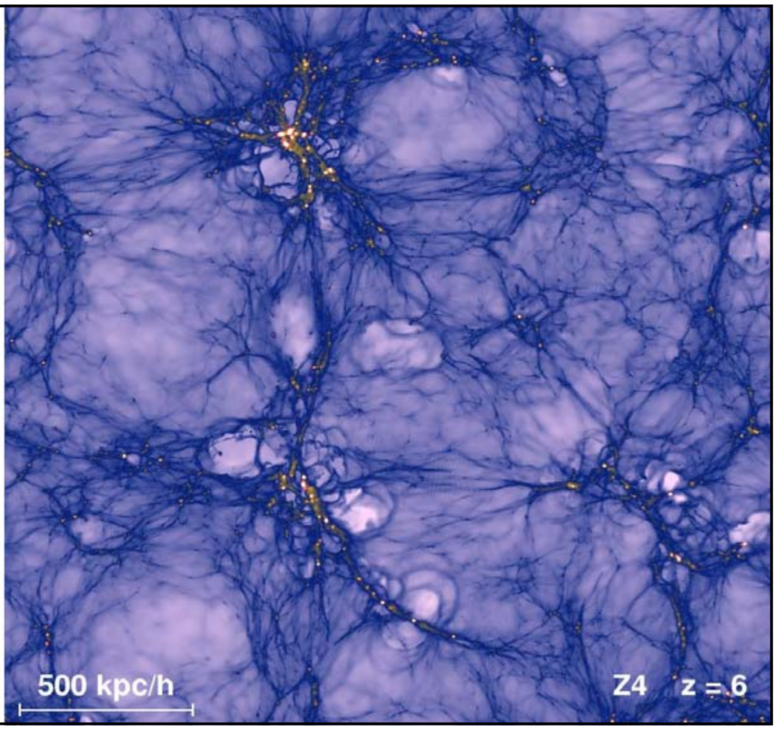
largely based on excellent Potsdam lectures on Nbody simulations (2006)
by V. Springel

Higher mass resolution can resolve smaller galaxies



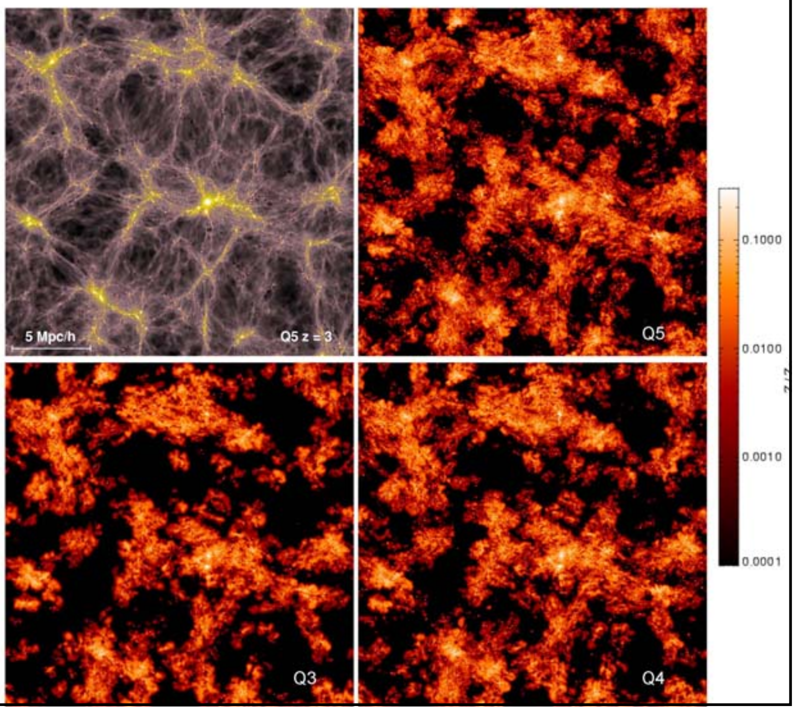
The "large-scale" structure seen at high redshift superficially resembles the morphology of structure seen at low redshift

