Neutral Territory Decomposition for Parallel MD

Aiichiro Nakano

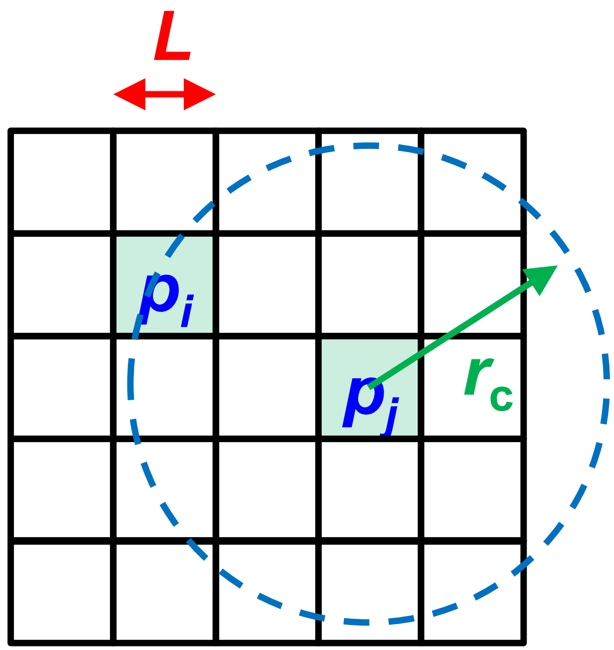
Collaboratory for Advanced Computing & Simulations
Department of Computer Science
Department of Physics & Astronomy
Department of Chemical Engineering & Materials Science
Department of Biological Sciences
University of Southern California

Email: anakano@usc.edu

D. E. Shaw, "A fast, scalable method for the parallel evaluation of distance-limited pairwise particle interactions," *J. Comput. Chem*. **26**, 1318 ('05)

Fine Granularity

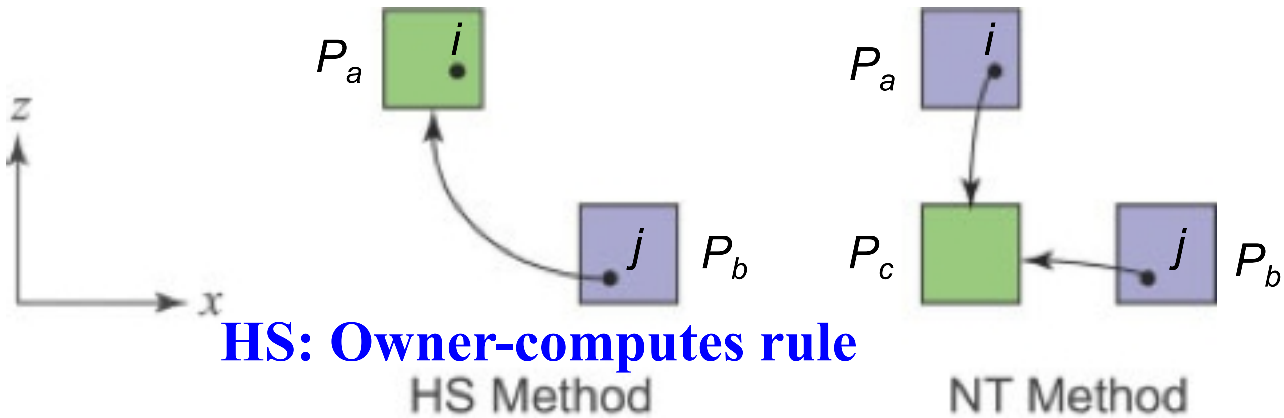
Number of atoms per process (N/P) ~ 1



spatial subsystem length (L) << interaction cutoff (r_c)

Spatial (Half-Shell) *vs.* NT Decompositions

Locus of interaction — who does what (2-dimensional example)

