CoMD in Chapel: The Good, the Bad, and the Ugly

Chapel Lightning Talks : Birds-of-a-Feather

November 18, 2014

David Richards, Luc Jaulmes

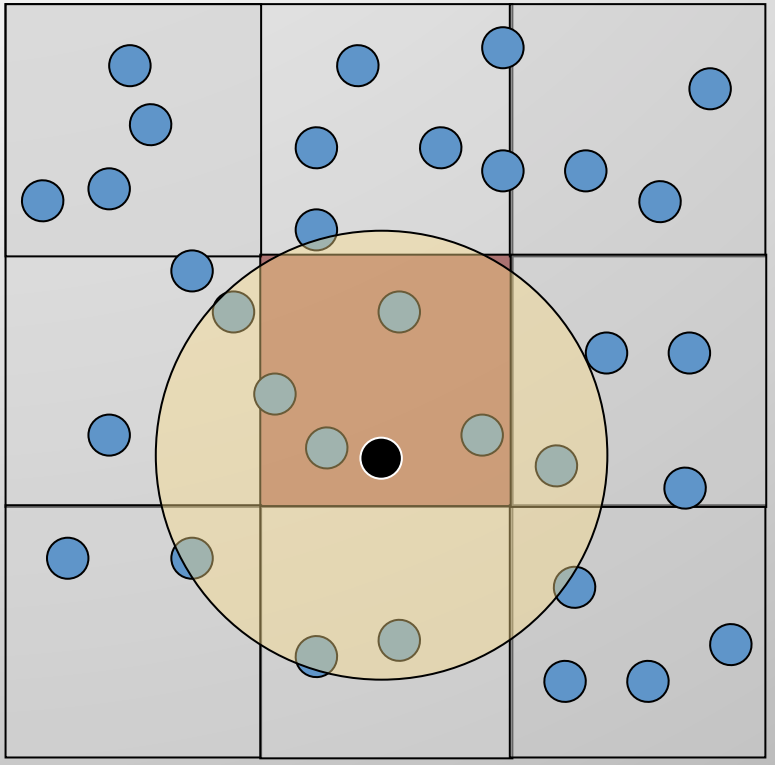
Lawrence Livermore National Laboratory

LLNL-PRES-664282

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC

Dramatis Personae

2014 LLNL Co-Design Summer School



CoMD Proxy App for Molecular Dynamics

The Good:

- The code works!
- Luc enjoyed programing in Chapel
- Very expressive language
- Dense, elegant code
  - Thanks to built-in domain maps and iterators
  - e.g., `KineticEnery = + reduce m*norm2(v)`
- Rich language constructs encourage algorithmic exploration
  - e.g., sparse subdomains