GAS DISTRIBUTION SEEN IN A SMALL PERIODIC BOX AT REDSHIFT z=6



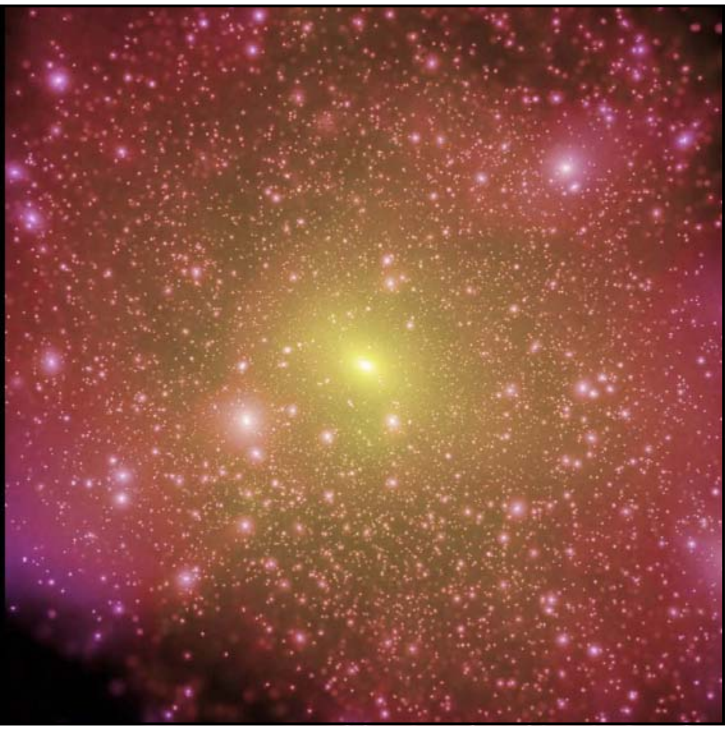
Projected metallcity maps reveal a highly non-uniform enrichment pattern

PROJECTED MEAN GAS METALLICITY



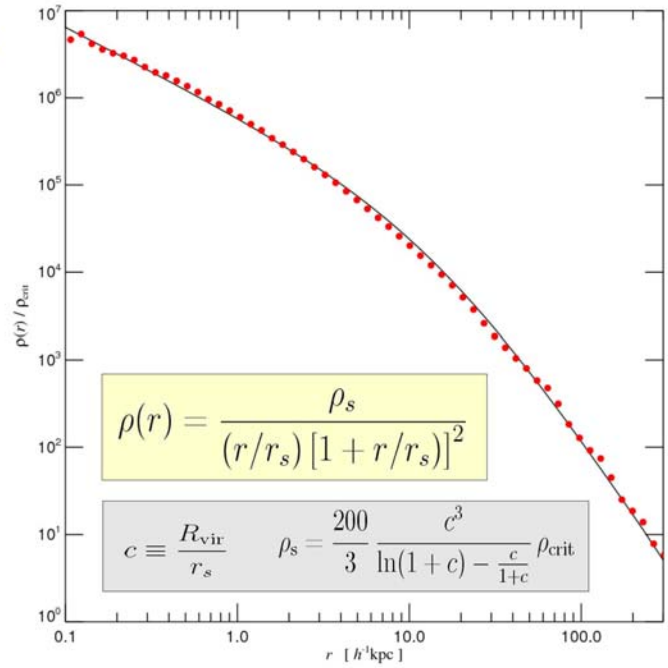
Simulated dark matter halos are not spherically symmetric, nor does their structure look as simple as assumed in the analytic models models

