PMFAST: fast parallel Particle-Mesh N-body

Hugh Merz (parallel programmer)

¥^´Ü8Li Pen

Hy Trac

CITA Canadian Institute for Theoretical Astrophysics

INCORPORATED 1996

Institut canadien d'astrophysique théorique ICAT

Cosmological N-body

- Science case: non-linear evolution of self-gravitating dark matter
- Specs: mass resolution, length resolution, ***speed***
- Applications: halo statistics, gravitational weak/strong lensing, halo structure/shapes, etc
- First two require high mass resolution, modest length resolution: perfect for PM

The Rise and Fall of PM

- Easy to code, very fast, large number of particles, memory limited. Vendor optimized FFTs achieve very high efficiency on commodity hardware: video gaming.
- Worst case for parallel machines: FFT moves all data several times per time step. O(1) communication/computation.

Parallel PM History

- Farrel+Bertschinger (1994): HPF on CM5
- PMtree: Xu/Bode, Gadget, Dubinski/Kim
- P^3M: Hydra, Shirakov
- Tree codes have swept the parallel field (gasoline, gadget, partree, ART, Humble, Salmon&Warren, …): high adaptive resolution, high computation/communication ratio

Parallel N-body Challenge

- Speed – FLOP/particle/timestep
- Memory – bytes/particle
- Latency – microsec/step
- Bandwidth – bytes/particle/step
- Portability – MPI
- Efficiency – vendor library
- Availability: free

Theoretical Limit

- Memory: 6 numbers/particle
- Computation: 6/particle+gravity O(n log n)
- Bandwidth: $O(n^{2/3})$ -- negligible
- Latency: 1 timestep – O(minutes)

Hardware Trends

- Celeron node + gigE: $200/node (fully switched), 10 Gflop/node = $20/Gflop!
- Typical codes cannot use SSE3, they run at ~1% of peak speed. FFT's run at over 50% of peak.
- Low latency expensive: Infiniband = $1000/node (4 _s, 30x faster than GigE).
- Local network cheap: 3-D torus

Parallel FFT

Density: Cubic-slab (move all data)

2x1-D FFT

Transpose

1-D fft

Slab-cubic

Do forces

Data moves 3x, only 2.5 $\log_2(N)$ operations.

12 bytes moved,100 operations.

P4: 10 GFlop/sec, GigE: 0.1 GB/sec,
network back plane usually much worse.

Gravity

- Non-local: need for high bandwidth, low latency?
- NO: long range force is smooth. E.g. Tree code.
- Decompose force into short range (which determines communication costs) and long range (can be done on coarse grid)

