IPCC @ RWTH Aachen University

Optimization of multibody and long-range solvers in LAMMPS

Rodrigo Canales **William McDoniel** Markus Höhnerbach
Ahmed E. Ismail Paolo Bientinesi

IPCC Showcase – November 2016

Team

RWTH

Prof. Paolo Bientinesi

Rodrigo Canales

William McDoniel

Markus Höhnerbach

Prof. Ahmed Ismail

Intel

Georg Zitzlsberger

Klaus-Dieter Oertel

Michael W. Brown

Introduction

2015

May:	**Kickoff – IPCC @ RWTH Aachen** *Optimizing LAMMPS kernels*
Oct.:	First results on Xeon & KNC, @ EMEA IPCC

2016

Feb.:	Showcase 1st year
March:	First results on KNL, @ IPCC & IXPUG Forum
May:	KNL Access
Nov.:	**Showcase**

2017

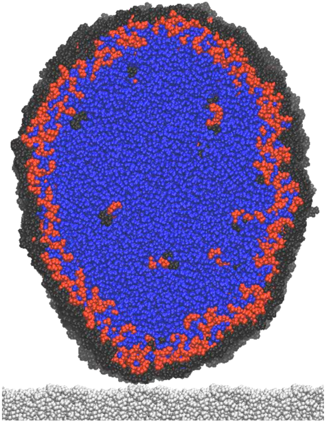
May:	End 2nd year

Agenda

- Intro to MD, LAMMPS
- Achievements 1st year
- Goals & Progress 2nd year
  - AIREBO
  - REBO
  - PPPM Electrostatics
  - PPPM Dispersion
- Future Projects

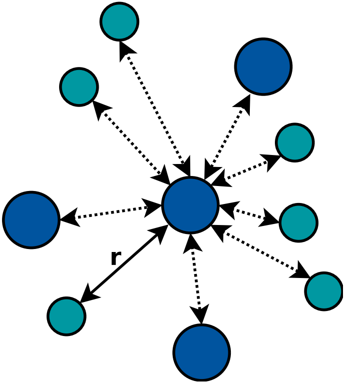
LAMMPS

Large-scale **A**tomic–**M**olecular **M**assively **P**arallel **S**imulator



- Sandia National Labs
  `http://lammps.sandia.gov`
- Widely used open source MD code
- Support for OpenMP, Xeon Phi, and GPU (CUDA and OpenCL)

Molecular Dynamics



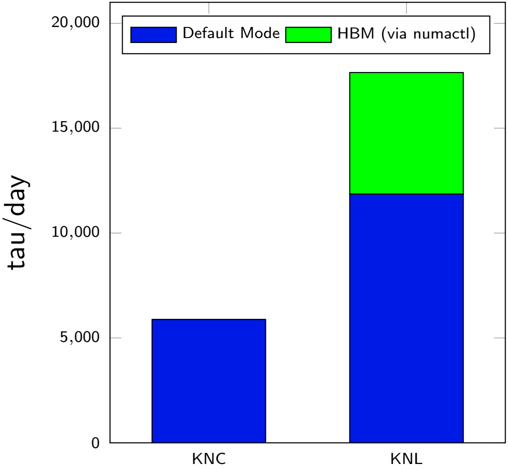
- Many particle systems
- Computes interactions between pairs of atoms

$$\Phi_{LJ} = 4\epsilon\left[\left(\frac{\sigma}{r_{ij}}\right)^{12} - \left(\frac{\sigma}{r_{ij}}\right)^{6}\right]$$

First Year

- Pair Potentials
- KNL Ready

Buckingham: KNC vs. KNL - Full Node



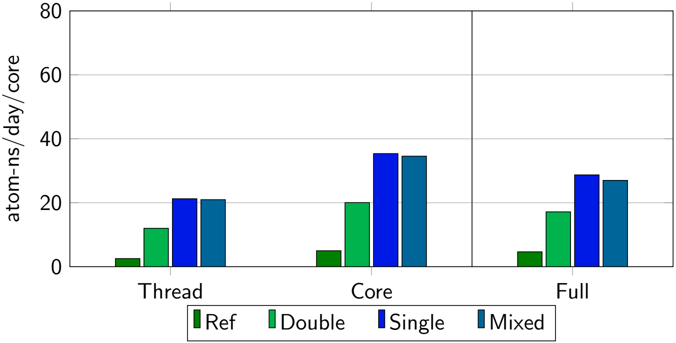
Tersoff: KNC

Thread	
Cores	1
SMT	1
Atoms	32.000

Core	
Cores	1
SMT	4
Atoms	32.000

Full	
Cores	60
SMT	4
Atoms	512.000

Measurements in 1000 atom-ns/day/core, SMT minimizes runtime.



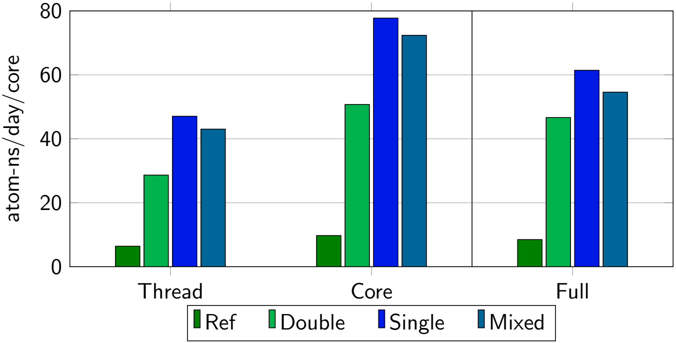
Tersoff: KNL

Thread	
Cores	1
SMT	1
Atoms	32.000
HBM	Yes

Core	
Cores	1
SMT	4
Atoms	32.000
HBM	Yes

Full	
Cores	64
SMT	4
Atoms	512.000
HBM	Yes

Measurements in 1000 atom-ns/day/core, SMT minimizes runtime.



The Vectorization of the Tersoff Many-Body Potential: An Exercise in Performance Portability

- Initial work: workshop on MD simulation software @ SC'15
  - Full portability across existing Intel archs
  - Focus on vector operation wrapper
- Submitted to SC'16 technical program
  - Additional architectures
  - KNL results
- KNL measurements via Mike (Thanks!)
- For submission: NDA waiver
- **Accepted**
- **Best Student Paper Finalist**
- **(Maybe) part of replication initiative SC'17**

Second Year (After Q2)

- Multi-body Potentials
- Long Range Interactions

Multi-body Potential: REBO

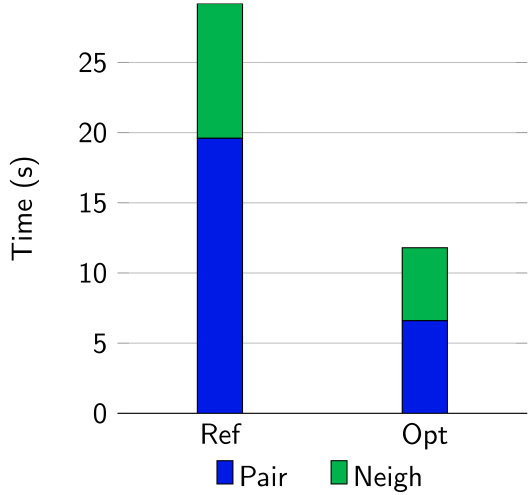
- Similar to Tersoff
- Applicable to Carbohydrates
- Improves Tersoff through additional terms

- Additional neighbor finding routines needed by REBO (Ready)
- Vectorized/Optimized code for KNC/KNL (Ready)
- Optimized code for CPU, same approach as Tersoff (Ready)

- Vectorized/Optimized code for CPU (In Progress)
- Offloading Performance (In Progress)

- Speedup KNL: ca. 2.5x total, and ca. 3x on kernel
- Bottleneck: Neighbor Lists

REBO Results – KNL

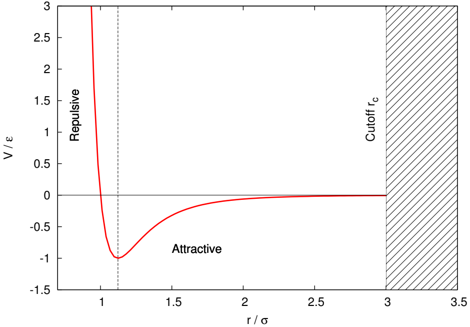


AIREBO

- Based on REBO
- Two additional terms: *Torsion* and *Lennard-Jones*
- *Torsion:* Easy to vectorize (Ready)
- *Lennard-Jones:* Hard to vectorize (In Progress)
- Search through neighbor list and branch
- Idea: Separate expensive and cheap cases

Long Range Interactions: PPPM

- Cutoff distances make pair potential calculations feasible



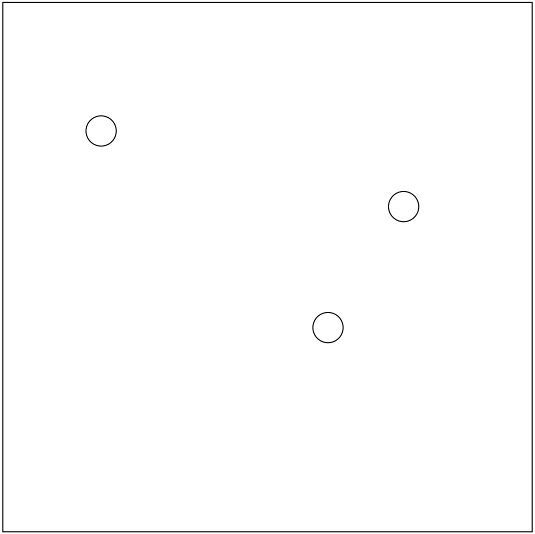
- Long-range calculations can still be important:
    - Electrostatics
    - Interfaces
- Particle-Particle Particle-Mesh (PPPM) approximates long-range forces without requiring pair-wise calculations