DARK MATTER DISTRIBUTION IN A HIGH-RESOLUTION "MILKY WAY" HALO



N-body simulations find a universal profile that is not a power-law

THE NFW-PROFILE

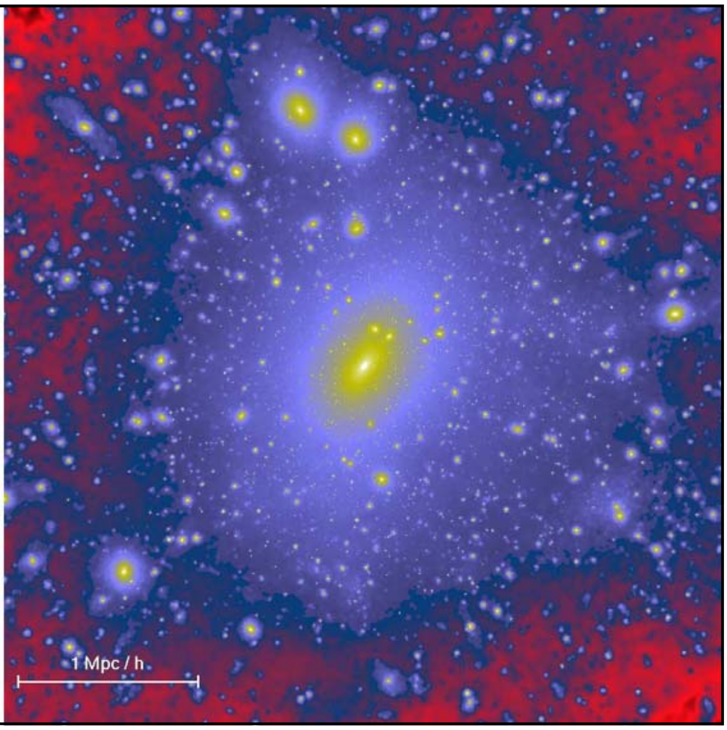


Halos formed in high-resolution simulations of cold dark matter show rich substructure

SUBHALOS IN A RICH CLUSTER

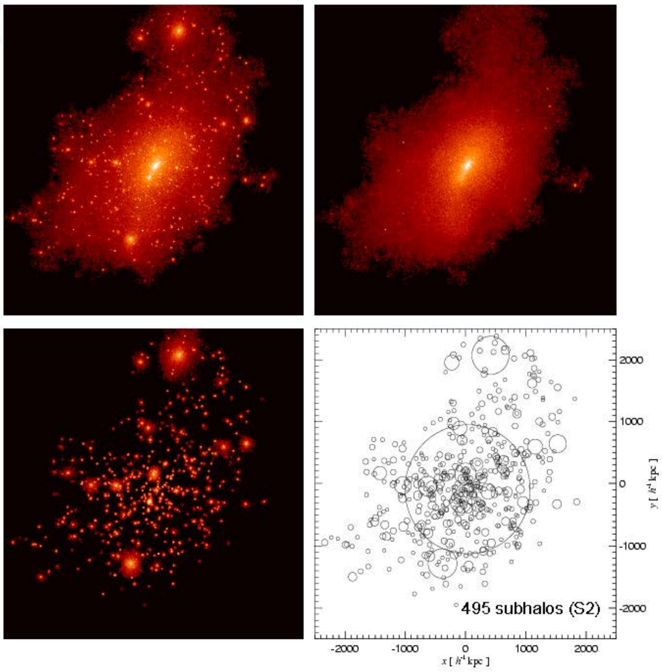
~ 20 million particles within virial radius of cluster

Springel, White, Kauffmann, Tormen (1999)



The subhalos formed in high-resolution simulations of cold dark matter can be reliably detected and extracted

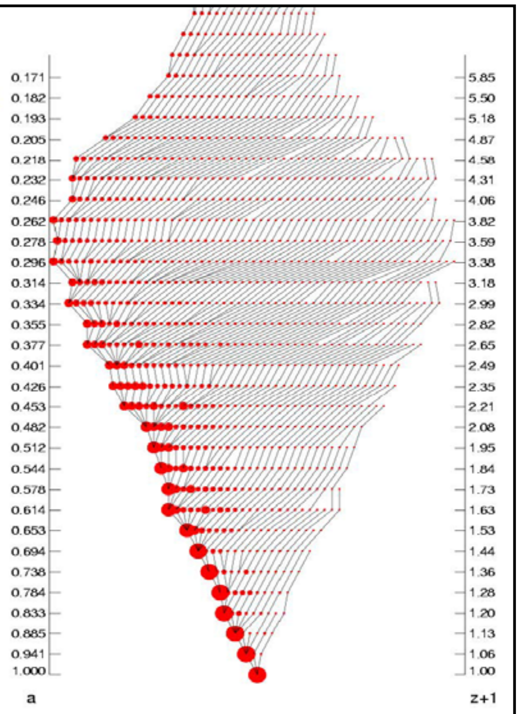
SUBHALOS IN THE S2 CLUSTER IDENTIFIED WITH SUBFIND