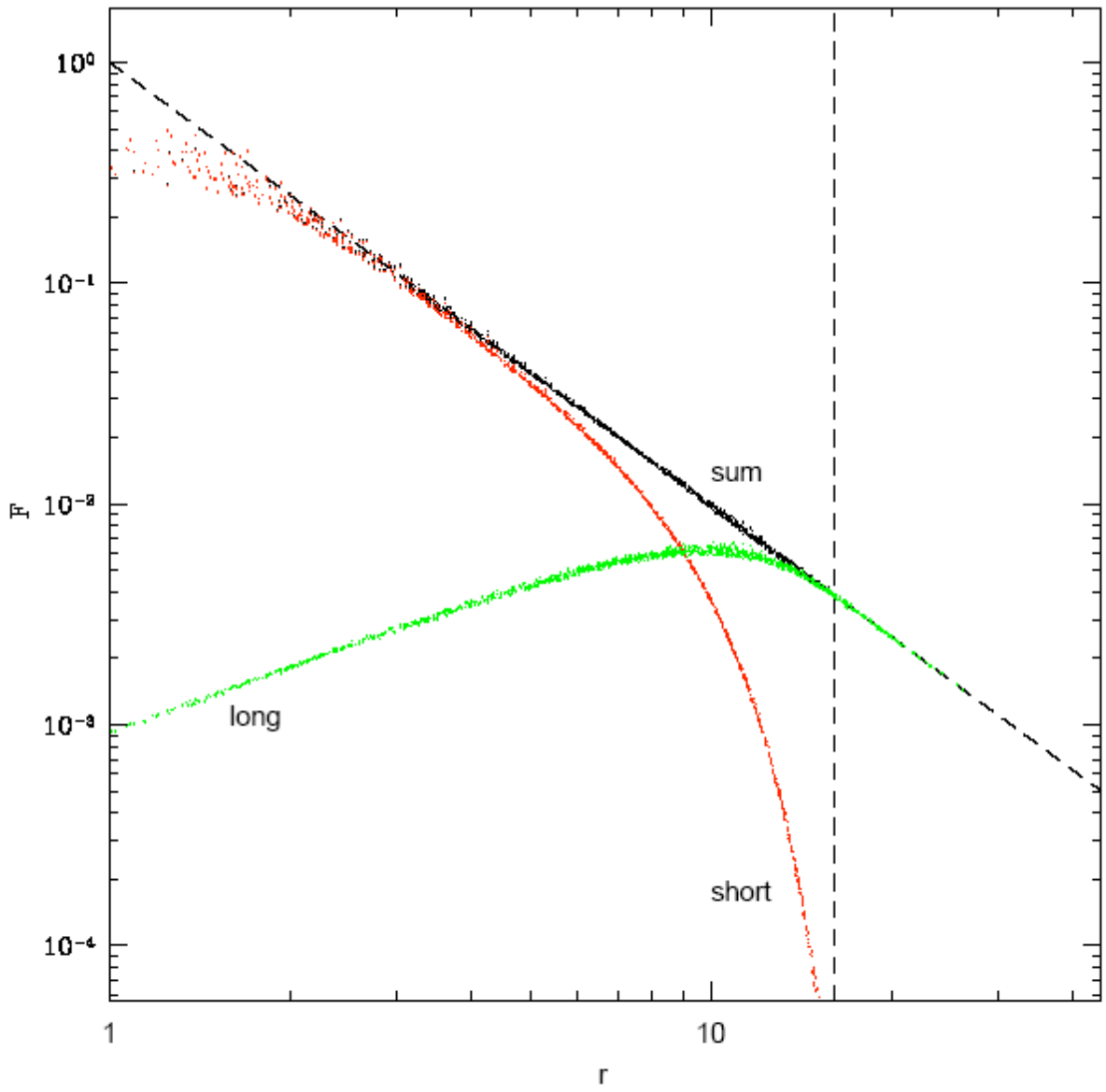


Figure 1. Short and long range force as determined from random particle pairs. r-axis is measured in fine grid cells, with the short-range cutoff indicated by the dashed vertical line.

Computing Kernels

a short-range component

$$w_s(r) = \begin{cases} w(r) - \alpha(r) & \text{if } r \leqslant r_c, \\ 0 & \text{otherwise,} \end{cases}$$