PPPM

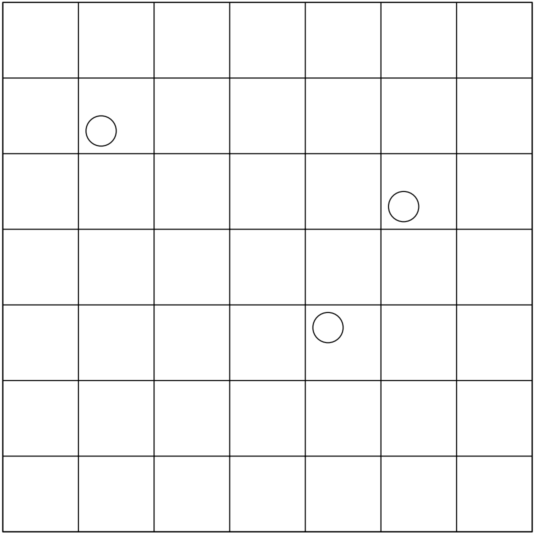
Four Steps:

1. Determine the charge distribution ρ by mapping particle charges to a grid
2. Take the Fourier transform of the charge distribution to find the potential:
$$\nabla^2 \Phi = -\frac{\rho}{\epsilon_0}$$
3. Obtain forces due to *all* interactions by inverse Fourier transform:
$$\vec{F} = -\nabla \Phi$$
4. Map forces back to the particles