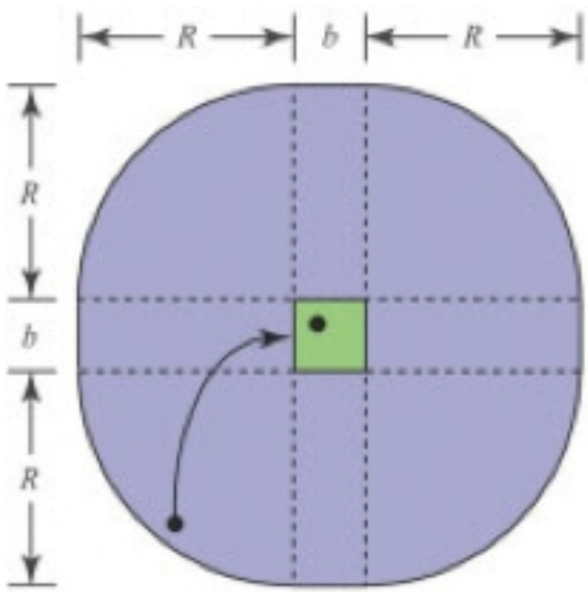


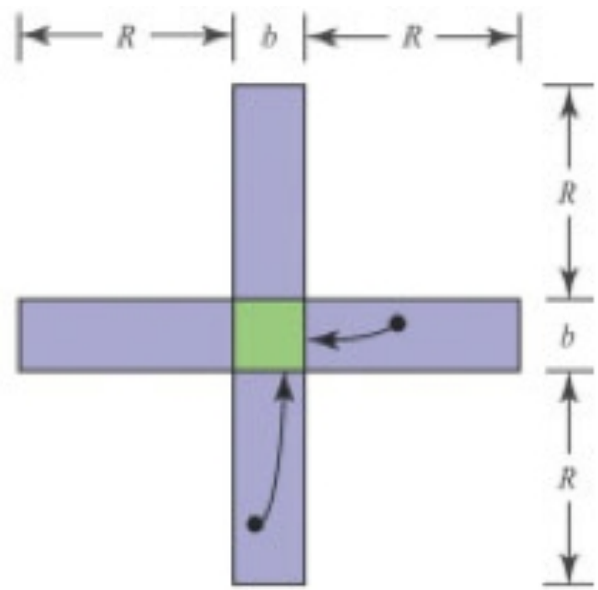
HS: Owner-computes rule

NT = hybrid spatial (data) & force (computation) decomposition with well-designed order/layout

Import regions or communication volume (2-dimensional example)

