Multilevel Summation Method for Calculating Electrostatic Interactions in NAMD

David J. Hardy

Theoretical and Computational Biophysics Group
Beckman Institute for Advanced Science and Technology
University of Illinois at Urbana-Champaign
http://www.ks.uiuc.edu/~dhardy/

15th Annual Workshop on Charm++ and its Applications
April 17-19, 2017

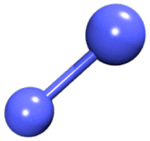
NIH Center for Macromolecular Modeling and Bioinformatics
http://www.ks.uiuc.edu/

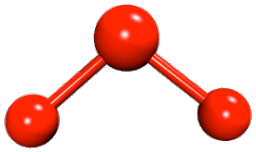
Molecular Dynamics

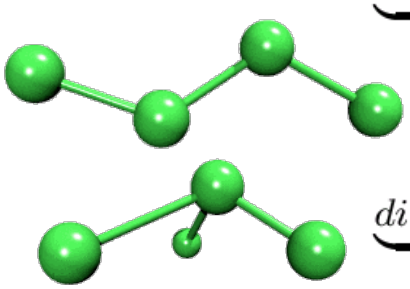
Integrate Newton’s equations of motion:

$$m_i \frac{d^2}{dt^2}\vec{r}_i(t) = -\nabla_i U(\vec{R})$$

for billions of time steps!







$$
\begin{aligned}
U(\vec{R}) \quad = \quad & \underbrace{\sum_{bonds} k_i^{bond}(r_i - r_0)^2}_{U_{bond}} + \underbrace{\sum_{angles} k_i^{angle}(\theta_i - \theta_0)^2}_{U_{angle}} + \\
& \underbrace{\sum_{dihedrals} k_i^{dihe}[1 + \cos(n_i\phi_i + \delta_i)]}_{U_{dihedral}} + \\
& \underbrace{\sum_i \sum_{j \neq i} 4\epsilon_{ij}\left[\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}}\right)^{6}\right] + \boxed{\sum_i \sum_{j \neq i} \frac{q_i q_j}{\epsilon r_{ij}}}}_{U_{nonbond}}
\end{aligned}
$$

Coulomb potential

NIH Center for Macromolecular Modeling and Bioinformatics
http://www.ks.uiuc.edu/

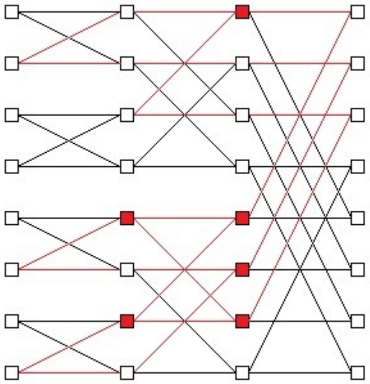
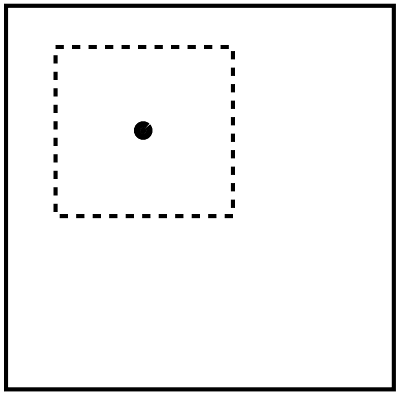
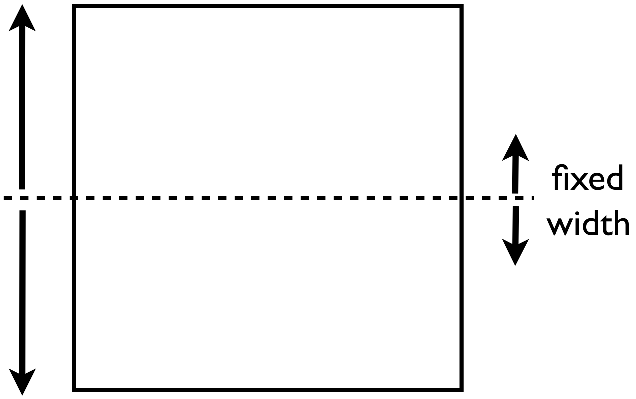
Motivation for multilevel summation method (MSM)

- Need to accurately represent electrostatic interactions - long-range, requires fast method
- Usually done using PME (particle-mesh Ewald)
- PME has two shortcomings
    - requires periodic boundary conditions
    - poses bottleneck to parallel scalability
- MSM overcomes both shortcomings!