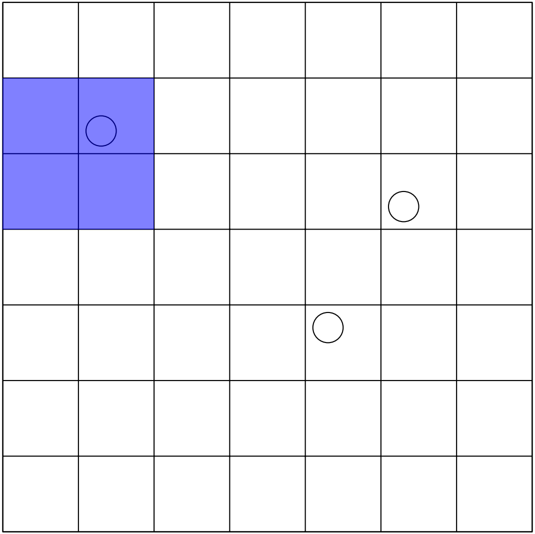
PPPM: Charge Mapping



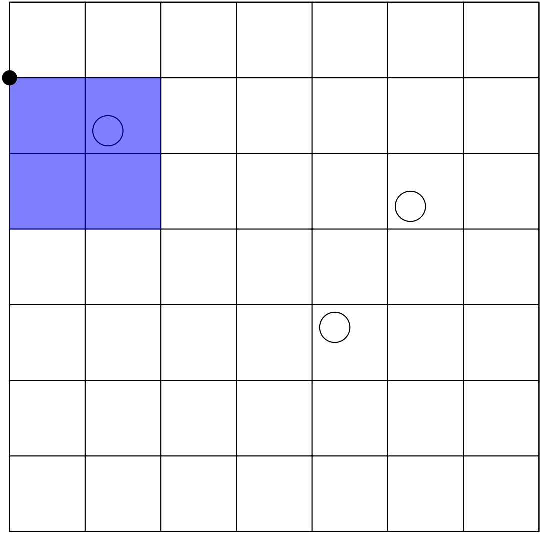
PPPM: Charge Mapping



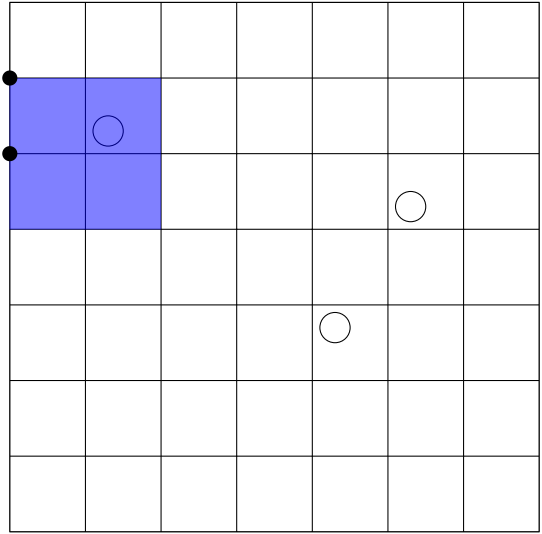
PPPM: Charge Mapping



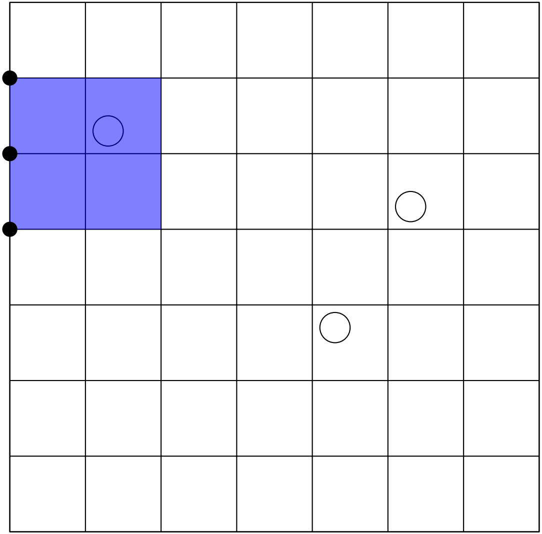
PPPM: Charge Mapping



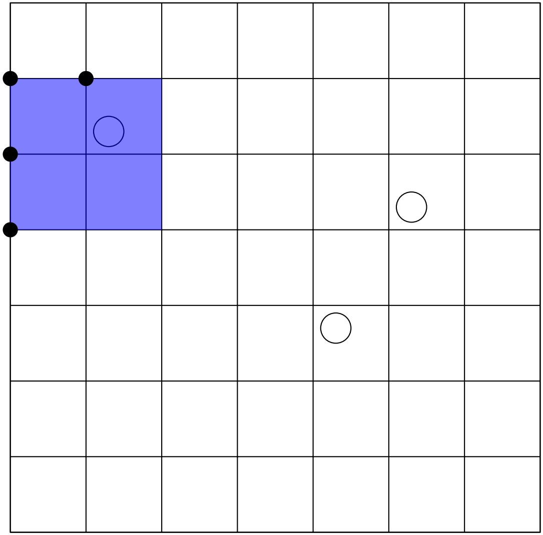
PPPM: Charge Mapping



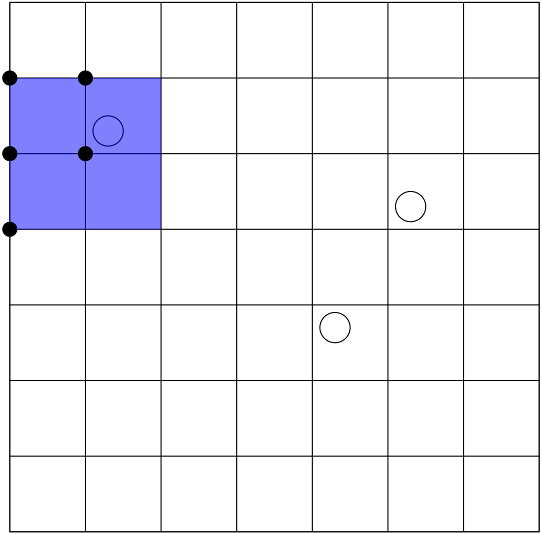
PPPM: Charge Mapping



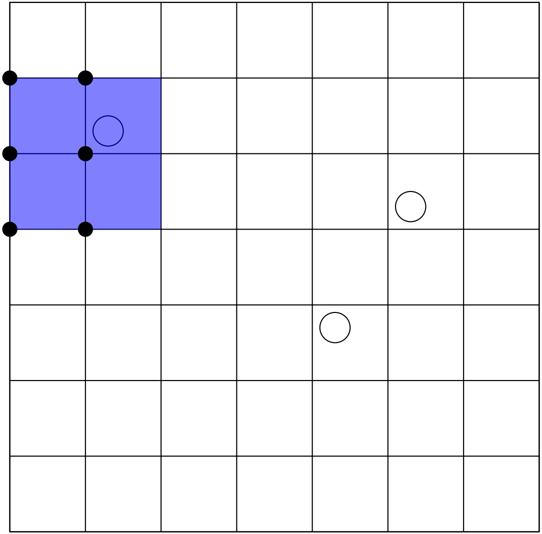
PPPM: Charge Mapping



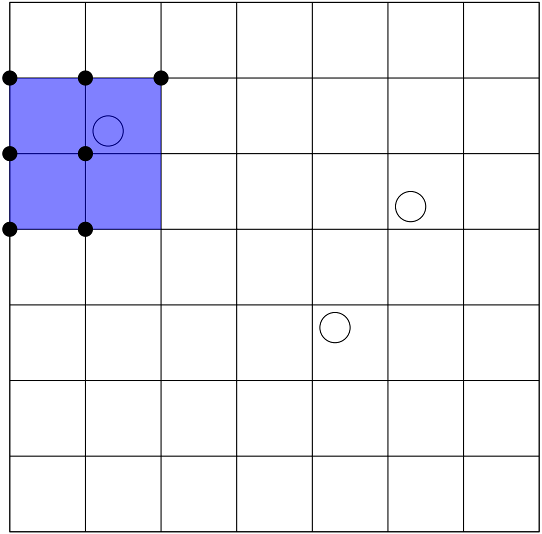
PPPM: Charge Mapping



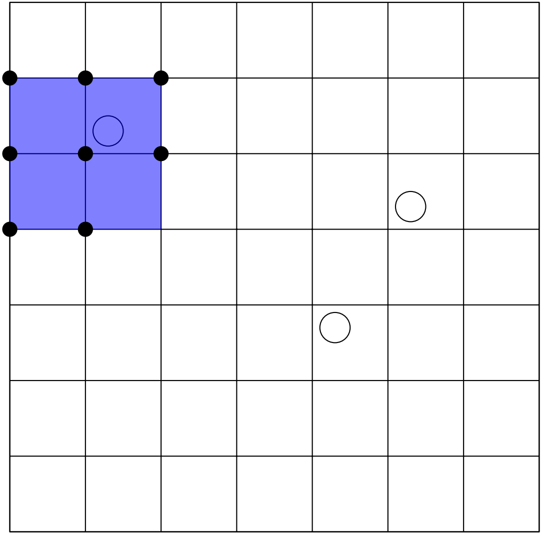
PPPM: Charge Mapping



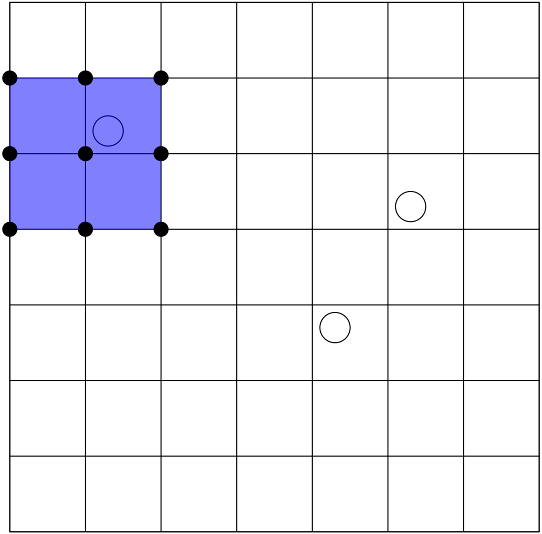
PPPM: Charge Mapping



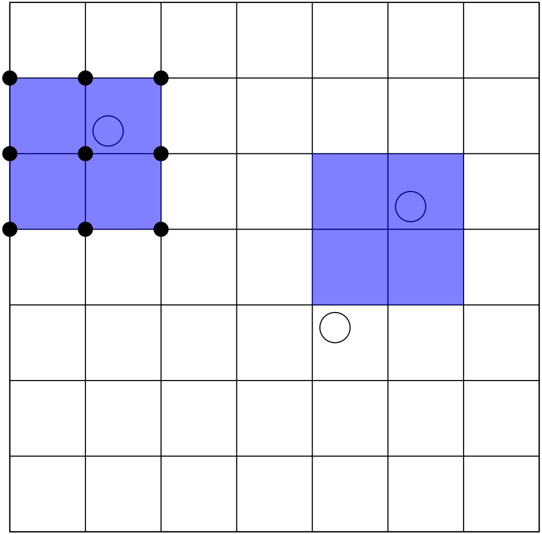
PPPM: Charge Mapping



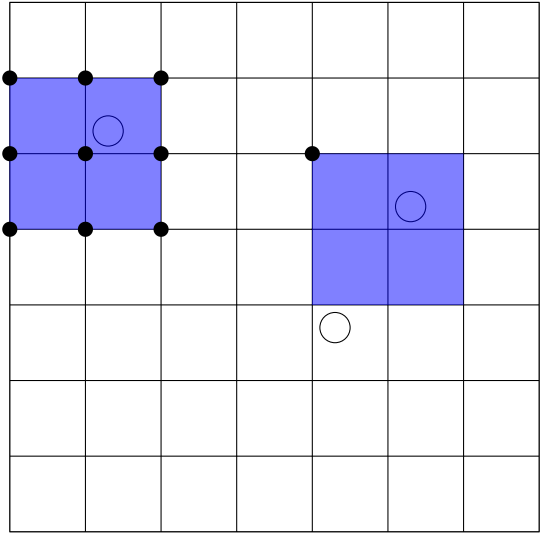
PPPM: Charge Mapping



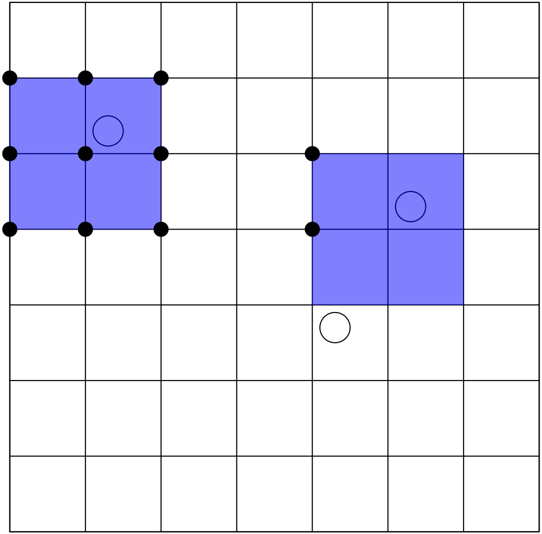
PPPM: Charge Mapping