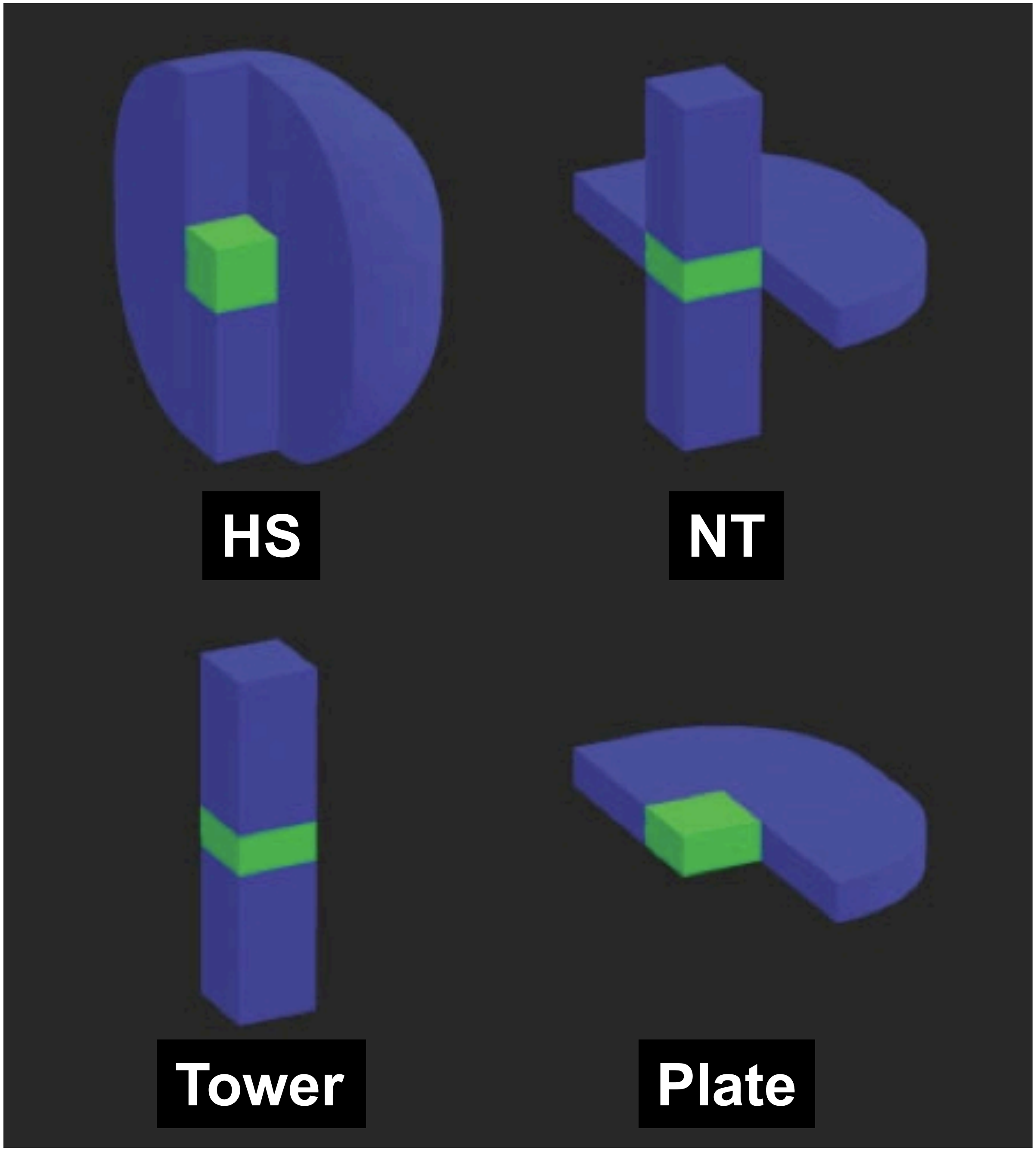
HS
$4bR+\pi R^2$
$\rightarrow$const.
$(b\rightarrow 0)$