Best features of MSM

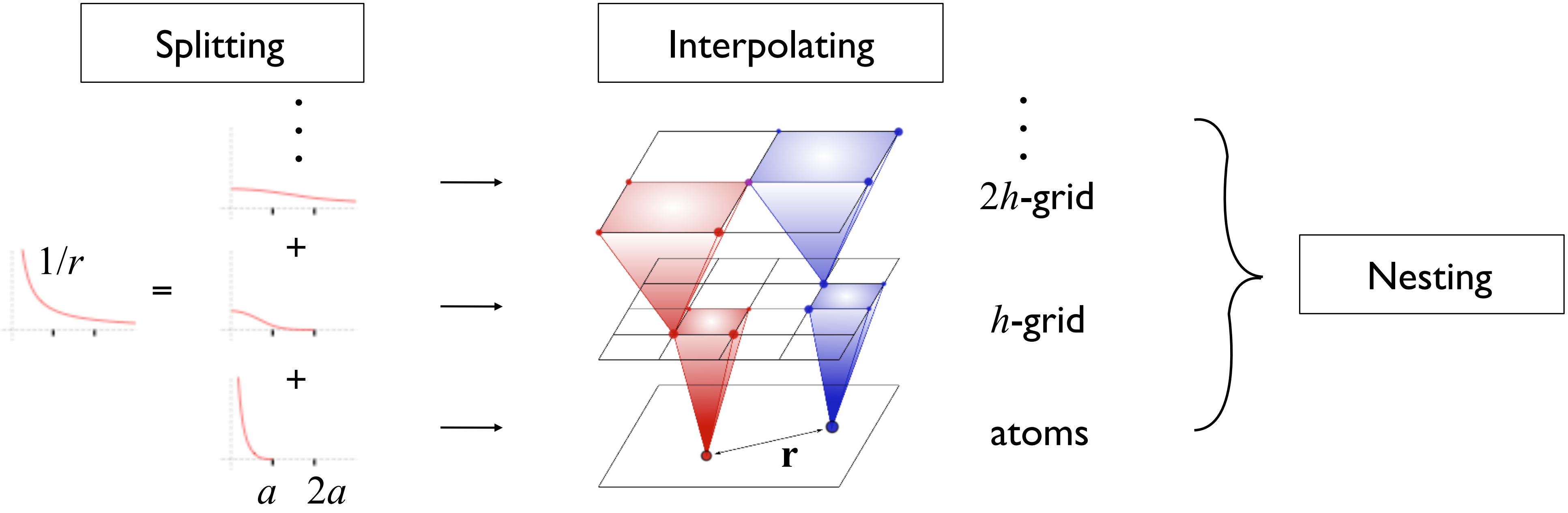
- Supports periodic boundaries and **also supports**:
    - non-periodic boundaries (e.g. protein folding in water droplet)
    - semi-periodic boundaries (e.g. membrane channel)
- Offers better parallel scaling through hierarchical structure (does not need FFT)
- Arithmetic intensity and localized memory access well suited to modern hardware (CPU vector instructions and GPUs)
- Produces smooth forces for stable dynamics
- Extends to other pairwise interactions (e.g. dispersion)
- Algorithm has linear time complexity

Comparing MSM with PME

	PME	MSM
Memory Access	scattered across grid  (depicting FFT in 1D)	highly localized  (depicting convolution in 2D)
Parallel Communication	many-to-many (matrix transpose)	tree-like (reduction and expansion)
Bisection Bandwidth on 3D torus (Blue Waters)	$\mathcal{O}\left(N/P^{2/3}\right)$  	$\mathcal{O}\left((N/P)^{2/3}\right)$

MSM essential ideas

- Splitting the interaction kernel
- Interpolating the slowly varying kernels from grids
- Nesting the approximation between levels



Splitting

Interpolating

Nesting

cutoff distances a, $2a$, $4a$, ...

grid spacings h, $2h$, $4h$, ...

Splitting the interaction kernel (i)

$$|\mathbf{r}' - \mathbf{r}|^{-1} = k_0(\mathbf{r}, \mathbf{r}') + k_1(\mathbf{r}, \mathbf{r}') + \cdots + k_L(\mathbf{r}, \mathbf{r}')$$

In one dimension, unparameterized, in terms of function γ :

$$\frac{1}{\rho} = \gamma_0(\rho) + \frac{1}{2}\gamma_1(\frac{1}{2}\rho) + \cdots + \frac{1}{2^L}\gamma_L(\frac{1}{2^L}\rho)$$

$$\begin{aligned} \gamma_0(\rho) &= (1/\rho) - \gamma(\rho), \\ \gamma_l(\rho) &= 2\gamma(2\rho) - \gamma(\rho), \quad l = 1, 2, \ldots, L-1, \\ \gamma_L(\rho) &= 2\gamma(2\rho) \end{aligned}$$

$k_l(\mathbf{r}, \mathbf{r}') = \dfrac{1}{2^l a}\gamma_l(\dfrac{r}{2^l a})$ parameterized by cutoff value a

Splitting the interaction kernel (ii)

For interpolation with degree $p-1$ piecewise polynomials we want splitting with C^{p-1} continuity:

$$\gamma(\rho) = \begin{cases} \tau_p(\rho^2), & \text{for } 0 \le \rho \le 1, \\ 1/\rho, & \text{for } \rho \ge 1 \end{cases}$$

$$\begin{aligned} s^{-1/2} &= 1 - \frac{1}{2}(s-1) + \frac{3}{8}(s-1)^2 - \frac{5}{16}(s-1)^3 + \cdots \\ &= \tau_p(s) + O((s-1)^p) \end{aligned}$$

Optimal in the sense that it minimizes $\int_0^1 \Big(\frac{\mathrm{d}^p}{\mathrm{d}\rho^p}\gamma(\rho)\Big)^2 \mathrm{d}\rho$ for $\gamma(\rho)$

Interpolating kernels on grids

$$\mathcal{I}_l\, k_l(\mathbf{r},\mathbf{r}') = \sum_m \sum_n \phi_m^l(\mathbf{r})\, k_l(\mathbf{r}_m^l,\mathbf{r}_n^l)\, \phi_n^l(\mathbf{r}'), \quad l = 1, 2, \ldots, L$$

where $\mathcal{I}$ is interpolation operator and

$$\phi_m^l(\mathbf{r}) = \Phi\Big(\frac{x - x_m^l}{2^{l-1}h}\Big)\Phi\Big(\frac{y - y_m^l}{2^{l-1}h}\Big)\Phi\Big(\frac{z - z_m^l}{2^{l-1}h}\Big)$$

Φ is piecewise polynomial of degree $p-1$ with stencil size p and h is the finest grid spacing

