Parallel Gravity Simulations

Jasjeet Singh Bagla

Harish-Chandra Research Institute, Allahabad

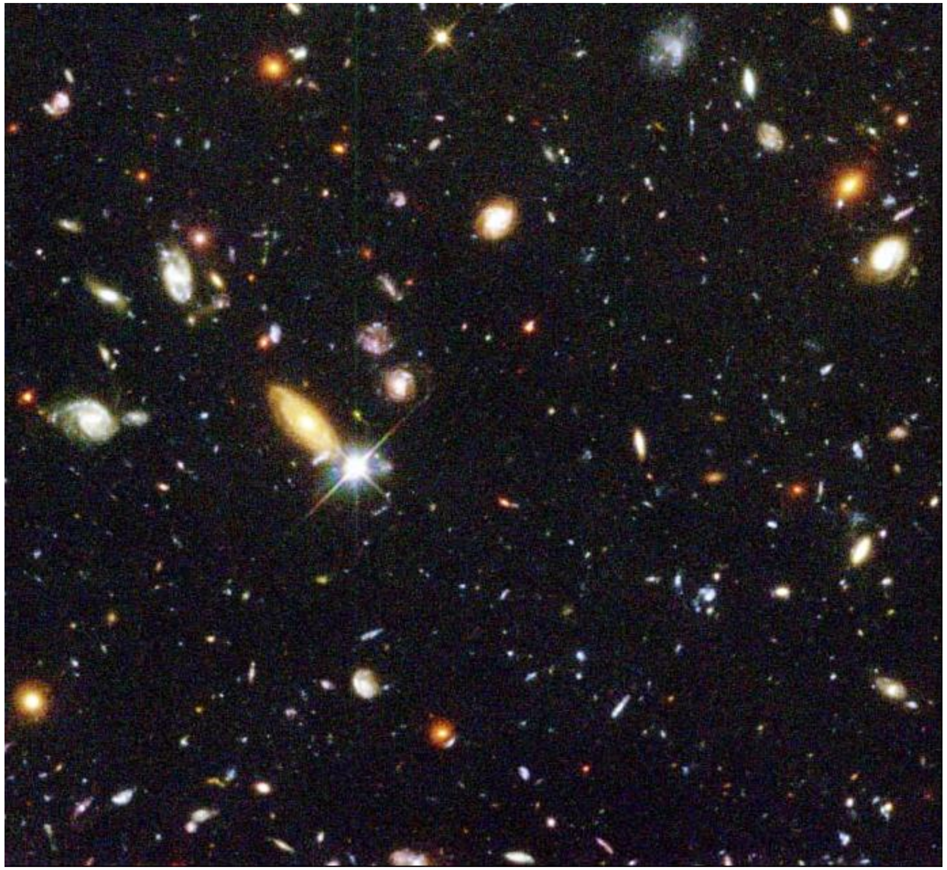
jasjeet@mri.ernet.in

http://www.mri.ernet.in/~jasjeet

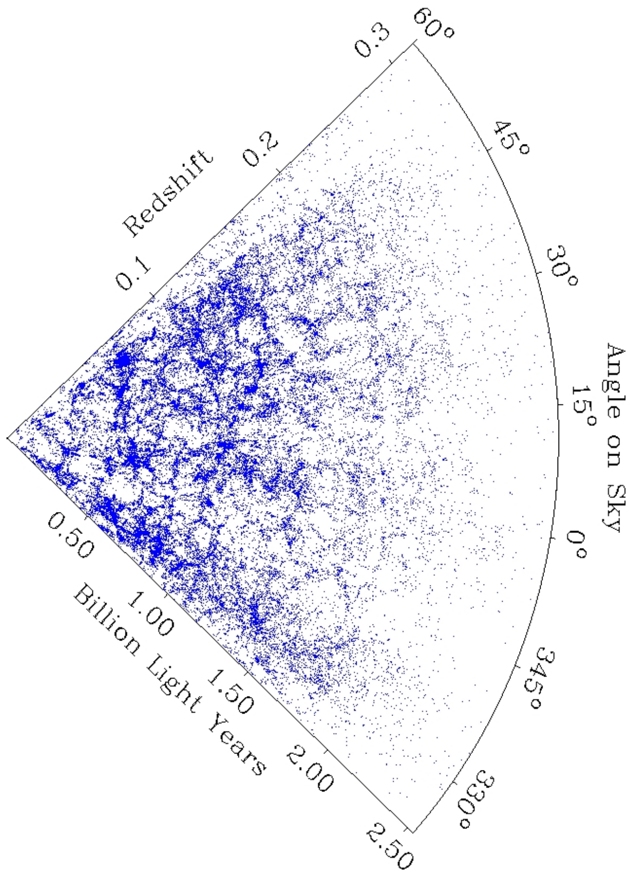
- Why do it?
- What is the problem?
- Solutions.