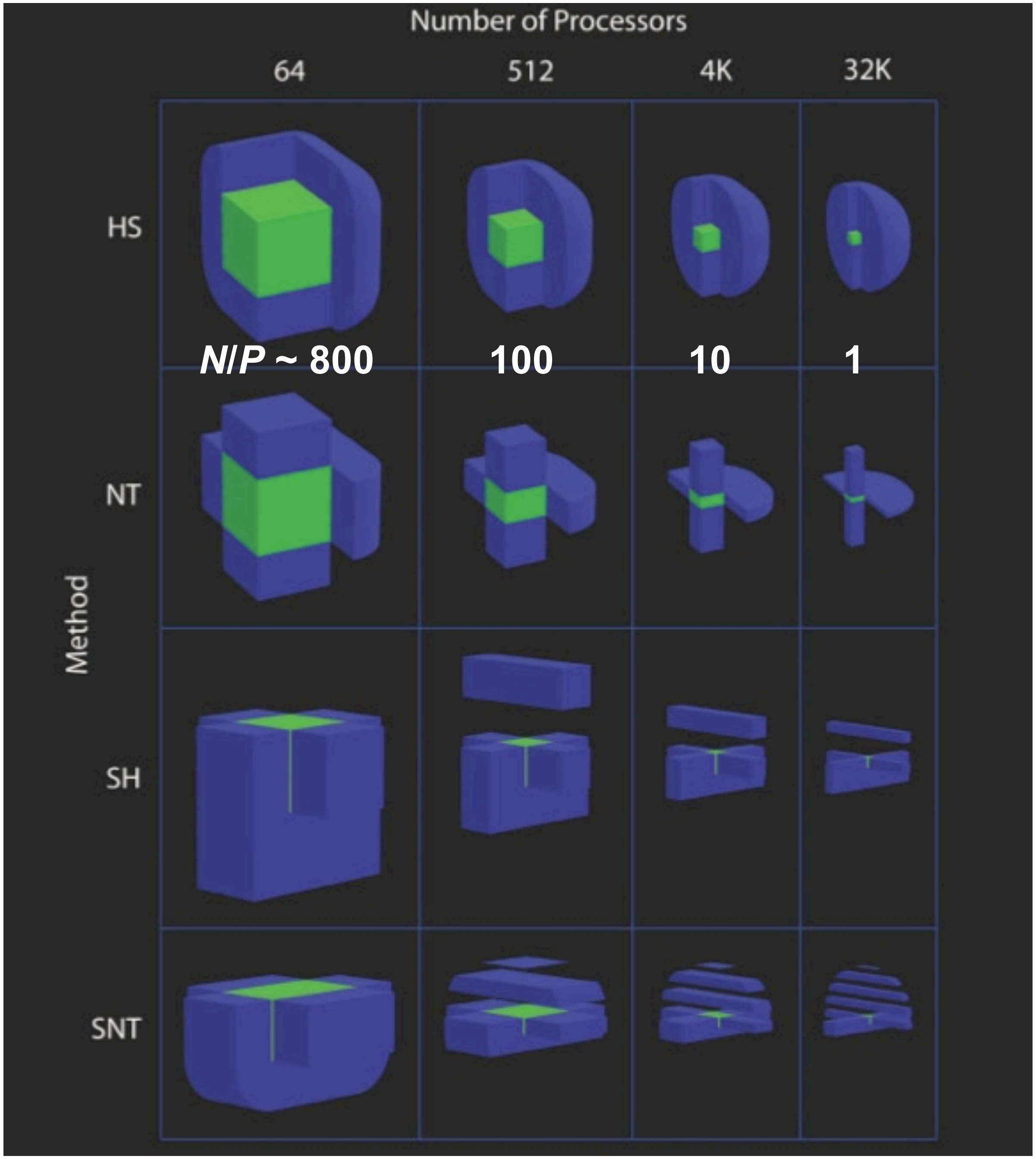


NT
$4bR$
$\rightarrow 0$
$(b\rightarrow 0)$

3D Import Regions



Scaling of Import Regions



Marc Snir

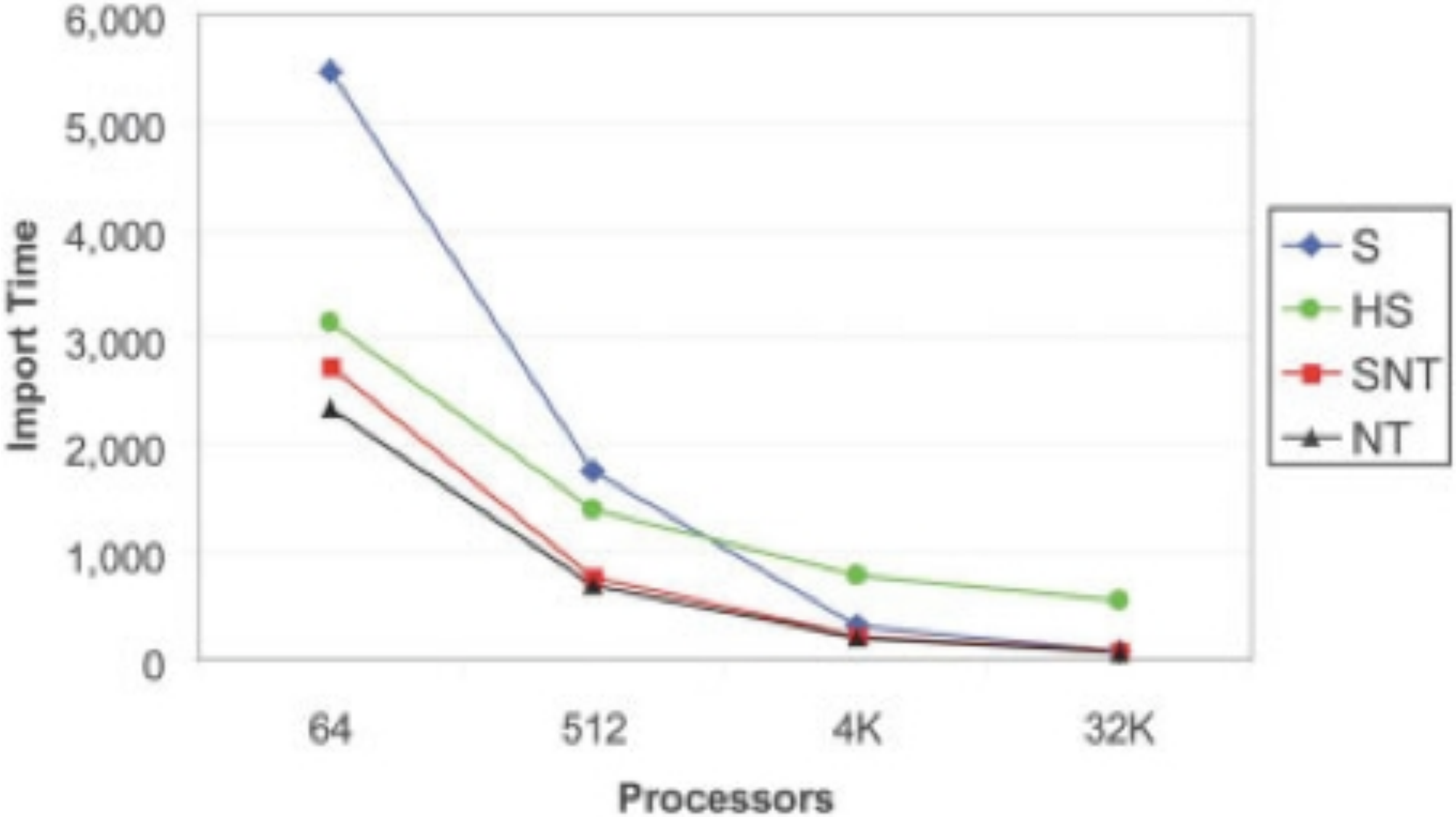
Scaling of the Volume of Import Regions

HS decomposition

$$V_i = O\left(R^3\right)$$

NT decomposition