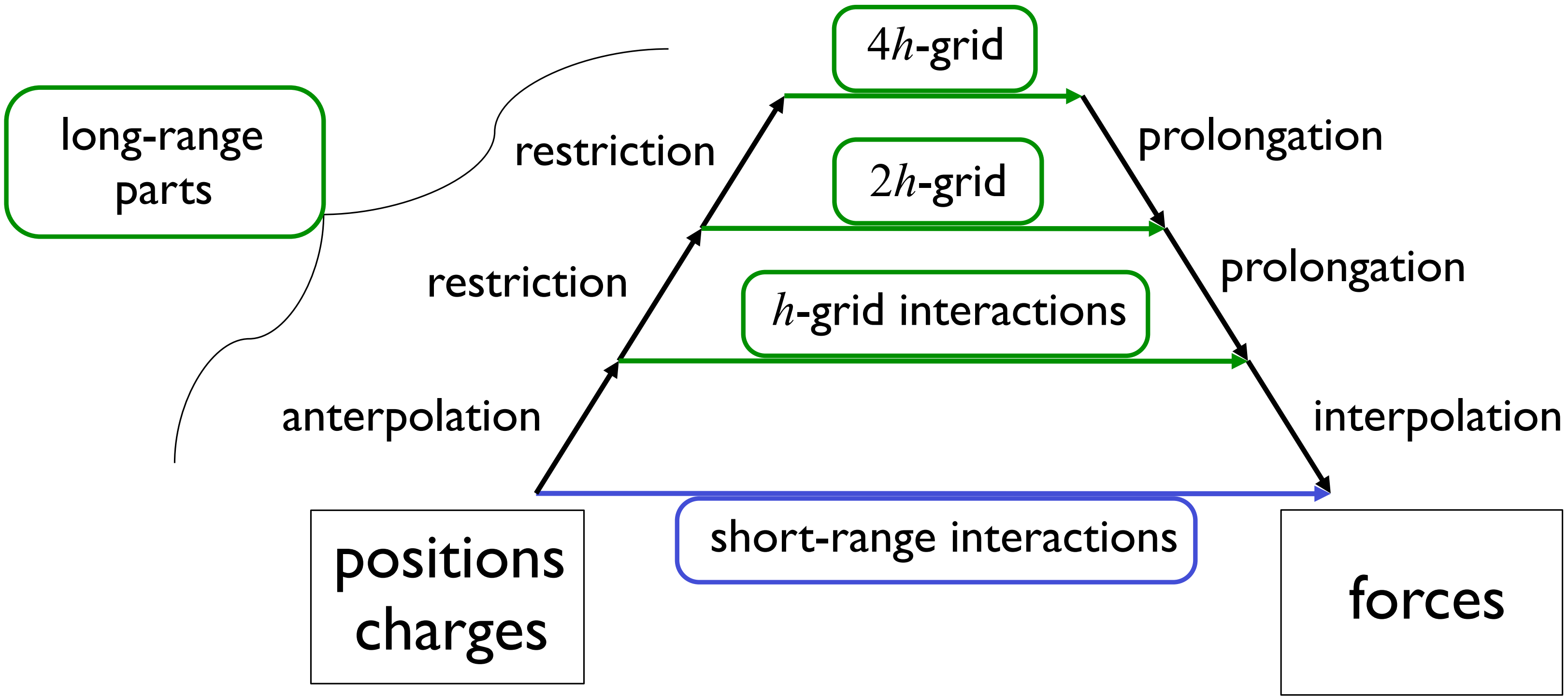
Nesting the approximation between grid levels:

$$k(\mathbf{r},\mathbf{r}') \approx \Bigg(k_0 + \mathcal{I}_1\Big(k_1 + \mathcal{I}_2\big(k_2 + \cdots \mathcal{I}_{L-1}(k_{L-1} + \mathcal{I}_L k_L)\cdots\big)\Big)\Bigg)(\mathbf{r},\mathbf{r}')$$

MSM computation

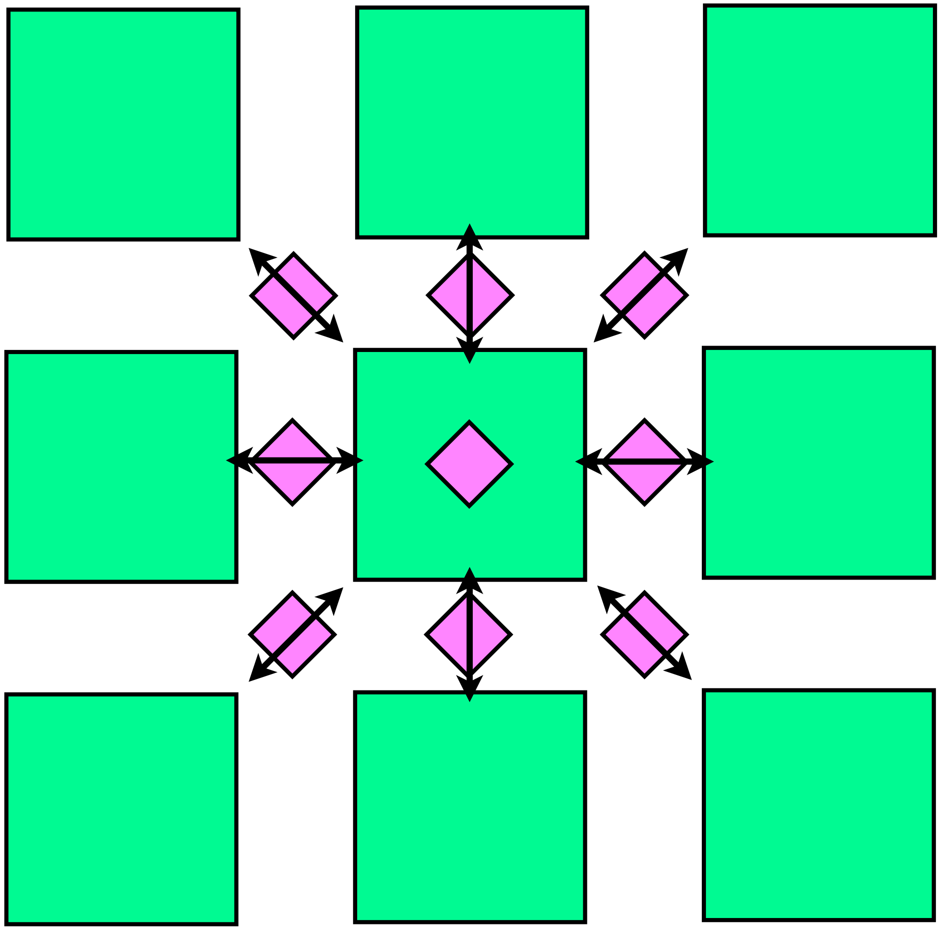
force = **exact short-range part** + **interpolated long-range part**