Large Scale Structures in the Universe

- The Universe contains very large "structures".
  - → Galaxies typically have masses in the range $10^9 - 10^{12}\mathrm{M}_\odot$, and are between $1 - 30$kpc (1pc $\approx 3 \times 10^{16}$m) across. Galaxies seem to be in equilibrium.
  - → Clusters of galaxies are large groups of galaxies (up to 10^3) in, or close to, hydrostatic equilibrium. Clusters can be up to a Mpc in diameter.
  - → Outside clusters, galaxies are found in filaments and sheets that surround largely "empty" regions. These "empty" regions are called voids. Voids can be up to 100Mpc across.



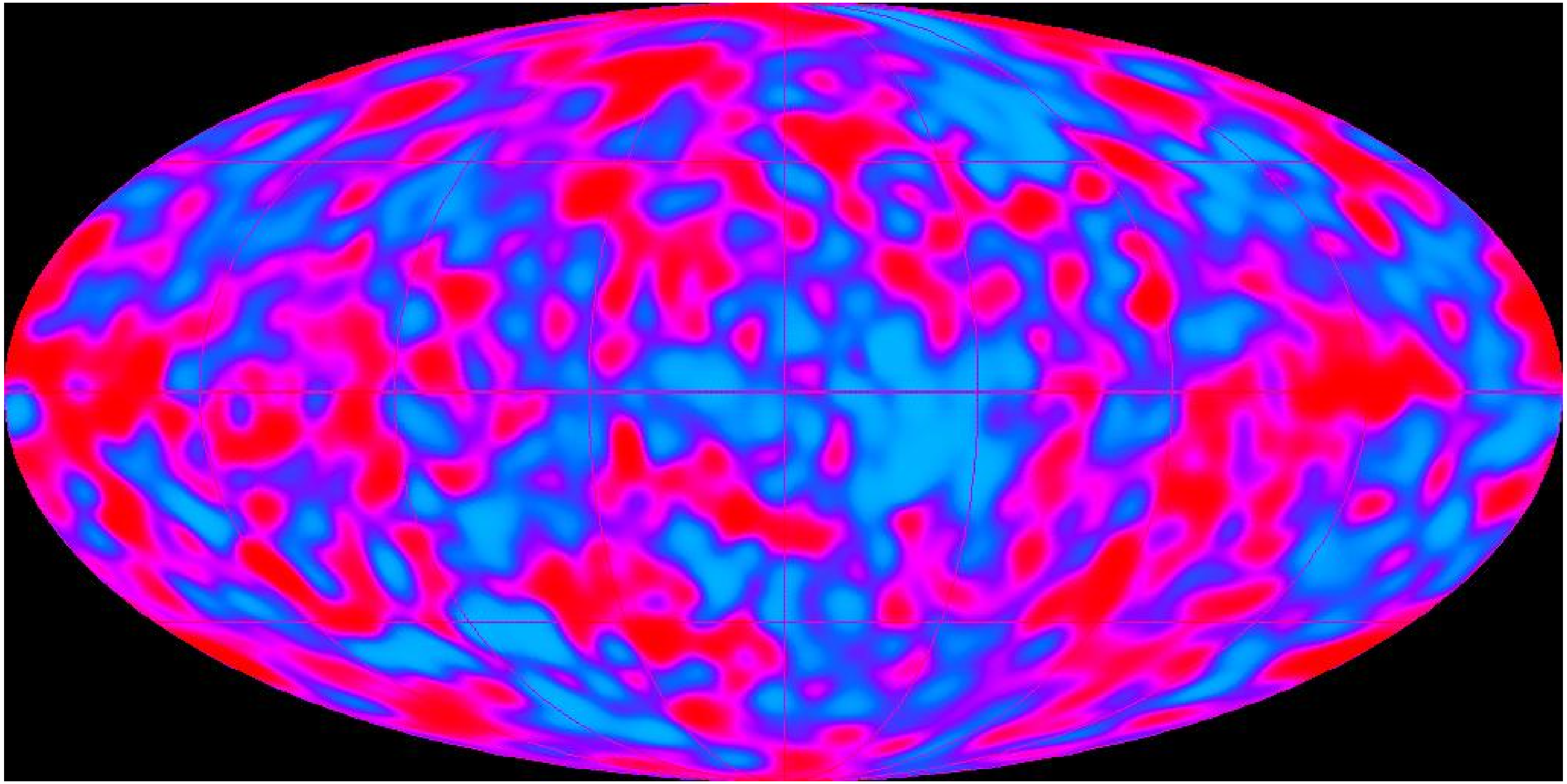
The Hubble Deep Field (North)



2dF Galaxy Redshift survey (100K release).

Evolution of LSS

- Observations of Cosmic microwave background radiation show that the Universe is nearly isotropic. Deviations from isotropy occur at a level of 10^{-5}.
- Galaxy distribution shows inhomogeneities and anisotropies at a level greater than 1 at small scales. Even at the largest scales probed, fluctuations in number density of galaxies is much larger than 10^{-5}.
  - $\Rightarrow$ Fluctuations in the Universe have been amplified by some process. Gravity is the only force that can lead to such amplification at large scales in a charge neutral Universe.