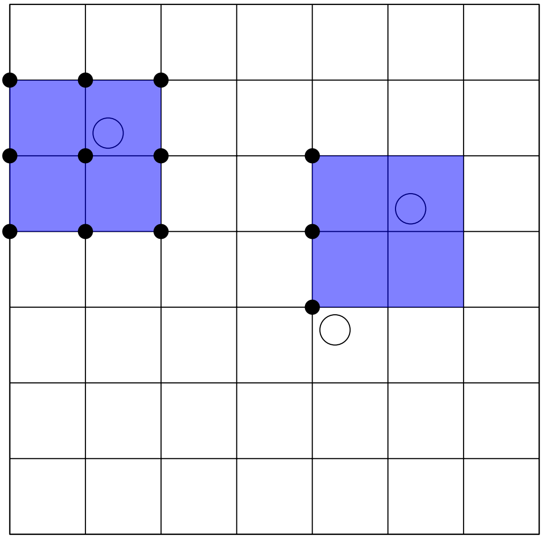
PPPM: Charge Mapping



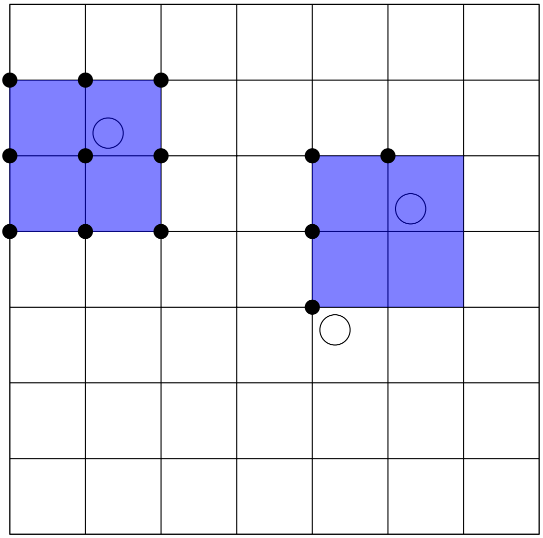
PPPM: Charge Mapping



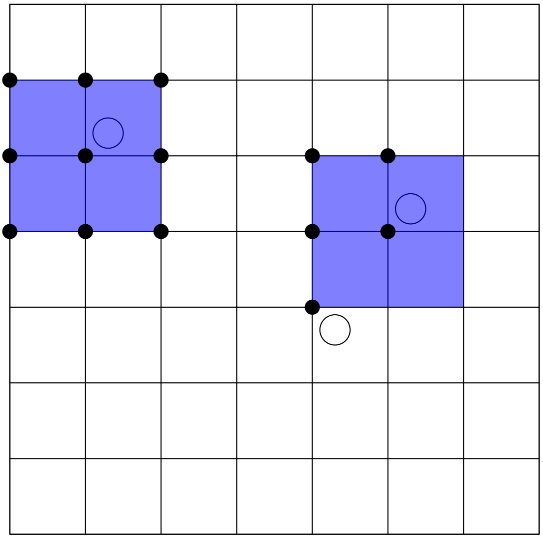
PPPM: Charge Mapping



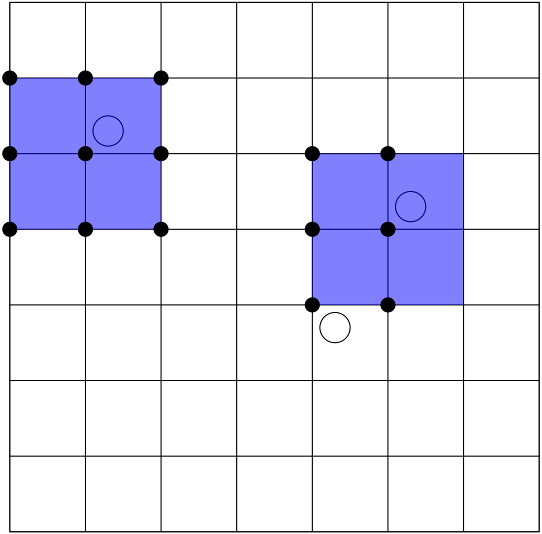
PPPM: Charge Mapping



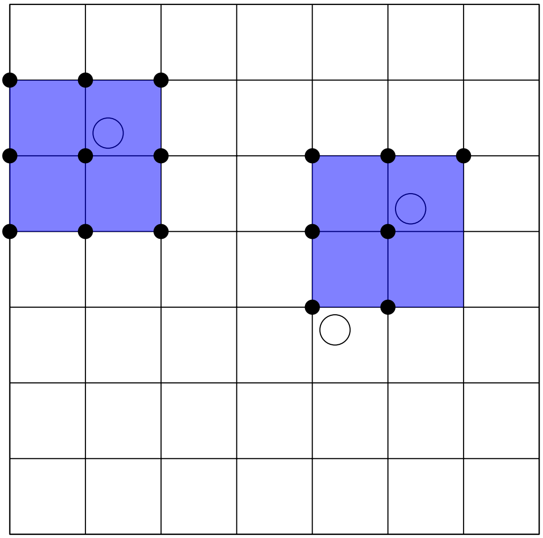
PPPM: Charge Mapping



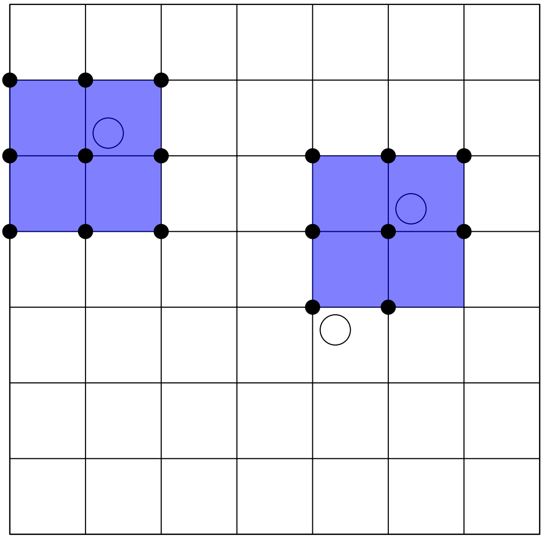
PPPM: Charge Mapping



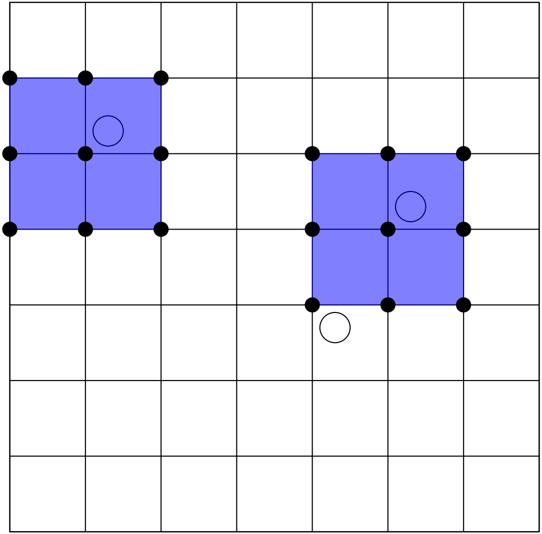
PPPM: Charge Mapping



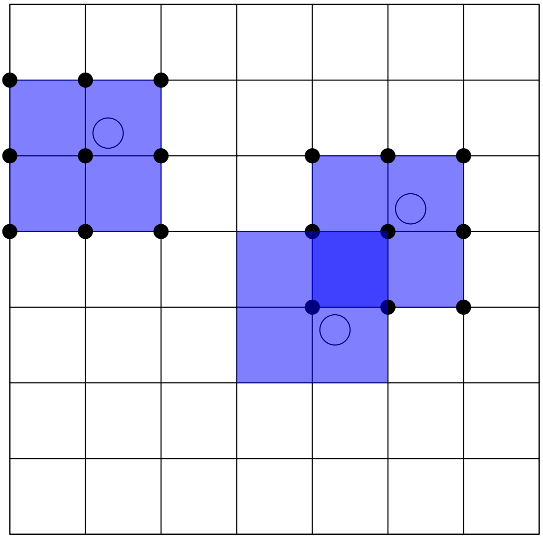
PPPM: Charge Mapping



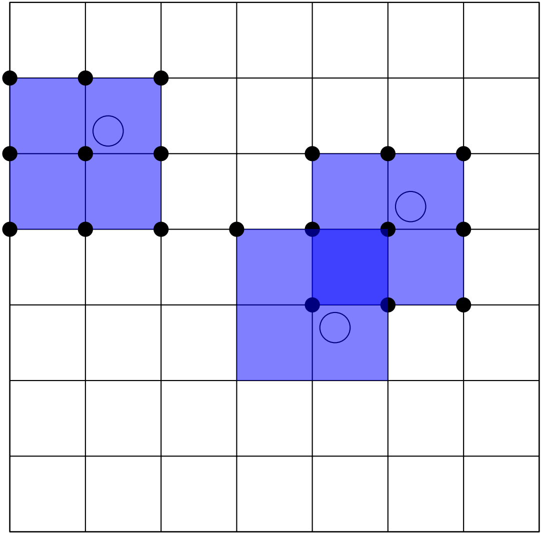
PPPM: Charge Mapping



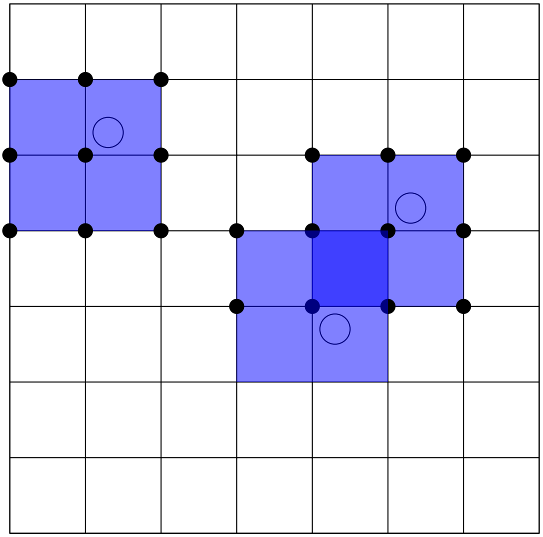
PPPM: Charge Mapping



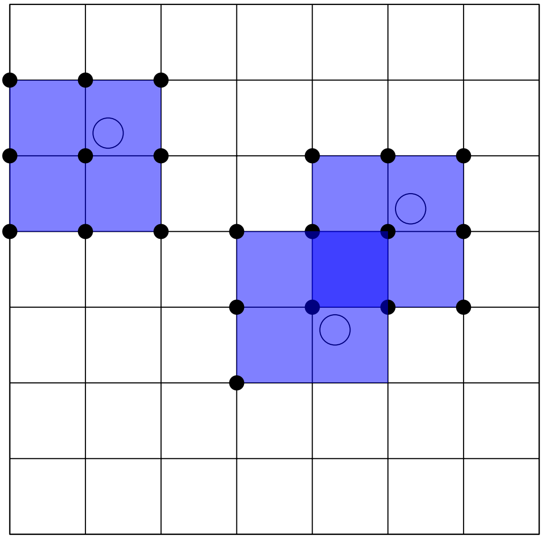
PPPM: Charge Mapping



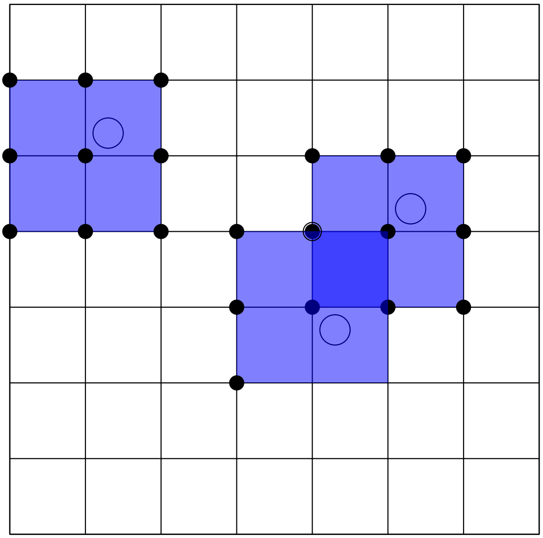
PPPM: Charge Mapping



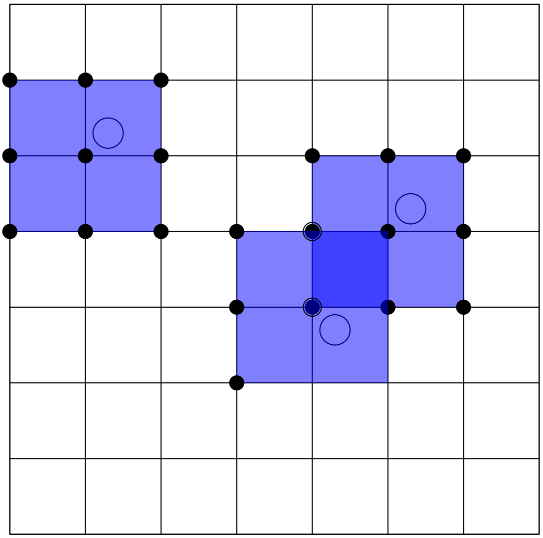
PPPM: Charge Mapping



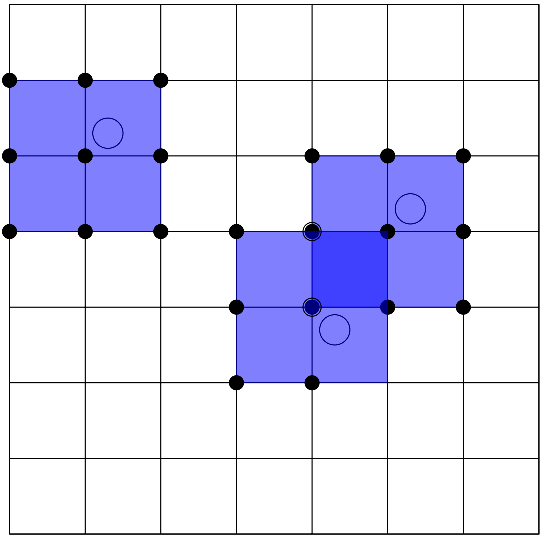