Computational Steps



NAMD hybrid decomposition for short-range

Kale, *et al.*, *J. Comp. Phys.* **151**:283-312, 1999

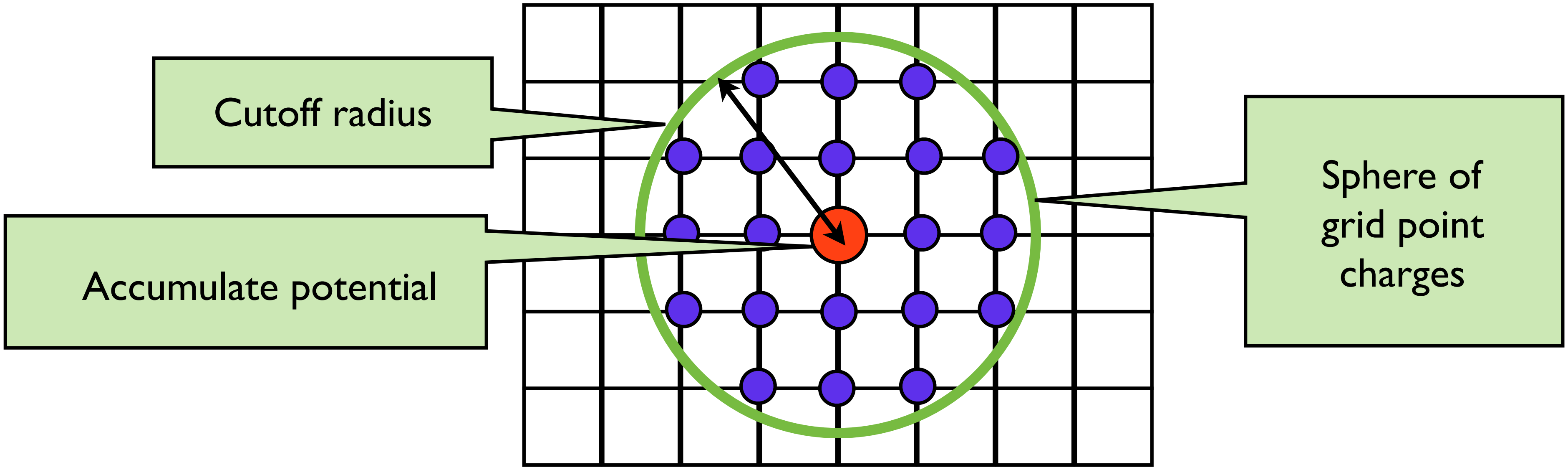


- Decompose atoms spatially into *patches*
- Decompose work into concurrent *compute objects*
- Compute objects facilitate iterative, measurement-based load balancing

MSM Grid Interactions

- Potential summed from grid point charges within cutoff
- Uniform spacing enables distance-based interactions to be precomputed as stencil of "weights"
- Weights at each level are identical up to scaling factor (!)
- Calculate grid potential as 3D convolution of weights with charges

$$e_m^l = \sum_n k_l(\mathbf{r}_m^l, \mathbf{r}_n^l) q_n^l, \quad l = 1, 2, \ldots, L$$