F^{fine}

and a long-range component

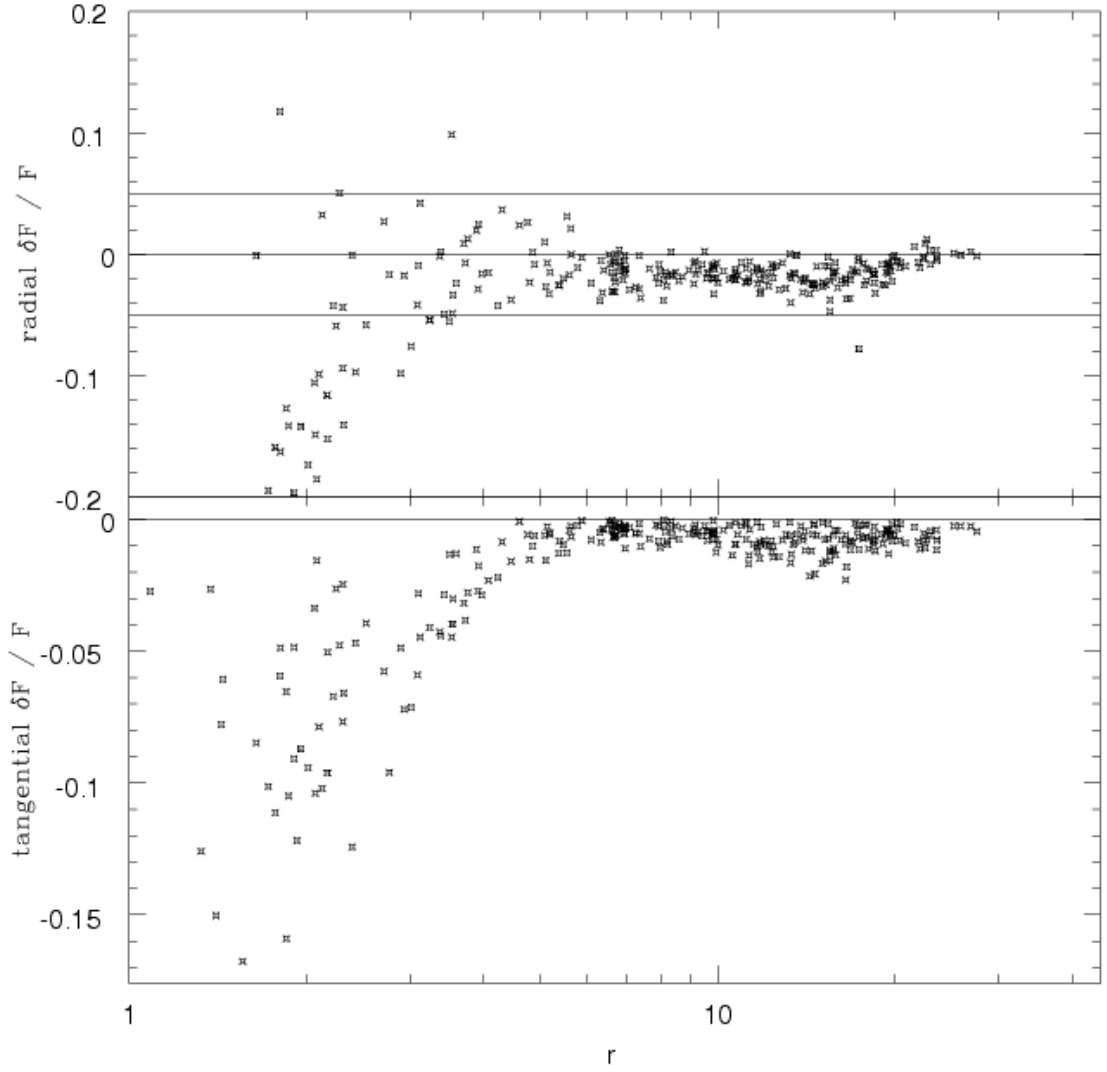
$$w_l(r) = \begin{cases} \alpha(r) & \text{if } r \leqslant r_c, \\ w(r) & \text{otherwise,} \end{cases}$$

$$F^{\mathbf{grid}} = F^{\mathbf{coarse}} + F^{\mathbf{fine}}$$

$$\epsilon = \sum_i \left[F^{\text{grid}}(\Delta x_i) - F^{\text{exact}}(\Delta x_i) \right]^2 w$$