PPPM: Charge Mapping



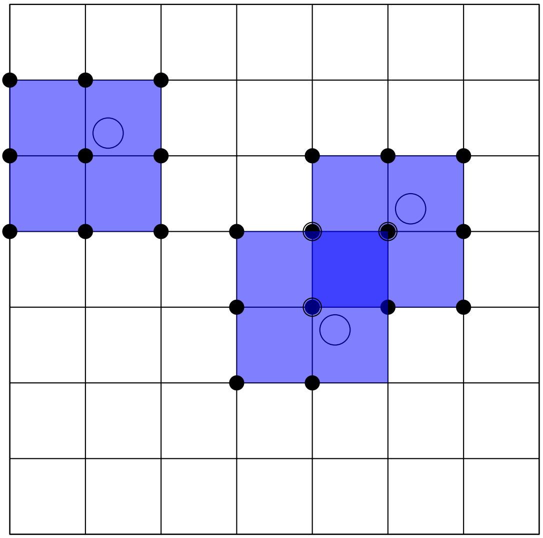
PPPM: Charge Mapping



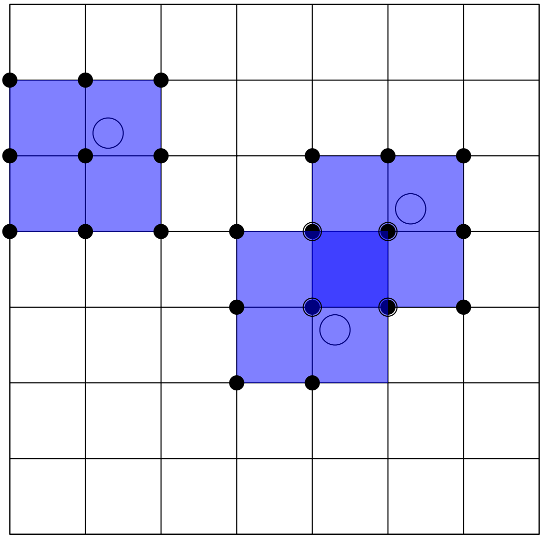
PPPM: Charge Mapping



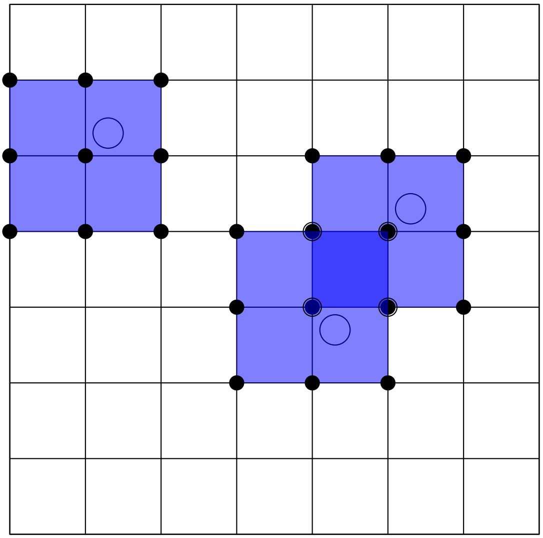
PPPM: Charge Mapping



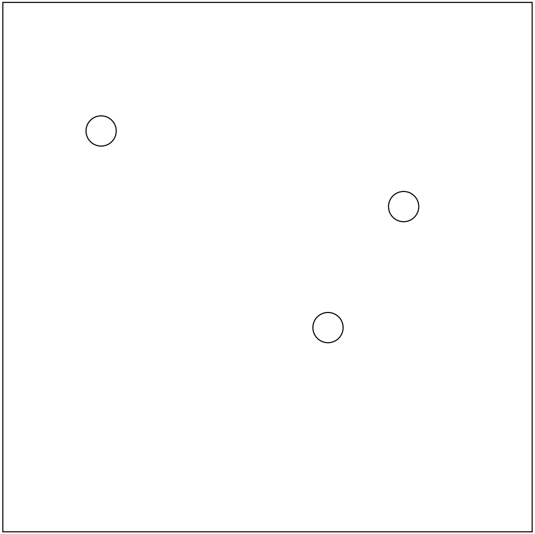
PPPM: Charge Mapping



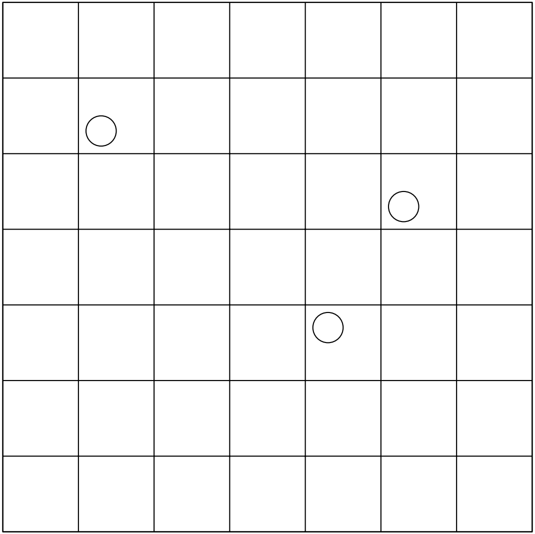
PPPM: Charge Mapping



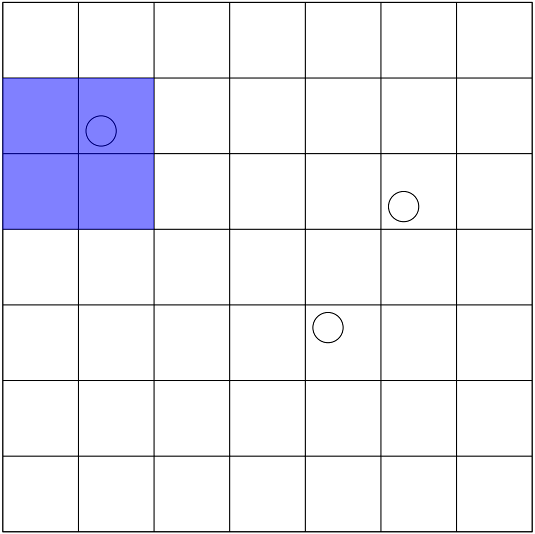
PPPM: Charge Mapping



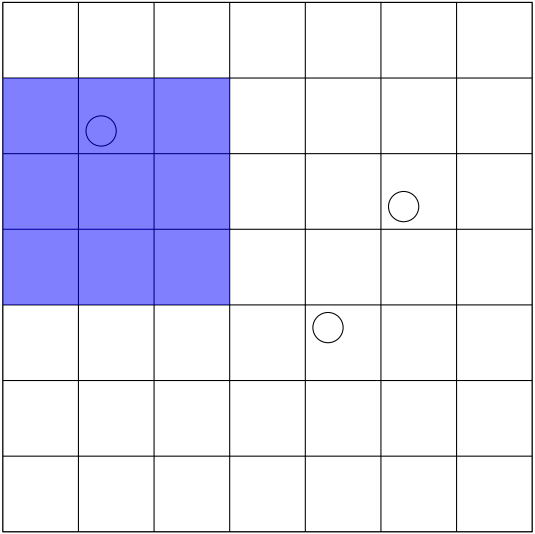
PPPM: Charge Mapping



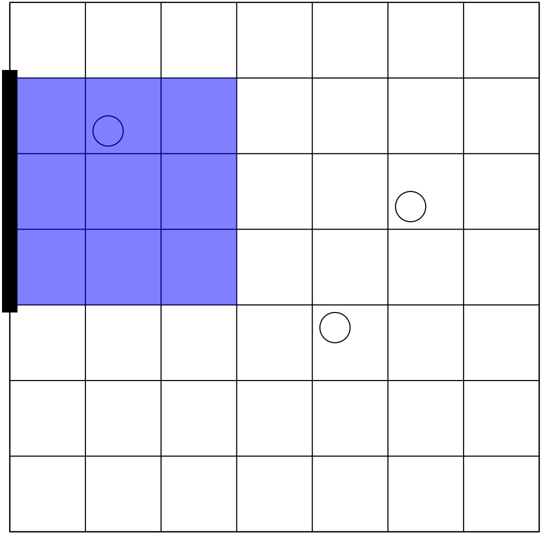
PPPM: Charge Mapping



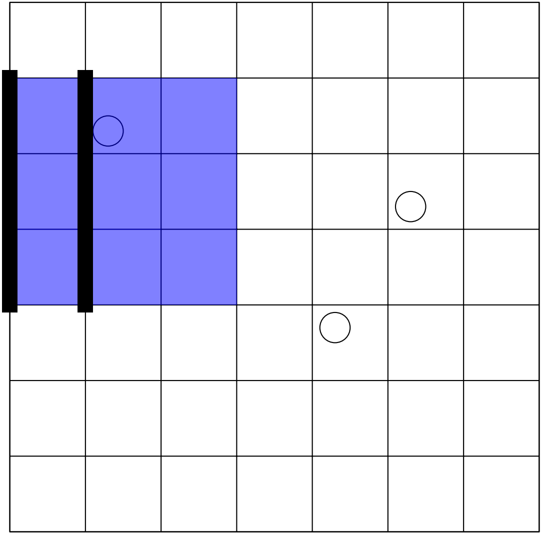
PPPM: Charge Mapping



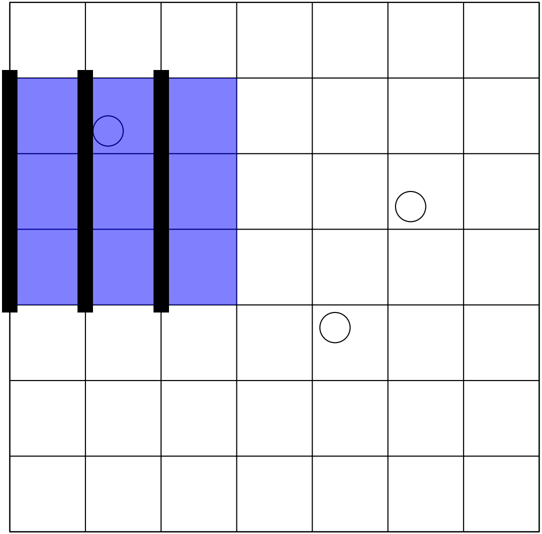
PPPM: Charge Mapping



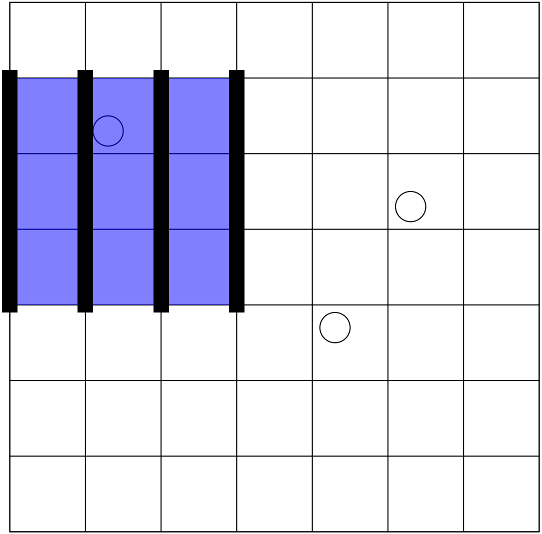
PPPM: Charge Mapping



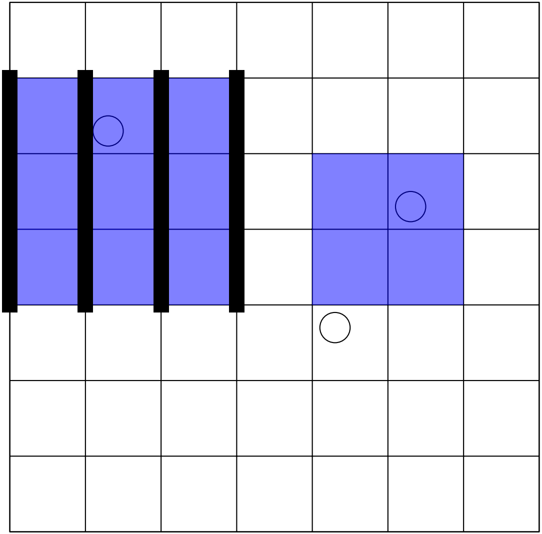
PPPM: Charge Mapping



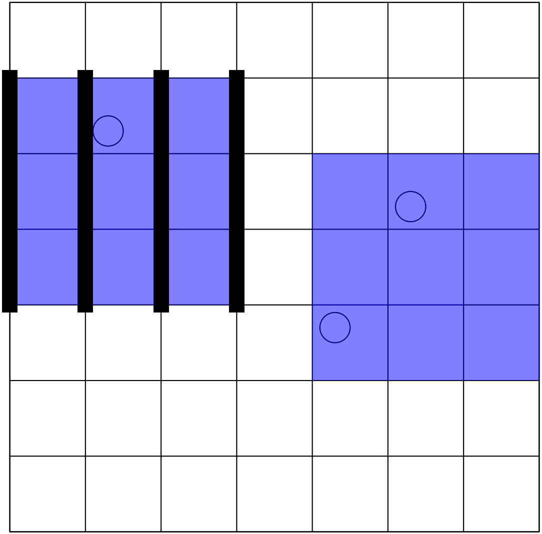