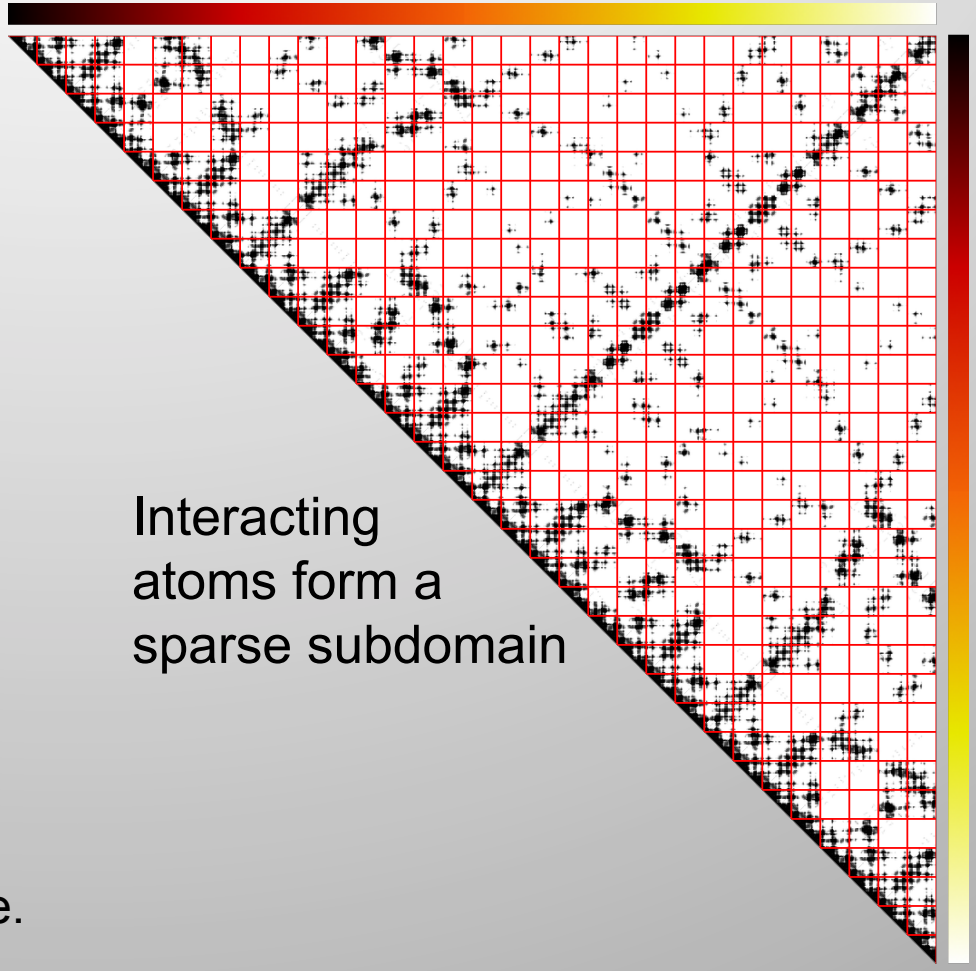
The Bad: Reach Exceeds Grasp

The code Luc wanted to write:

```
forall(i,j) in N {
   var (f_ij,pe_ij) = force( x(i), x(j) );
   f(i) += f_ij;
   f(j) -= f_ij;
   pe(i) += pe_ij / 2;
   pe(j) += pe_ij / 2;
}
```

Actual code turned out to be about 70 lines including mutex constructs to manage race conditions.

Parallel iterators might help, but we were unable to implement an example.



Interacting atoms form a sparse subdomain

The Ugly: Domain Decomposition

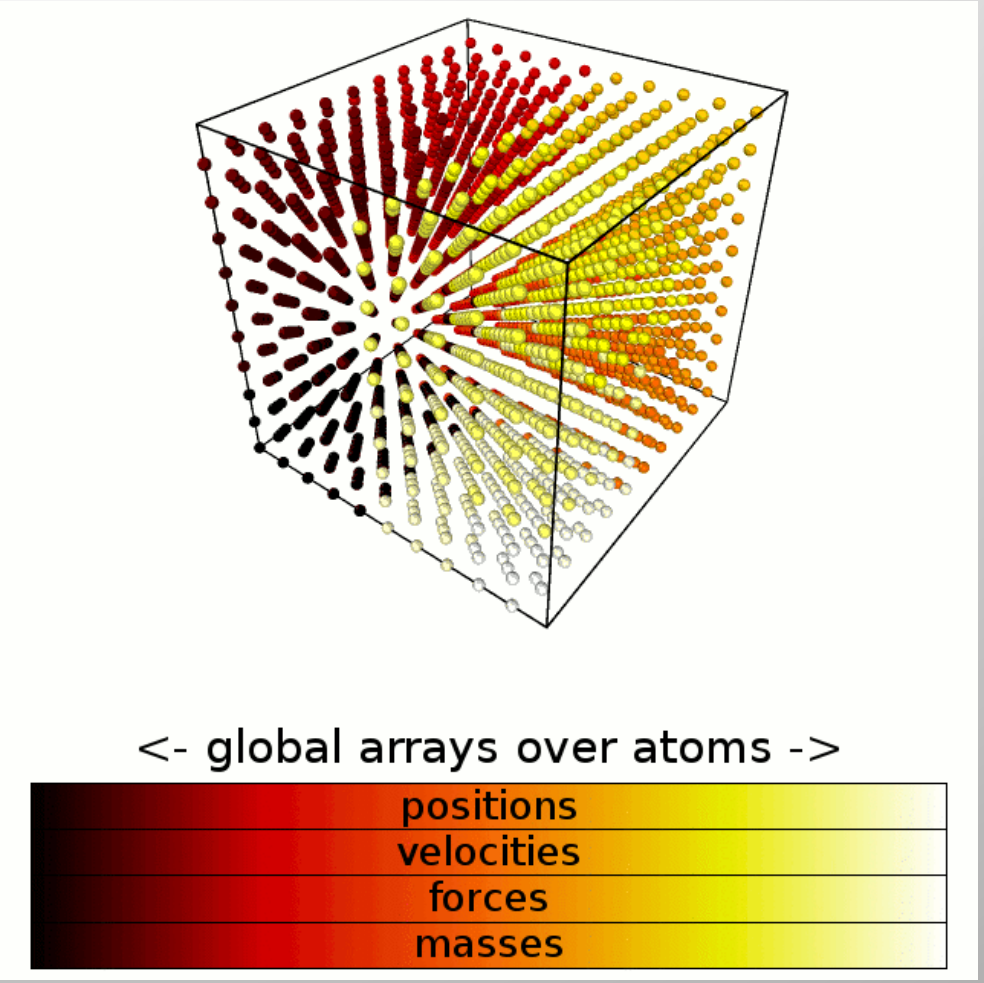
Problem:
Static Domain Maps produce poor data locality