Semi-analytic Galaxy Formation & Subhalos

Analysis of many simulation outputs allows a measurement of the hierarchical build up of dark matter halos

FOLLOWING DARK MATTER IN TIME



Merger tree of a cluster
(only progenitors above a minimum mass are shown)

Semi-analytic models are one of the most powerful techniques to study **galaxy formation**

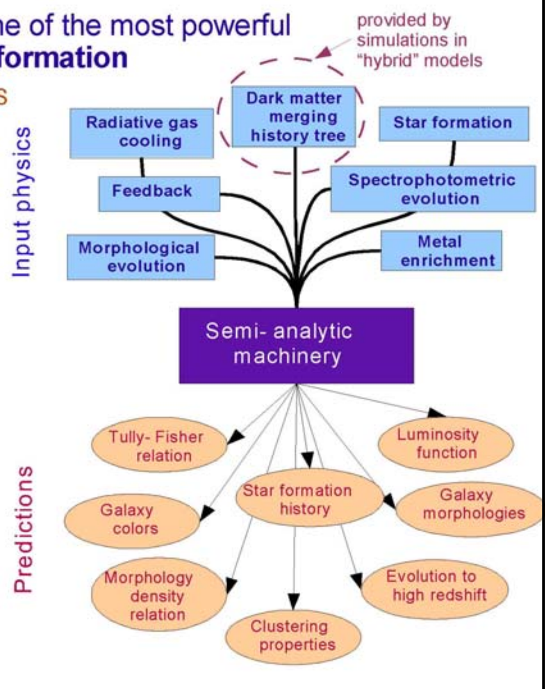
MOST IMPORTANT INPUT PHYSICS

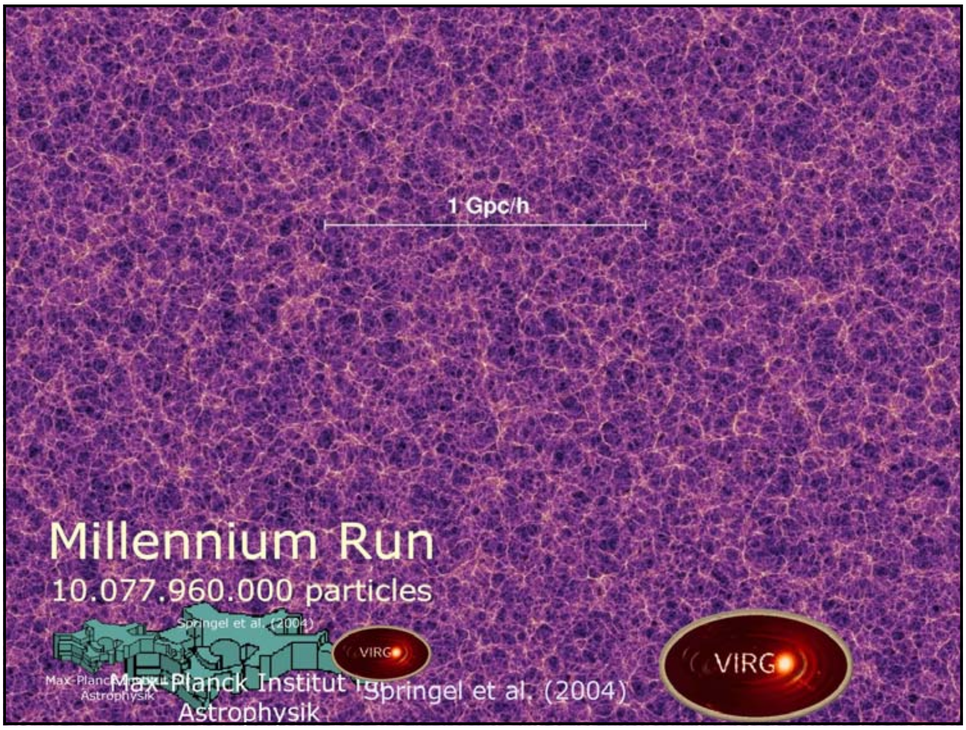
Hierarchical growth of dark matter halos
→ *understood with high accuracy*