PPPM: Charge Mapping



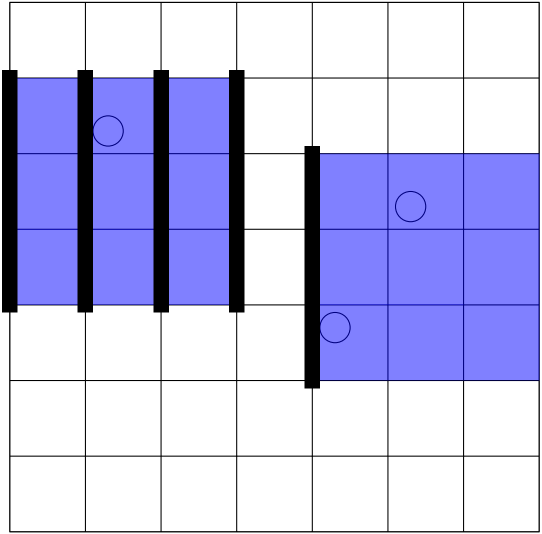
PPPM: Charge Mapping



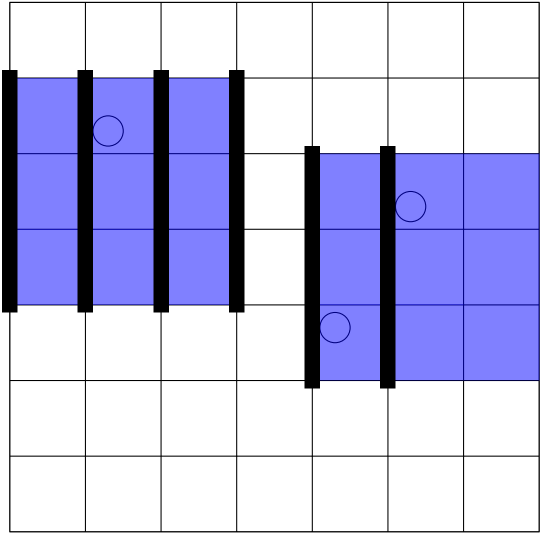
PPPM: Charge Mapping



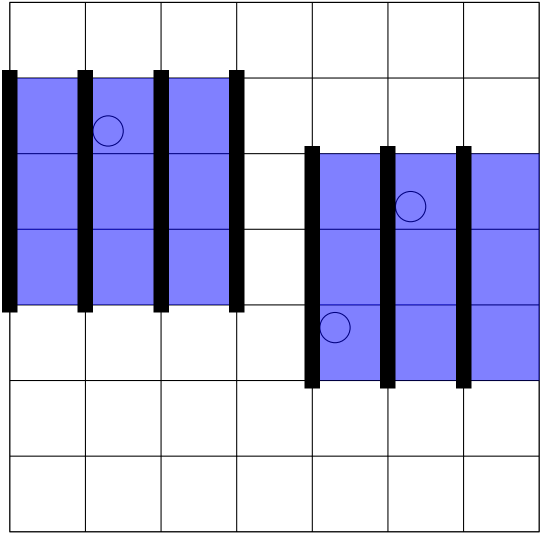
PPPM: Charge Mapping



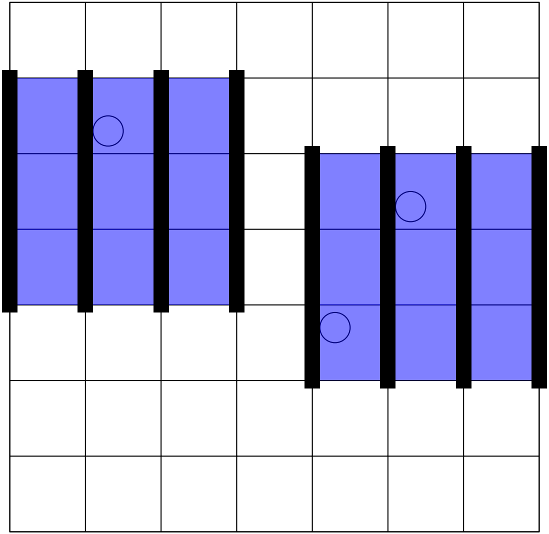
PPPM: Charge Mapping



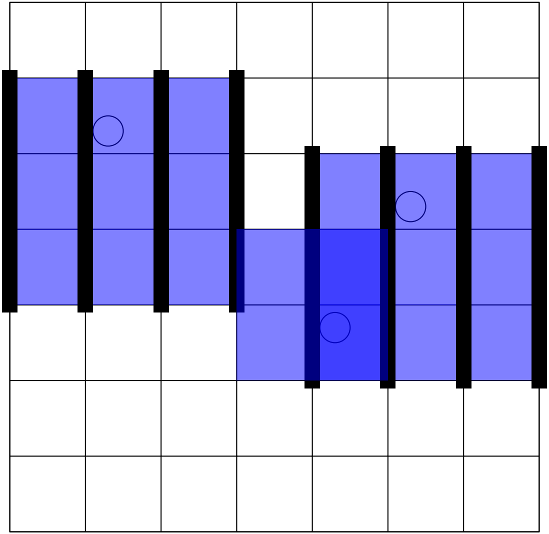
PPPM: Charge Mapping



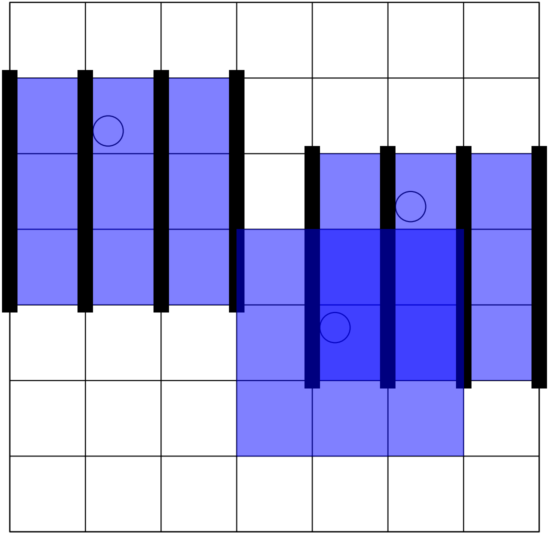
PPPM: Charge Mapping



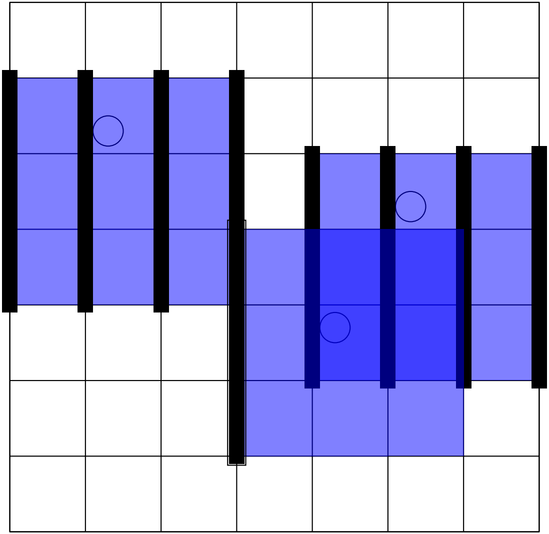
PPPM: Charge Mapping



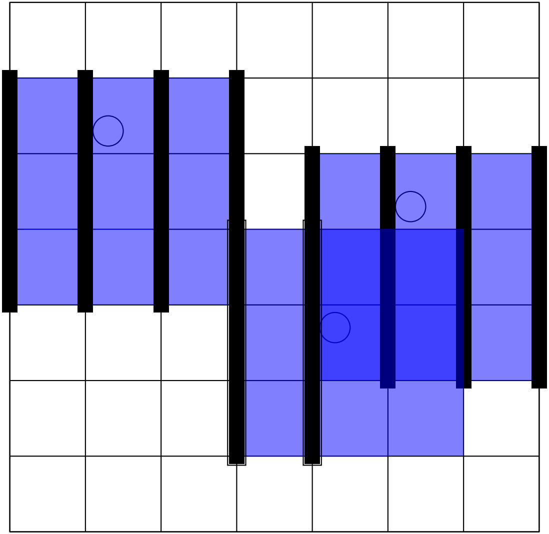
PPPM: Charge Mapping



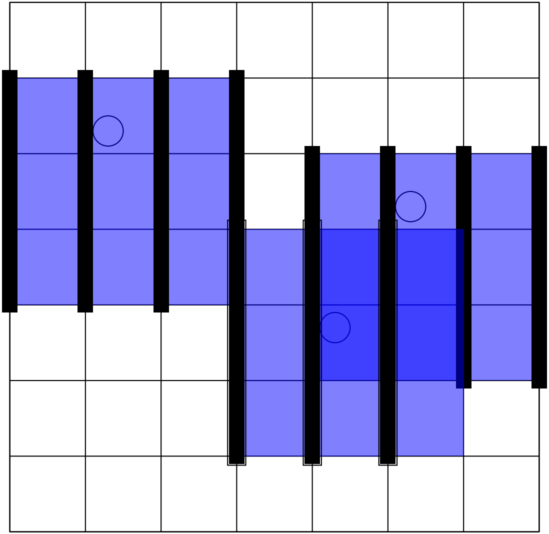
PPPM: Charge Mapping



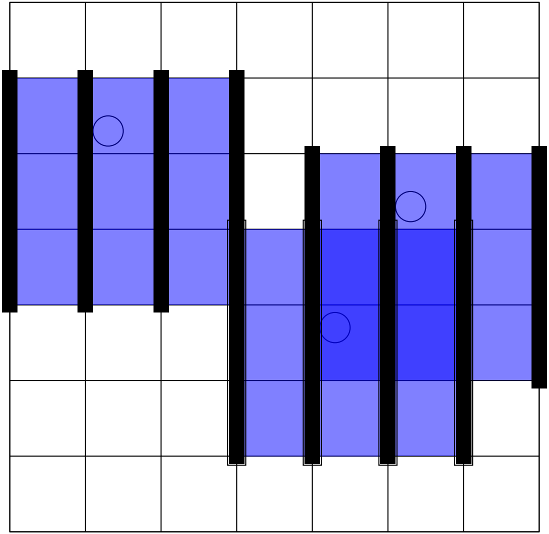
PPPM: Charge Mapping



PPPM: Charge Mapping



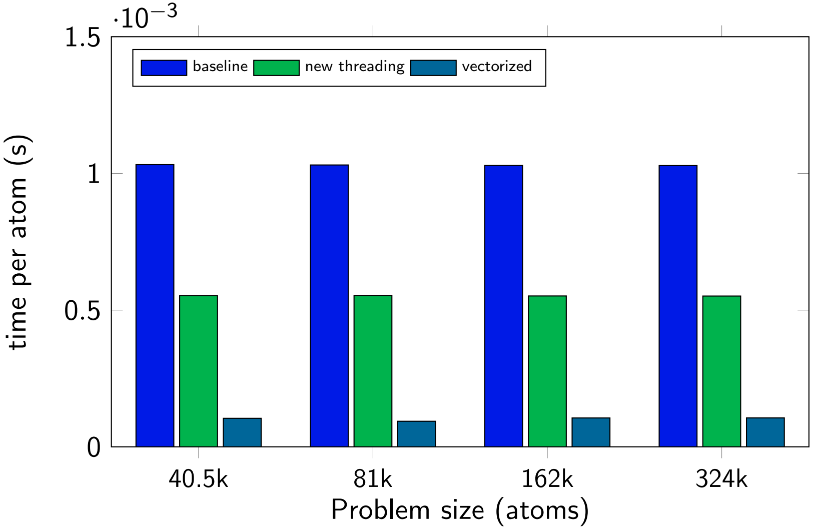
PPPM: Charge Mapping



PPPM: Charge Mapping

- Threading across atoms instead of across grid points