Solution:
Sort atoms by position along a space filling curve to obtain locality

Good:
Sort implementation is 200 lines

Bad:
No data on performance or scalability

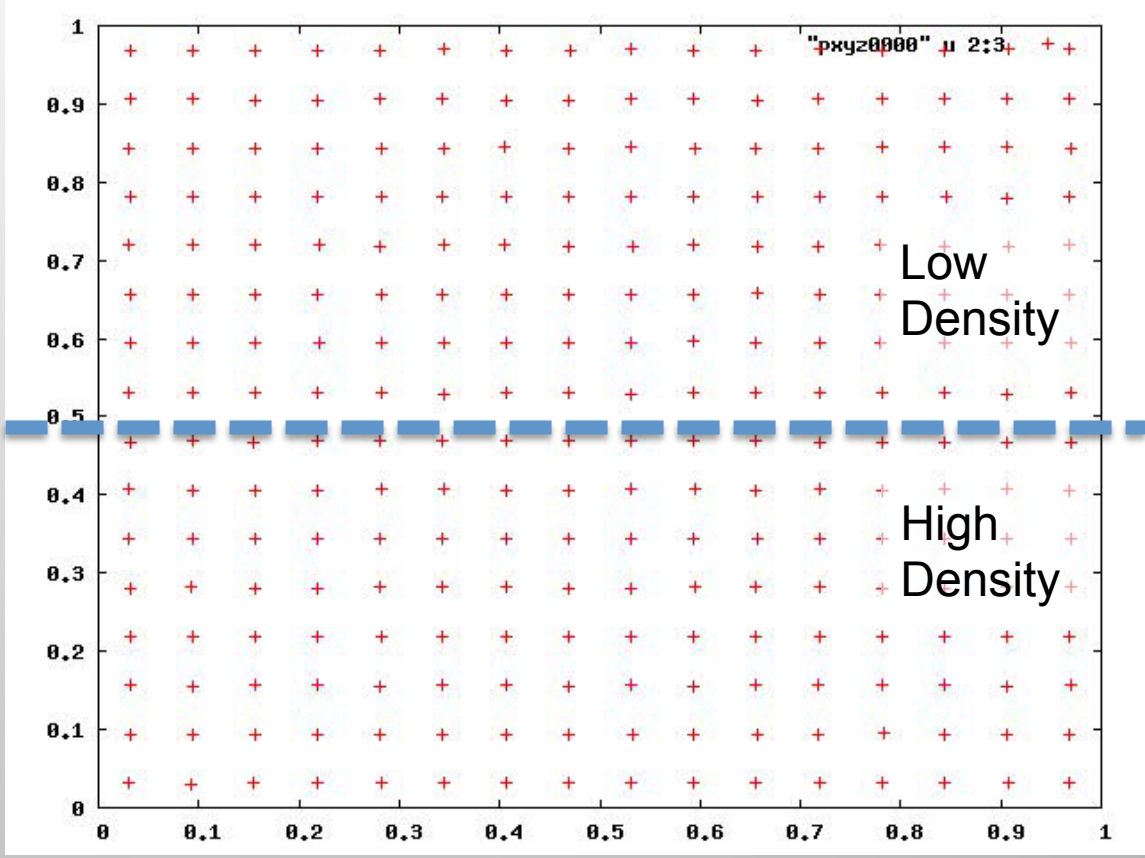
Ugly:
Dynamic Domain Maps not yet available



The Sequel: Koradi Test Problem

Koradi load balance algorithm:

- Moveable domain centers
- Atoms assigned by Voronoi diagram
- Distance calculated with biased metric
- Distance is larger for overloaded tasks



Lawrence Livermore
National Laboratory

Students and Models

- Sam Reeve (Purdue) LAMMPS, CoMD, leanMD
- Riyaz Haque (UCLA), CNC (Habenero-C)
- Luc Jualmes (Barcelona) OmpSs, Chapel
- Sameer Abu Asal (LSU) C++11 futures, HPX
- Aaron Landmehr (Delaware) OCR
- Sanian Gaffer (NM-State) UPC++
- Gheorghe-Teodor Bercea (Imperial) PyOP2
- Zach Rubinstein (Chicago), Troels Henriksen (Copenhagen) Embedded DSL