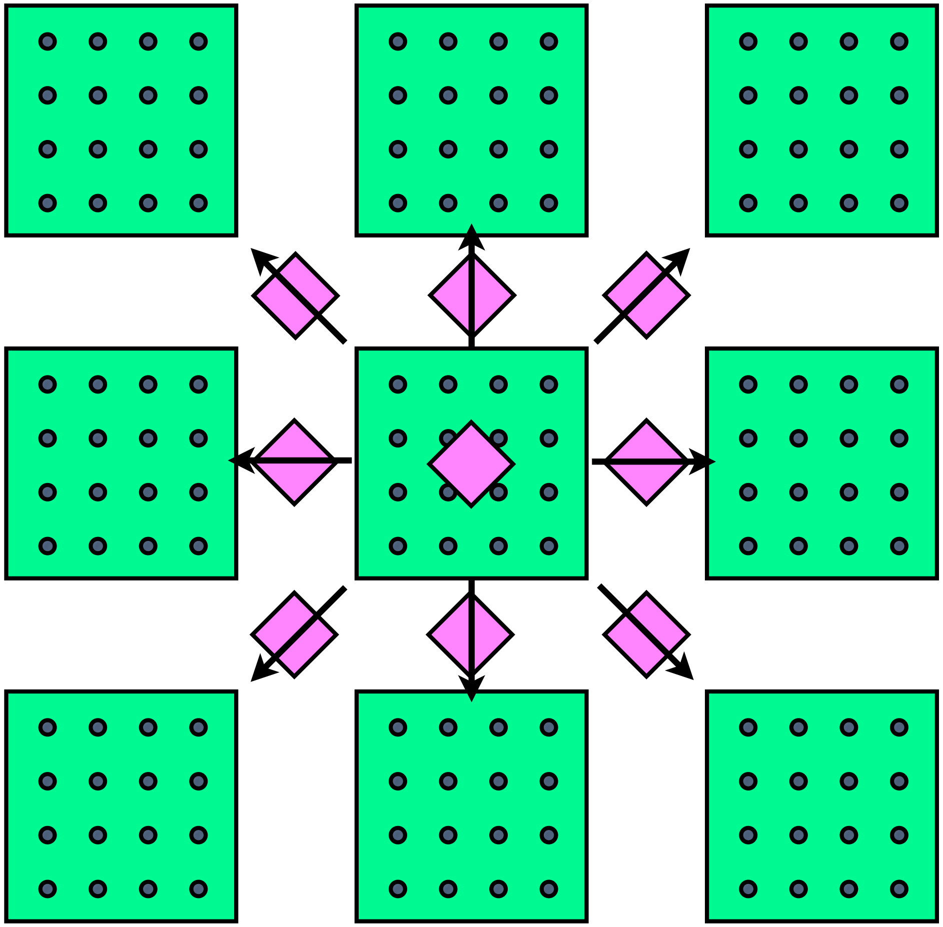
Cutoff radius

Accumulate potential

Sphere of grid point charges

MSM decomposition for grid interactions

Hybrid spatial-work decomposition, similar to short-range



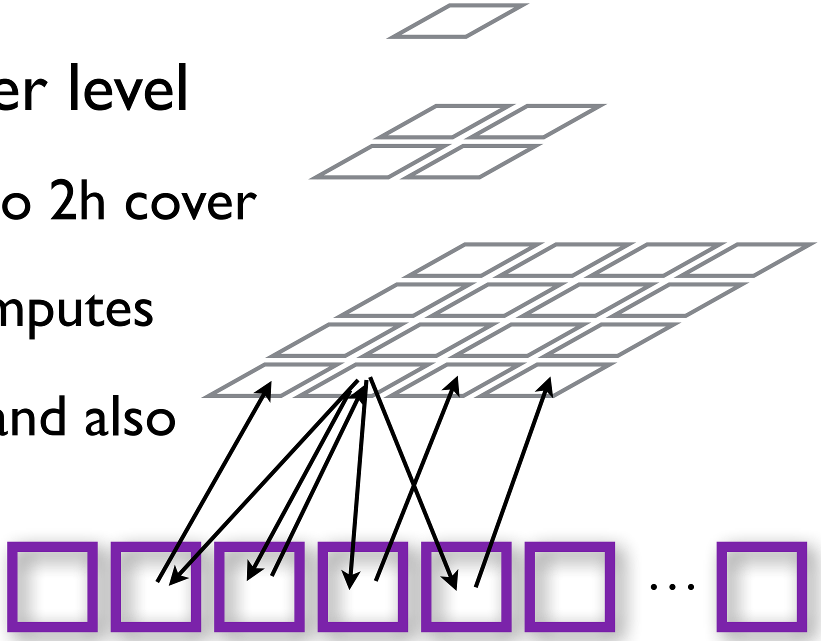
- Grids of charge and potential are decomposed into *blocks*

- Interactions between blocks are separately scheduled as *block computes*

- Need only charges to calculate potentials, send in one direction

MSM use of Charm++

- 3D chare arrays of grid blocks, one per level
  - Performs restriction and prolongation to 2h cover
  - Sends charges up and then to block computes
  - Receives partial potentials from above and also from block computes
- 1D chare array of block computes
- Associate an object with each NAMD patch to perform anterpolation and interpolation



Some Charm++ coding paradigms

part of an MSM block

```
class MsmBlock {
  public:

    void add_charge_from_below(const Grid<float>& qh) {
      my_qh += qh;  // qh is a subgrid of my_qh
      if (++cnt_recv_charge == max_recv_charge) {
        compute_restriction();  // calculate my_q2h_cover from my_qh
        send_charge_up();       // send my_q2h_cover
        send_charge_across();   // send my_qh
      }
    }

};
```

Most compelling use I've ever seen for multiple inheritance in scientific computing!