Radiative cooling of gas within halos (dissipation)
→ *in principle well within reach of current simulations, yet plagued with numerical difficulties*

Star formation and associated feedback processes
→ *highly uncertain physics, numerically extremely difficult*

Spectrophotometric modeling of stellar populations
→ *some uncertainties, but no/small coupling to gas dynamics*

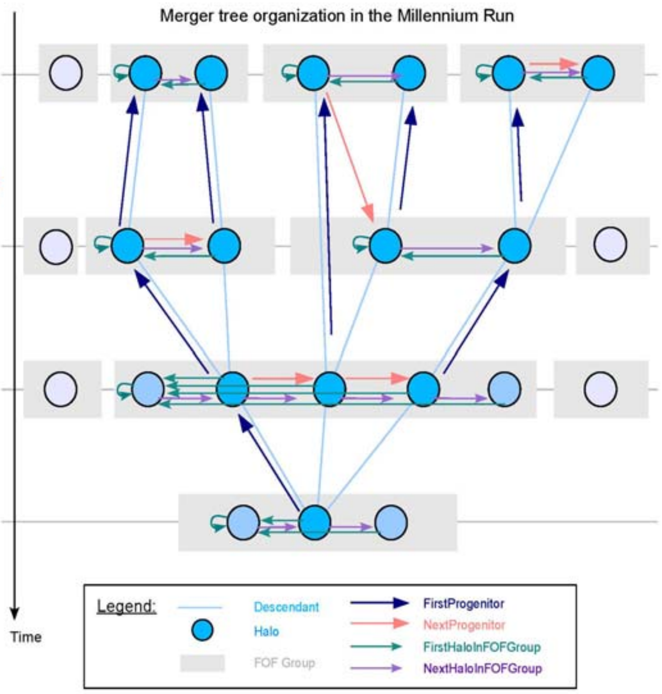




The semi-analytic merger-tree in the Millennium Run connects about 800 million subhalos

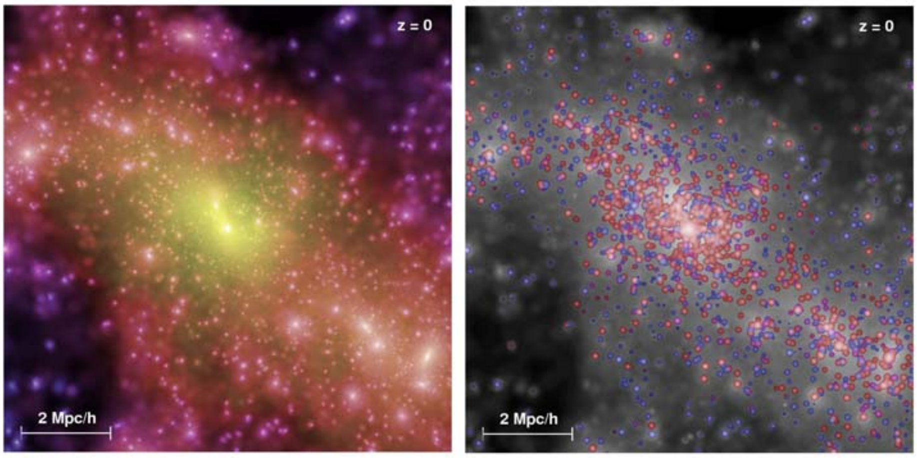
SCHEMATIC MERGER TREE

- The trees are stored as self-contained objects, which are the input to the semi-analytic code
- Each tree corresponds to a FOF halo at z=0 (not always exactly)
- The collection of all trees (a whole forest of them) describes all the structures/galaxies in the simulated universe