Least squares solution for the grid force

16/4 LSQ matching error



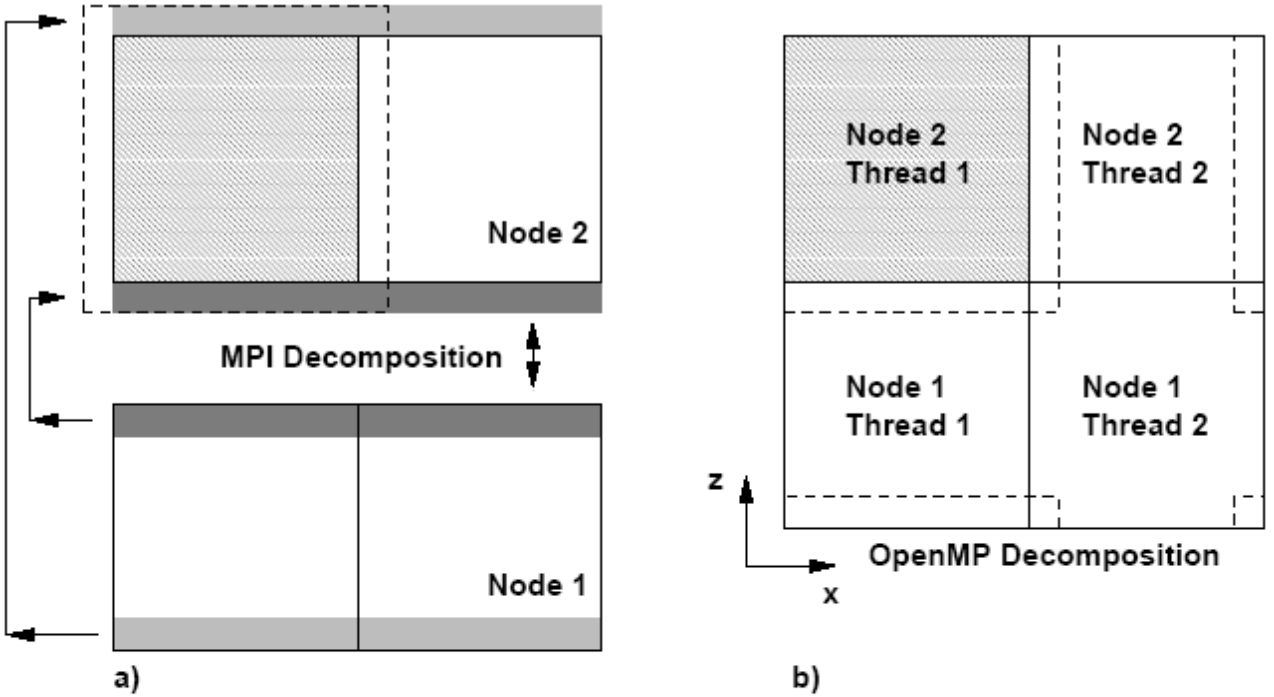


Figure 3. a) An example of the data decomposition on the fine mesh for 2 nodes using 2 threads per node. The local data for each fine mesh is bounded by the solid line and the mesh boundary including buffer region is the dashed line. The buffer regions acquired from the adjacent node are indicated. b) The same data set, showing the fine mesh overlap. The fine grid is only stored for the region that is actively worked on, which reduces memory overhead.

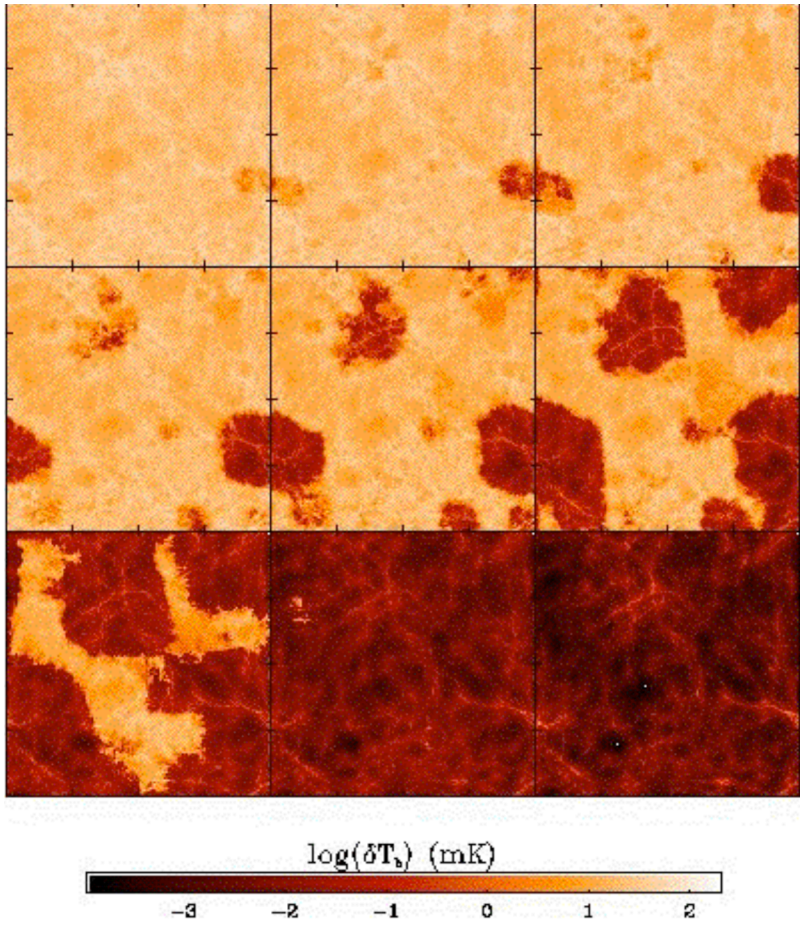
PMFAST architecture

- 3712^3 fine grid cells, 928^3 coarse grid, 1856^3 (~6 billion) particles on CITA 32 proc itanium-1 cluster
- 2 time steps/hour
- 200 Mpc run takes a week.
- Initial conditions with similar 2-scale decomposition (Trac 2004)