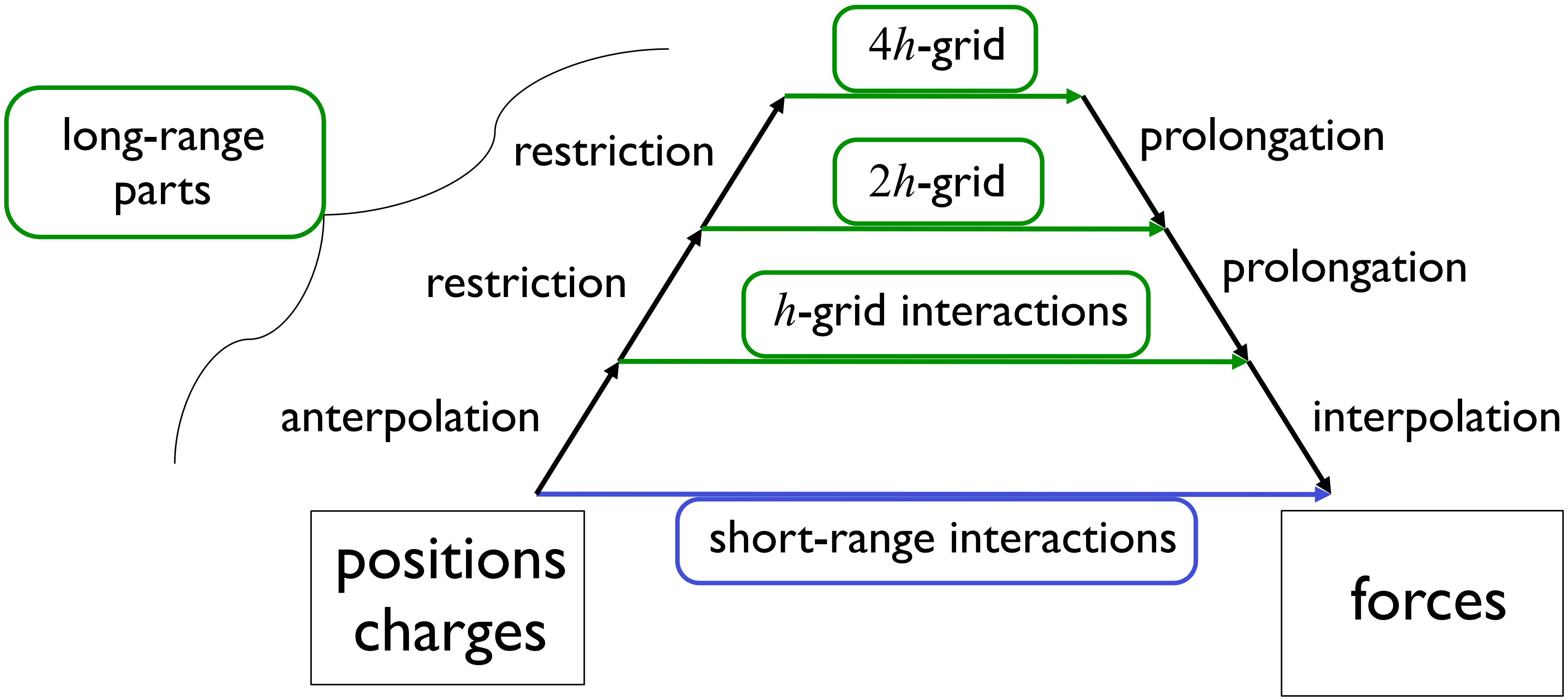
```
class MsmBlockChare :
  public MsmBlock,
  public CBase_MsmBlockChare {

    // communication wrapper for MsmBlock

};
```

Static load balancing

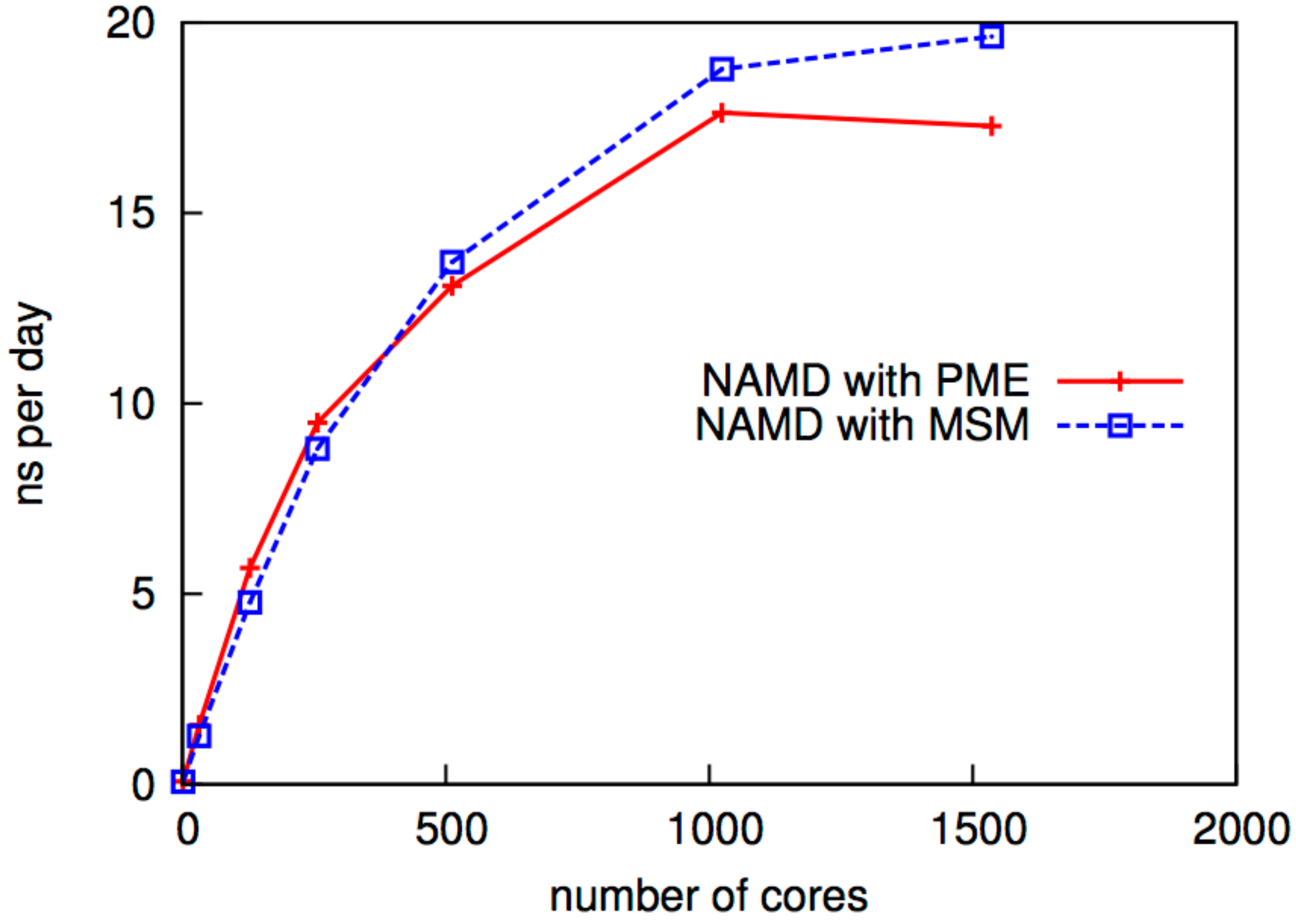
- Distribute grid blocks evenly among nodes
- Assign block computes to sender or receiver node (trying to minimize inter-node communication)
- Each node distributes the block compute objects evenly among cores