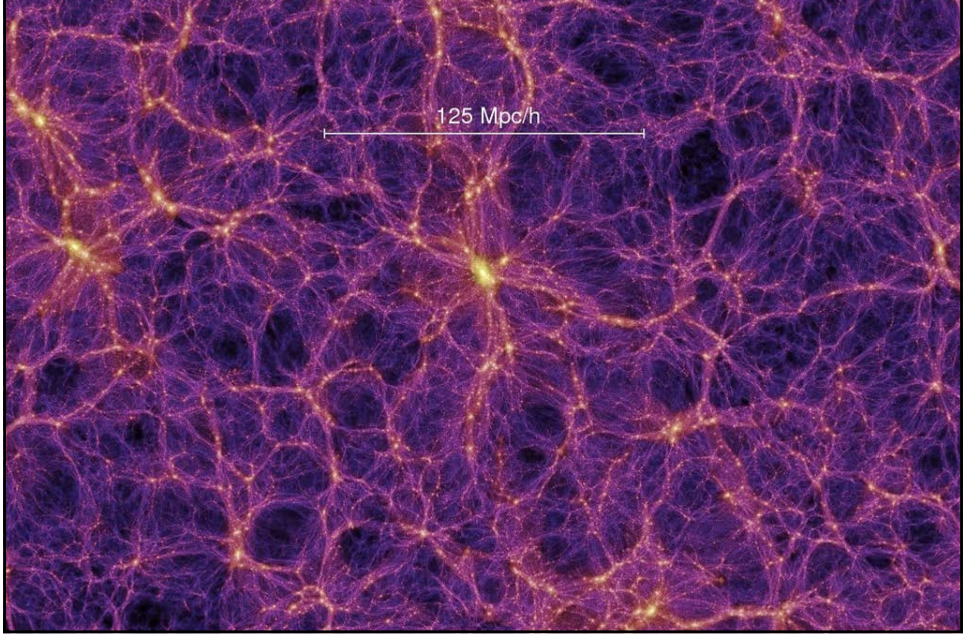
The merger tree in the Millennium simulation describes the orbits of all galaxies brighter than about 0.1 L_*

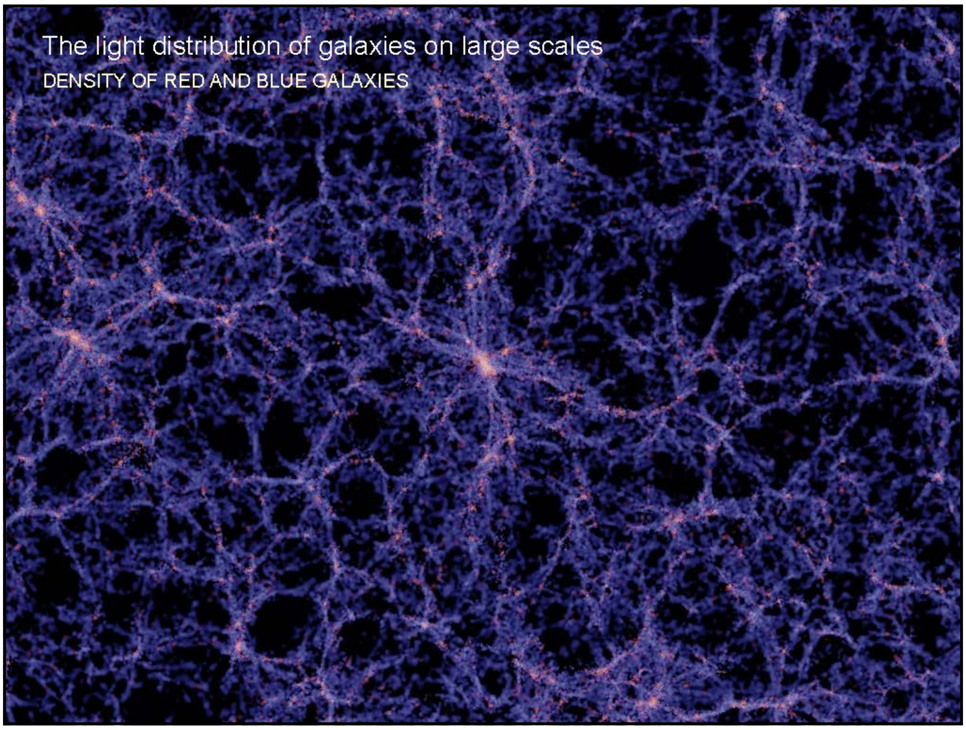
DARK MATTER AND GALAXY DISTRIBUTION IN A CLUSTER OF GALAXIES



The distribution of dark matter on large scales

DARK MATTER DENSITY, COLOR-CODED BY DENSITY AND VELOCITY DISPERSION




The light distribution of galaxies on large scales
DENSITY OF RED AND BLUE GALAXIES