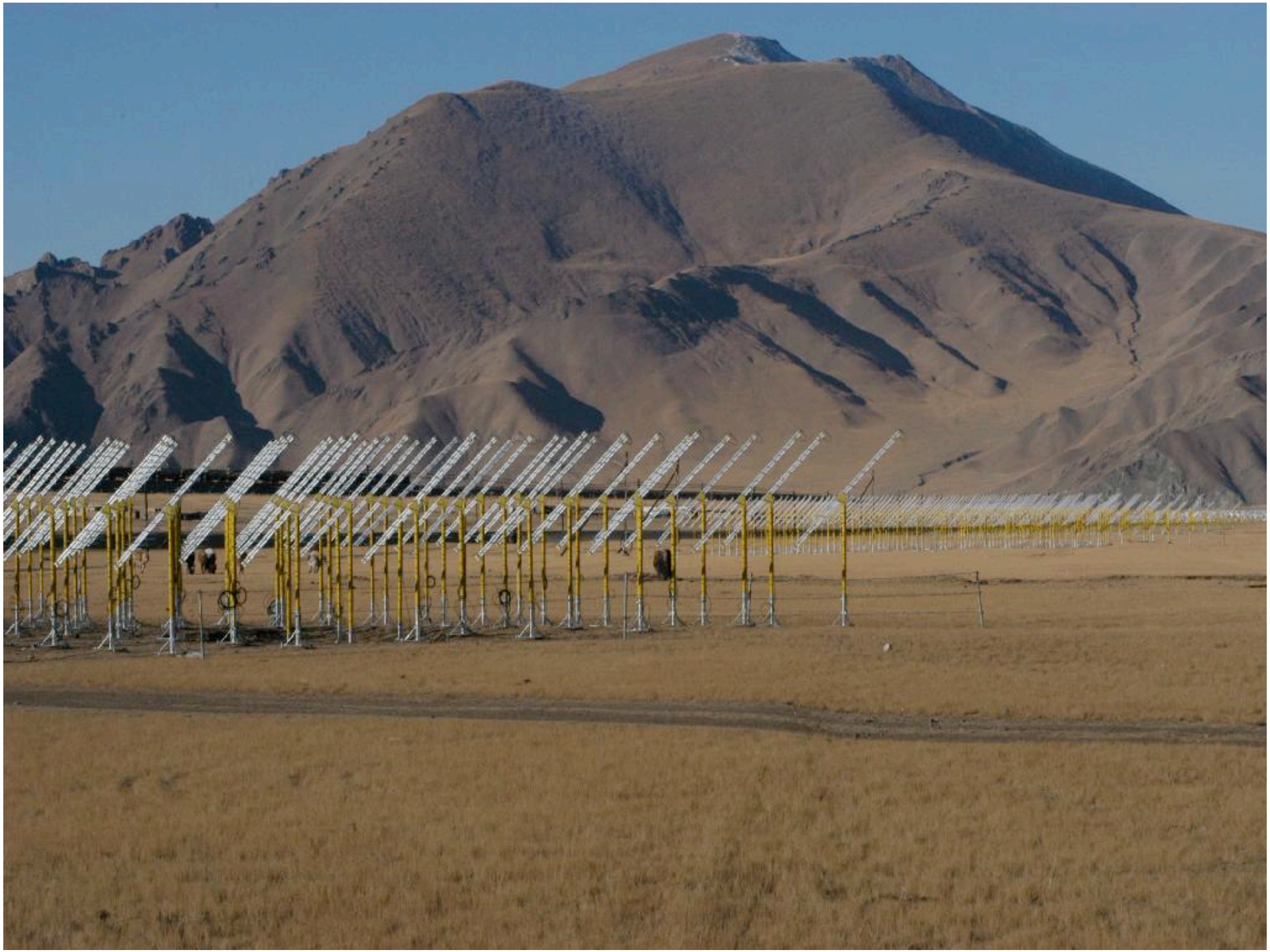
Animation

Reionization

- First objects:
- 21cm @ 20>z>6
- 70-200 Mhz: TV 4-11
- ¶§ = 23 mK, ~0.3 mJy
- Angular scale 5'<¶®20', freq res 500 khz