Optimizing the critical path

- Highest message priority assigned to restrictions going up the hierarchy, then block computes and prolongations going from the top down



MSM scaling results



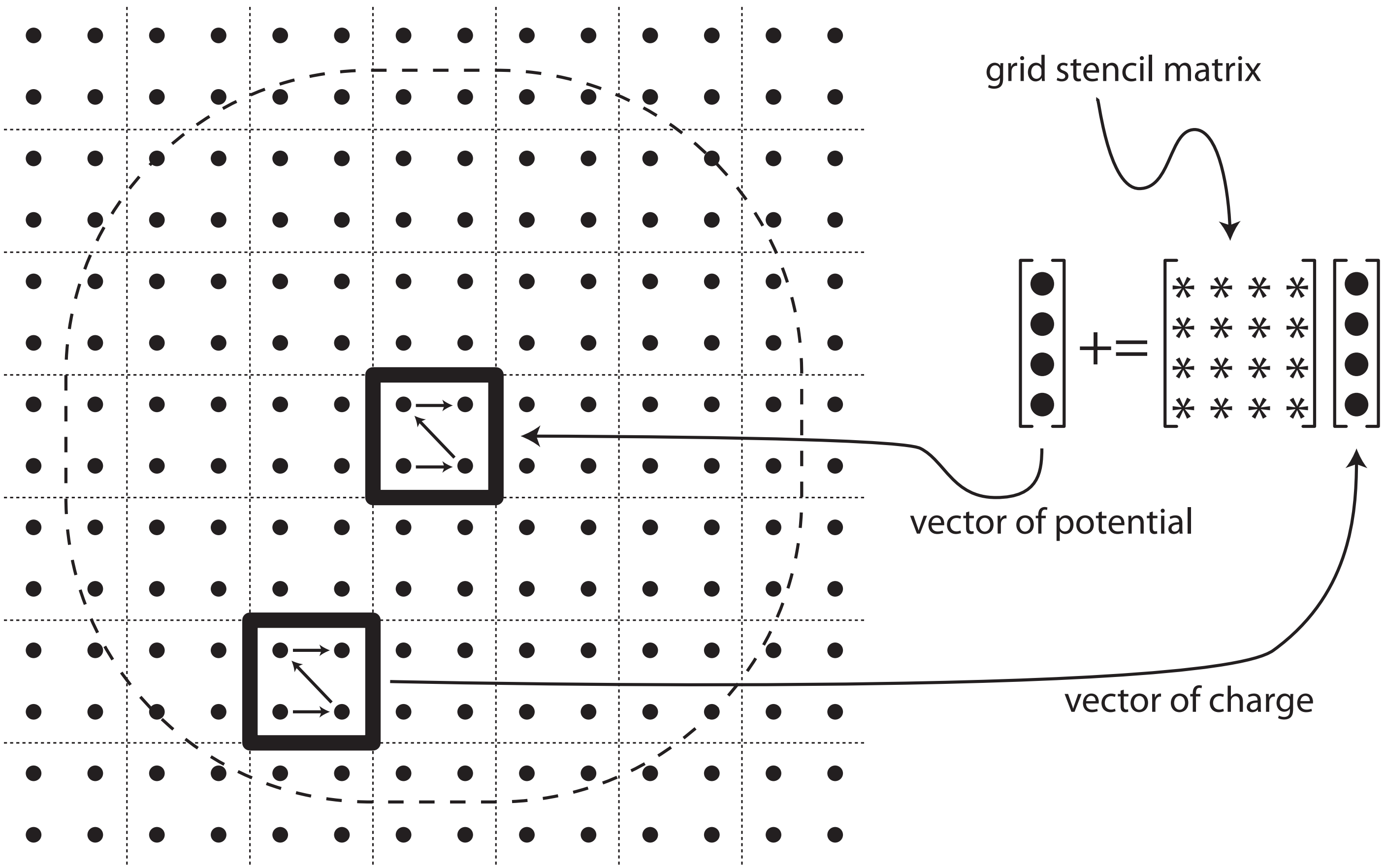
Strong scaling
~92K-atom ApoA1
on Cray XE6
Blue Waters
hardware

Hardy, *et al.*, *J. Chem. Theory Comput.* 11:766-779, 2015

Recent MSM advances

- B-spline interpolation
  - improves accuracy by an order of magnitude for the same computational effort
  - caveat: more expensive to calculate stencils
- CPU vectorization
  - improves single core performance
  - caveat: requires extensive data reorganization

Clustering grid points



Enables use of CPU vector instructions (AVX/FMA)

Cluster into 8-point cubes single precision

Shows about 7x improvement over non-vector version

B-spline interpolation

- Basis set for splines
- Interpolation with p-1 degree splines gives pth order accuracy
- Smallest possible local support of p
- Continuity is C(p-2)
- B-splines provide *nested interpolation*: a coarse level B-spline is exactly represented by finer level B-splines

However, the B-splines are not nodal basis functions for interpolation!