COBE DMR map

Gravitational Clustering: Equations

- The equations are just Newton's law of motion and gravitation clubbed together.

$$\ddot{\mathbf{r}}_i = -G \sum_{j=1}^{N} \frac{m_j \mathbf{r}_{ij}}{r_{ij}^3}$$

  where $\mathbf{r}_i$ is the position of a particle, m_i is its mass and $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$.
- The problem is that this is an $O(N^2)$ operation. It is simply undoable for more than a few thousand particles, and we need more than 10^6.

Barnes-Hut tree Method

- Set up an oct-tree structure for the particle distribution.
- The simulation box is the "trunk". This is sub-divided into smaller volumes/cells at each level. The cells are "branches". Cells are to be sub divided till there is at most one particle in the smaller cells. Particles correspond to "leaves".
- Force is computed by adding contribution of distant cells instead of adding contribution of particles individually.
- The most popular cell acceptance criterion is $d/r < \theta$, where d is the size of the cell under consideration and r is the distance to the centre of mass of the cell. θ is a parameter, usually around 0.5.
- Error due to the tree approximation scales as θ^3 for generic distributions of particles.

Parallelising the tree method

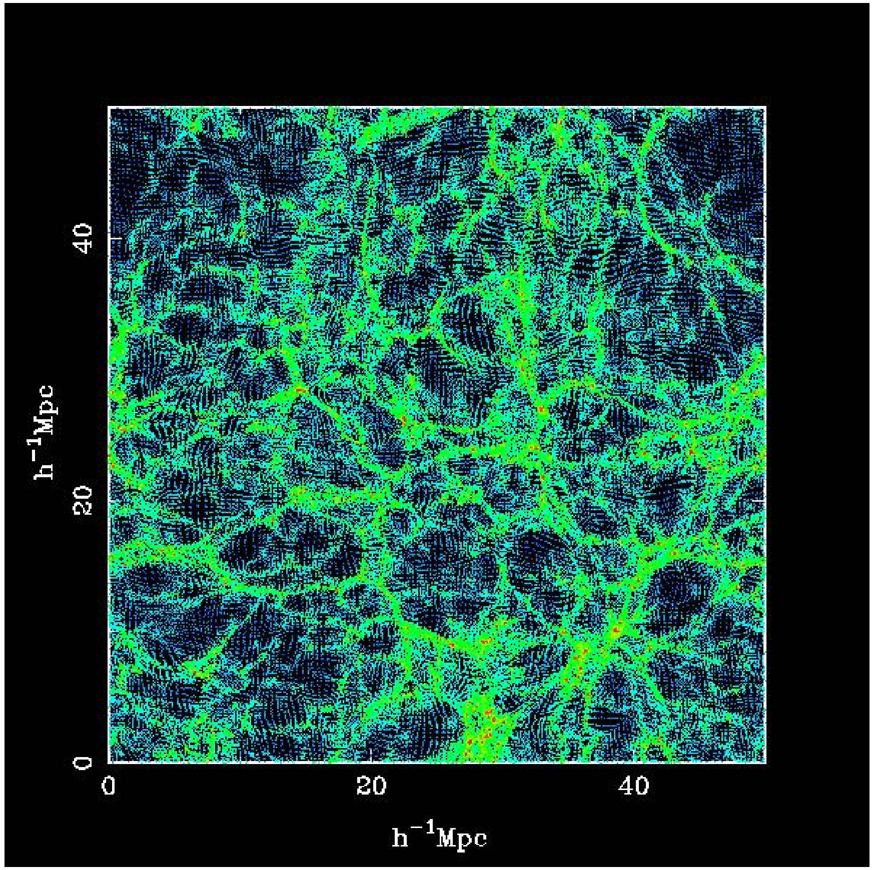
- Domain decomposition: Divide the simulation box into domains with equal computational load.
- Use Recursive Orthogonal Bisection method.
  - → Divide the simulation into two halves with equal computational load at each step.
  - → Requires 2^n computing nodes, where n is an integer.
- For a fixed n, we can set up a Cartesian mesh of computing nodes.

Domain decomposition

- Nodes will exchange particles that move out to the volume of another node. This is done after every time step.
- Domains need to be re assigned after every few steps to ensure load balancing.
- Every node communicates to every other node after each time step. Communication overhead is very high for large number of compute nodes.

An improved method

- Add one communication node for every m compute nodes. Compute nodes send their data to the communication node and continue local computation (which is large).
- Communication nodes exchange all the relevant data amongst themselves, and then send the required data to each compute node in their domain.
- The ratio of number of messages exchanged in the standard domain decomposition model to this method approaches m^2 for a large number of compute nodes.
- Choice of m depends on the granularity, of course. For most simulations this number is between 8 and 32.



A slice from the simulation