z=10 simulation, Furlanetto et al, 2004

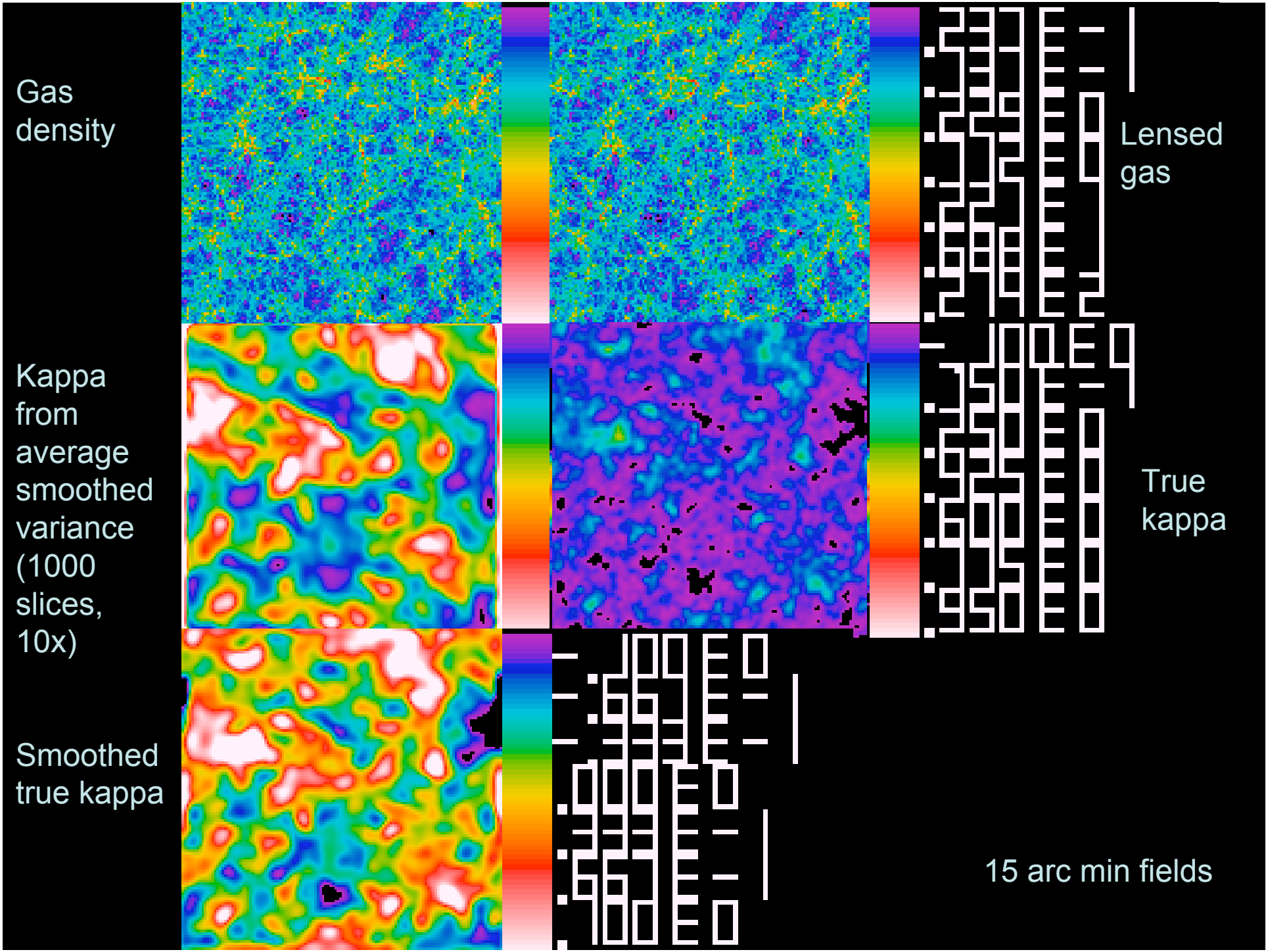


Lensing of diffuse sources (w/ T. Lu)

- Reionization structures are ideal lensing sources: high-z, small scale structure, full redshift info.
- Naive procedures: variance maps
- optimal shear and magnification reconstruction
- power spectrum estimation

Naive variance maps

- Scenario: single lens plane at z_l, many source planes at z_s, _(x) (Pen 2003)
- Smooth source at fixed angular scale
- Power spectrum makes variance dependent on smoothing scale
- At fixed point on sky, average variance along the line of sight gives kappa


Gas density
Lensed gas
Kappa from average smoothed variance (1000 slices, 10x)
True kappa
Smoothed true kappa
15 arc min fields