We want the interpolant in the form

$$\tilde{f}(x) = \sum_n \hat{f}_n \varphi_n(x) \qquad \text{where} \qquad \varphi_n(x) = \Phi(x/h - n)$$

Find the "fundamental" spline $\Psi(u) = \begin{cases} 1, & u = 0, \\ 0, & u = \pm 1, \pm 2, \ldots . \end{cases}$

by solving for ω_m

$$\Psi(u) = \sum_m \omega_m \Phi(u - m)$$

(an infinite banded linear system)

Then we can use the B-splines like nodal basis functions:

$$\tilde{f}(x) = \sum_n f(nh)\Psi(x/h - n) = \sum_m \hat{f}_m \Phi(x/h - m)$$

where $\hat{f}_m = \sum_n \omega_{m-n} f(nh)$

Computationally, it is quite cheap to calculate ω_m and we do it only once up front for choice of spline degree.

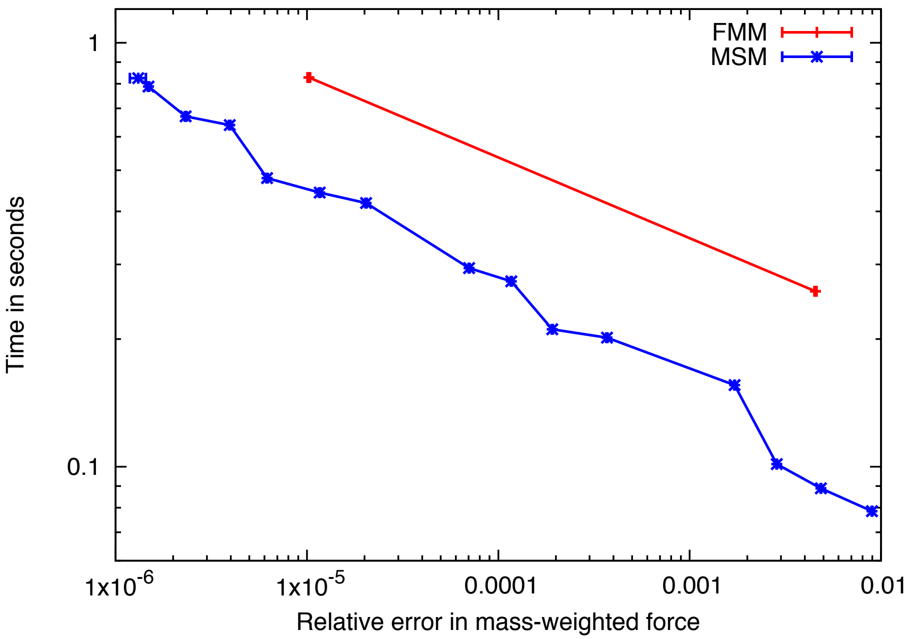
The coefficients have to be convolved with the grid interaction stencils which is expensive.

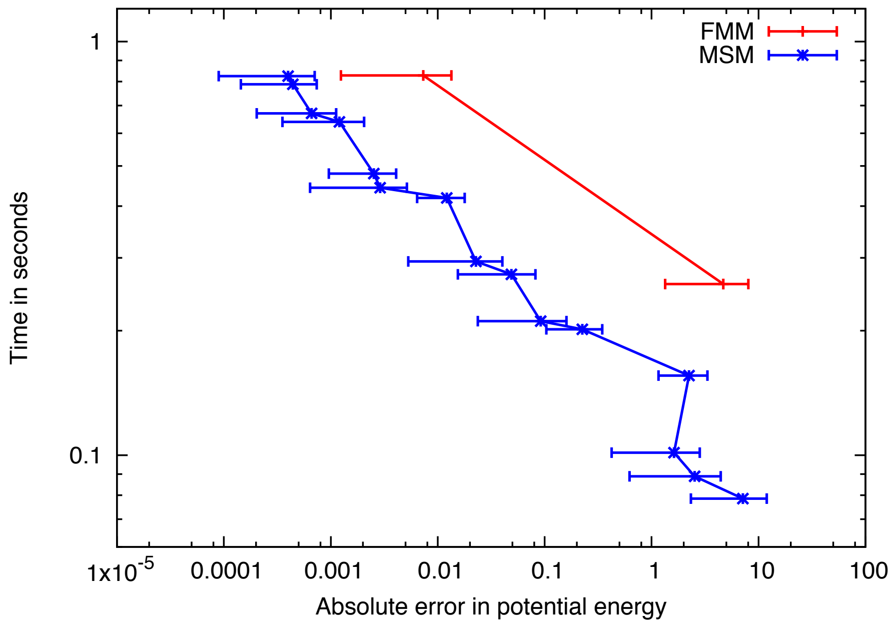
We can use symmetry (up to 48-fold) to reduce the work.

The stencils are no longer spherical, the corners are also filled.

Keeping the grid interaction stencil sizes the same, this is no longer pure interpolation, rather quasi-interpolation, exact for degree p-1 polynomials so preserving pth order accuracy.

Performance of MSM vs. FMM





Comparing single core performance with
Uniform FMM Laplace Solver (B Zhang and J Huang)
on 30K-atom water sphere

Hardy, *et al.*, *J. Chem. Phys.* **144**:114112, 2016

Acknowledgments

- Collaborators:
  - Robert Skeel (Purdue)
  - Zhe Wu, Jim Phillips, John Stone (UIUC)
- Funding:
  - NIH Grant 9P41GM104601
  - NSF Grants CHE090957273 and CCF08-30582