- Look-up table for stencil coefficients
- Vectorized inner stencil loop
- Larger stencil takes advantage of KNL vector length

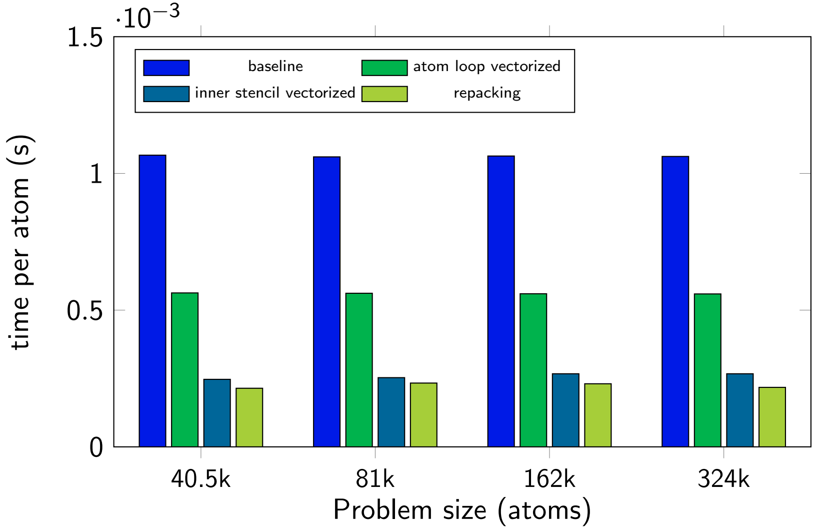
PPPM: Charge Mapping on KNL 1c/1t



PPPM: Distributing Forces

- Look-up table for stencil coefficients
- Vectorized inner stencil loop
- Larger stencil takes advantage of KNL vector length
- Repack force data to get multiple components simultaneously

PPPM: Distributing Forces on KNL 1c/1t



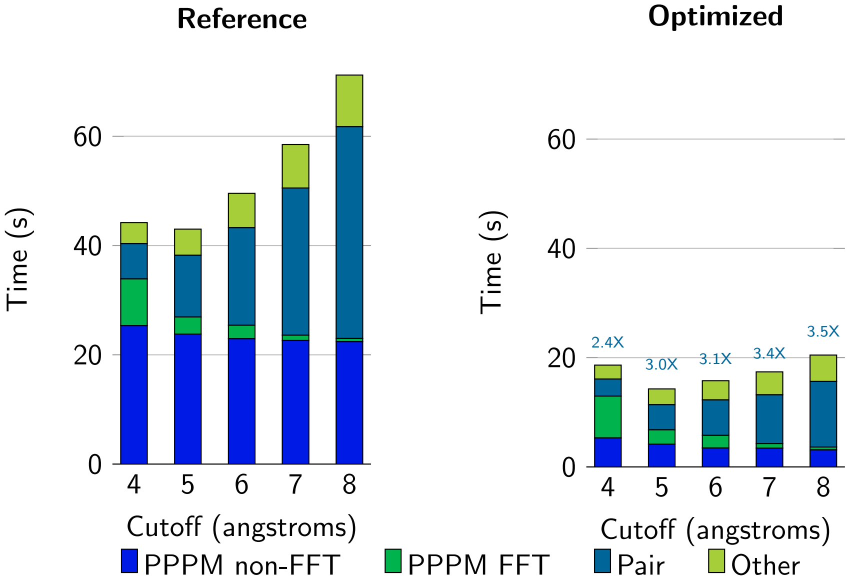
PPPM: FFTs

- Larger stencil allows coarser grid while preserving accuracy

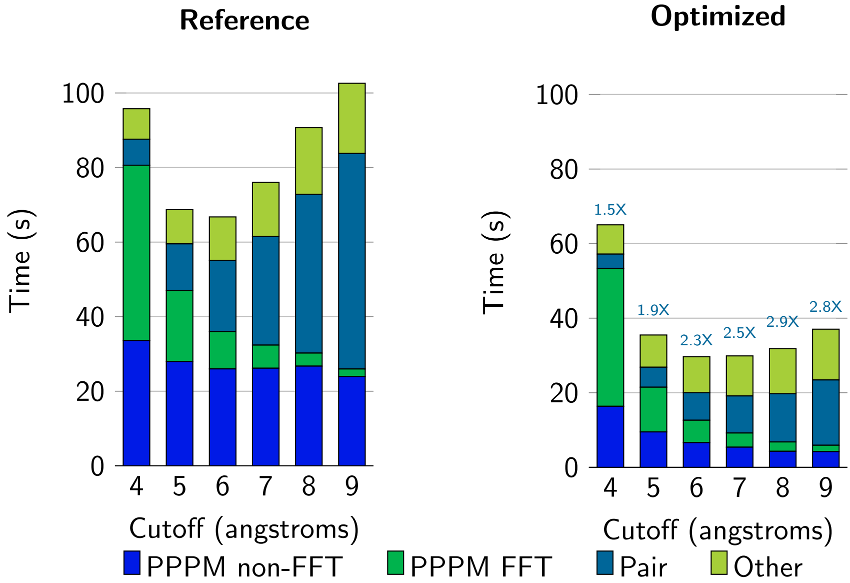
- Reduce communication by doing:
  $2D \rightarrow \textit{remap} \rightarrow 1D \rightarrow \textit{remap}$
  instead of:
  $1D \rightarrow \textit{remap} \rightarrow 1D \rightarrow \textit{remap} \rightarrow 1D \rightarrow \textit{remap}$