Optimal max likelihood

lensing changes length scale relations:

$$k \rightarrow k' = k[1 + \kappa + \gamma_1 \cos(2\theta) + \gamma_2 \sin(2\theta)]$$

to first order in κ:

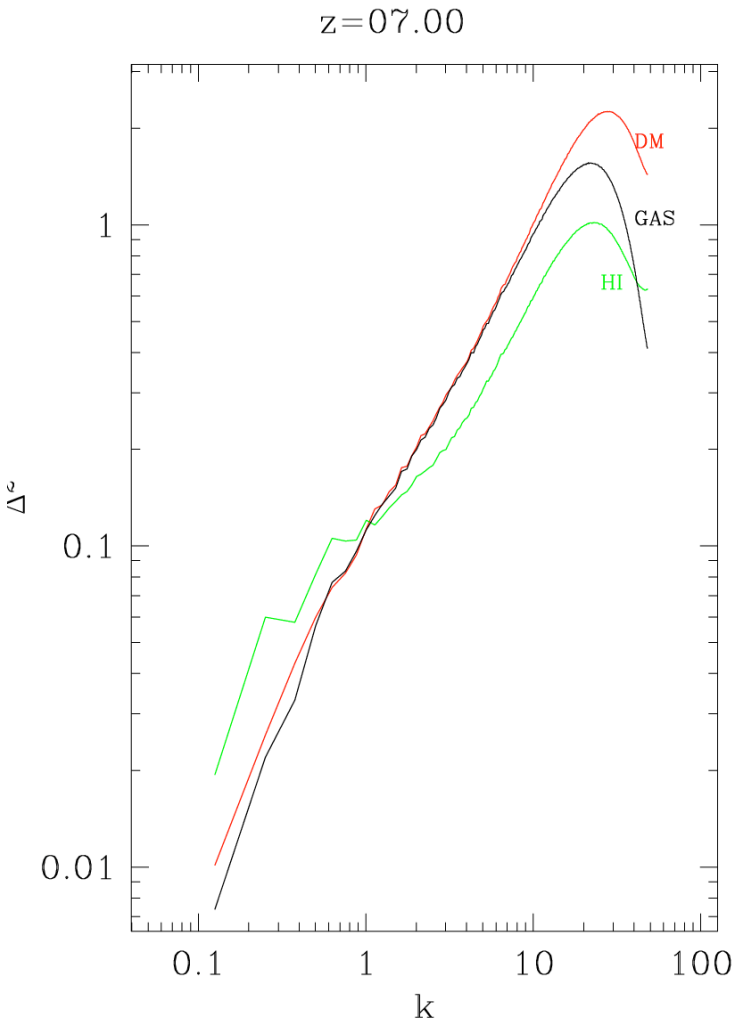
$$\frac{k'^3}{2\pi^2}\langle\delta(k')\delta(-k')\rangle \equiv \Delta^2(k') = \Delta^2(k) + \Delta^{2\prime}(k)k\kappa$$

The likelihood

$$P[\delta(k)] = \left[\Pi\Delta^2(k)\right]^{-1/2} \exp\left[-\int \frac{\delta(k)^2}{\Delta^2(k)}\right]$$

so differentiating its log for κ,

$$\begin{aligned}
\mathcal{L} &\equiv -2\log(P) = \int \log(\Delta^2(k')) + \int \frac{\delta(k)^2}{\Delta^2(k')} \\
\frac{d\mathcal{L}}{d\kappa} &= \int \frac{\Delta^{2\prime}k}{\Delta^2} - \int \frac{\delta^2\Delta^{2\prime}k}{(\Delta^2)^2} = 0 \\
\kappa &= \frac{\int \frac{\delta^2\Delta^{2\prime}}{(\Delta^2)^2} - \frac{\Delta^{2\prime}}{\Delta^2}}{\int \left(\frac{\Delta^2}{\Delta^2}\right)^2} \\
&= N\int [\delta(k)W(k)]^2 - N_0
\end{aligned}$$



Conclusions

- PMFAST highly efficient: asymptotically no memory overhead, computing is FFT dominated, uses IPP FFT libraries
- Works well on slow networks: cheap clusters.
- Freely available: http://www.cita.utoronto.ca/webpages/code/pmfast/
- Current public version slab decomposed, cubic version under test.
- P^3M under study.