- Vectorization elsewhere makes *ad* differentiation relatively more appealing than *ik* differentiation – half as many FFTs in exchange for more work in stencil loops

Water Benchmark – KNL 1c/1t



Water Benchmark – KNL 64c/1t



PPPM Dispersion

Similar particle mapping concept, but with two potentials:

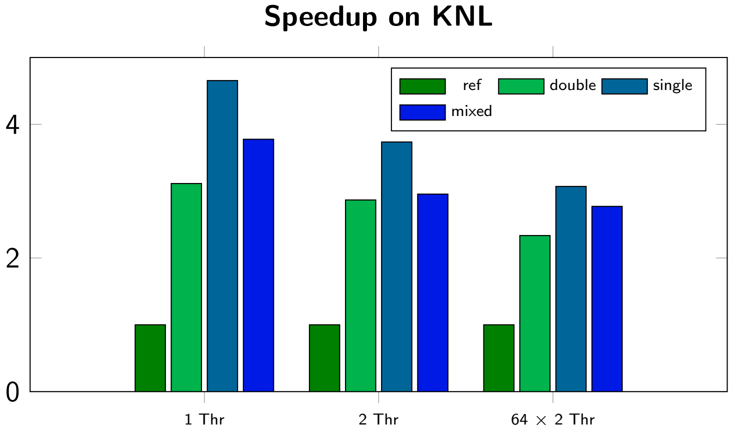
- Electrostatics $\sim \frac{1}{r}$
- Dispersion interactions $\sim \frac{1}{r^6}$

- Optimize compatible pair potentials (Ready)
  - Buckingham (buck/long/coul/long)
  - Lennard Jones (lj/long/coul/long)

- Optimize PPPM-dispersion solver (In Progress)

Buckingham – Dispersion

- SiO_2 model, 19200 atoms - coulomb and buck potentials
- KNL Cache mode
- Reference : USER_OMP



PPPM Dispersion: Components

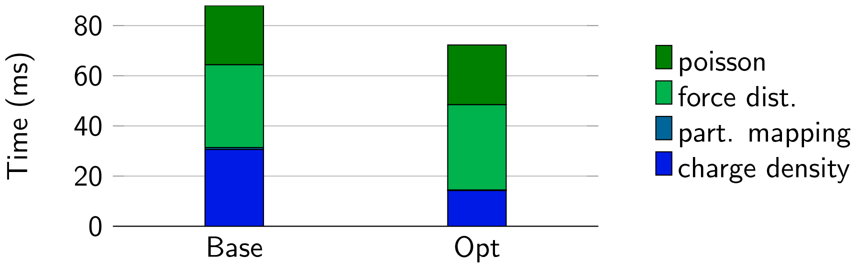
Having multiple types of forces requires different mixing rules:

- Equivalent routines operate on different stencils
  - 2 versions of particle mapping
  - 4 versions of charge density
  - 12 versions of force distribution & poisson solver
- We use templates, optimizing only once
  - Minimize control structures

PPPM Dispersion: Results

- Optimized charge & particle mapping
  double precision + single precision FFTs
  - Between 1.4X and 1.6X speedup on K-space

- Potential speedups for poisson & force



Code Availability

Code	Github[1]	LAMMPS
Tersoff	✓	✓
Buckingham	✓	✓
Buckingham Coul Long	✓	✓
Buckingham Long Coul Long	✓	. . .
Lennard-Jones Long Coul Long	✓	. . .
PPPM	. . .	. . .
PPPM Dispersion	✓	. . .
REBO	. . .	. . .
AIREBO	. . .	. . .

[1] Our group's repositories are at `github.com/HPAC`.

Dissemination and Community Involvement

- SIAM CSE 2017, Atlanta: MD Exascale Mini-Symposium
  - Bientinesi (Aachen), McDoniel (Aachen), Tchipev (München)
- ISC'17 Paper
- SC'16 Technical Program Talk
- IPCC Meeting Toulouse Talk
- Paper for IXPUG Workshop @ ISC'16: "Dynamic SIMD Lane Scheduling"
  - Krzikalla (Dresden), Wende (Berlin), Höhnerbach (Aachen)
- ISC'16 Booth and IPCC Meeting Talks
- Parallel'16 Talk
  - Krzikalla (Dresden), Höhnerbach (Aachen)
- IPCC Meeting Ostrava Talk
- SC'15 Workshop Talk
- IPCC Meeting München Code Dungeon

Other activities & future work

Other research activities

- Tensors operations
  - Tensor transposition, summations, contractions
  - Applications from Chemistry and Machine Learning
  - Collaboration with IPCC UT Austin
- BLAS
  - Idea: CPU + stream to Phi
  - MKL – limited functionality
  - Application in Density Functional Theory
  - Initial results: 1610 vs 1350 GFLOPS/s (MKL)

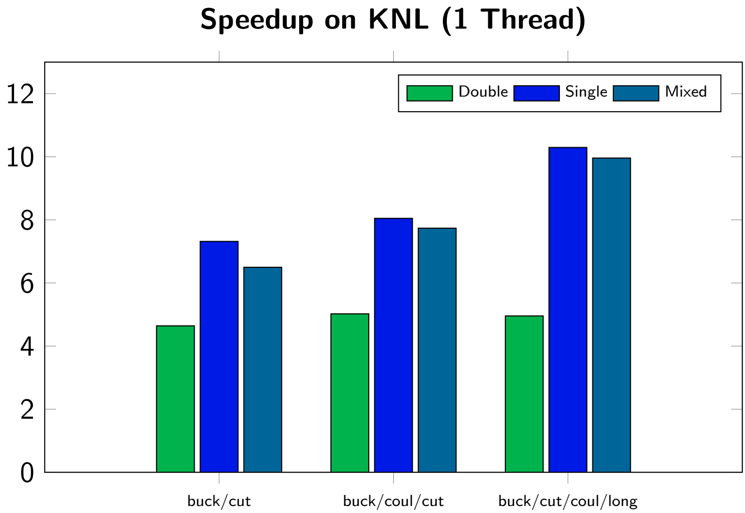
LAMMPS

- Continue collaboration with Mike Brown
- Additional Long-Ranged Solvers
  - Multi-Level Summation (MSM): $\mathcal{O}(n)$ algorithm
    - 2/3rd of routines similar to PPPM (particle to grid and back)
    - 1/3rd: Stencil application (Research topic)
    - MSM Dispersion solver developed by our group
  - Gaussian split Ewald: Mesh-based real/frequency space
    - Might provide better accuracy than MSM
    - First implementation into LAMMPS
- Extend KOKKOS to enable vector classes (avoid GPU bias)

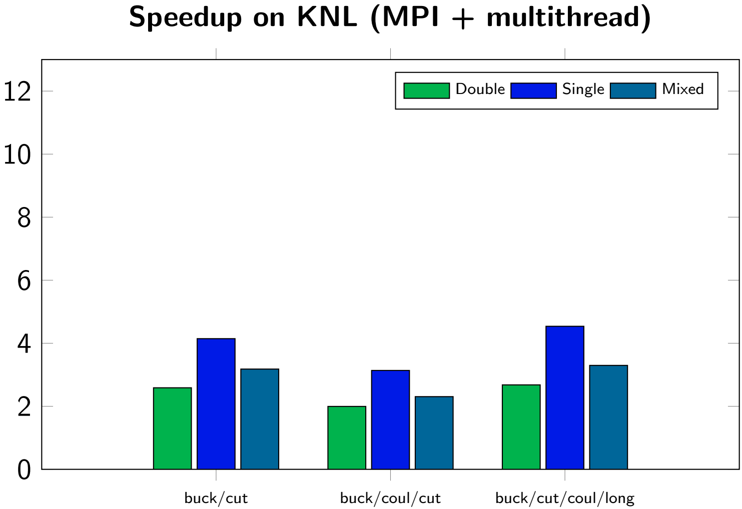
DSMC

- Particle-based method for rarefied gas
- Similar to molecular dynamics and LAMMPS

Buckingham: vectorization, single thread

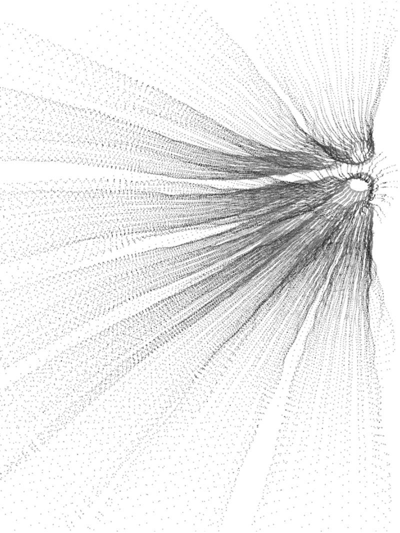


Buckingham: vectorization, full node



Part 2: Tersoff multibody potential

Markus Höhnerbach



```
for $i$ in local atoms of the current thread do
    for $j$ in atoms neighboring $i$ do
        $\zeta_{ij} \leftarrow 0$;
        for $k$ in atoms neighboring $i$ do
            $\zeta_{ij} \leftarrow \zeta_{ij} + \zeta(i,j,k)$;
        $E \leftarrow E + V(i,j,\zeta_{ij})$;
        $F_i \leftarrow F_i - \partial_{x_i} V(i,j,\zeta_{ij})$;
        $F_j \leftarrow F_j - \partial_{x_j} V(i,j,\zeta_{ij})$;
        $\delta\zeta \leftarrow \partial_{\zeta} V(i,j,\zeta_{ij})$;
        for $k$ in atoms neighboring $i$ do
            $F_i \leftarrow F_i - \delta\zeta \cdot \partial_{x_i}\zeta(i,j,k)$;
            $F_j \leftarrow F_j - \delta\zeta \cdot \partial_{x_j}\zeta(i,j,k)$;
            $F_k \leftarrow F_k - \delta\zeta \cdot \partial_{x_k}\zeta(i,j,k)$;
```