$$V_i = O\left(R^{3/2} p^{-1/2}\right)$$



Communication time

$$T_{\text{comm}} = t_{\text{latency}} N_{\text{message}} + \frac{1}{b_{\text{bandwidth}}} V_{\text{message}}$$

# of messages

volume (Bytes) of messages

ns ~ many μs

GB/s

Combine NT with ...

Cache-oblivious recursive blocking?

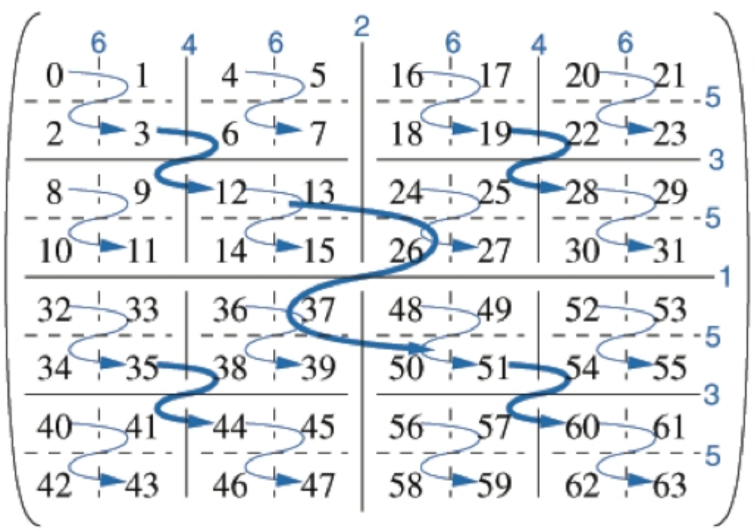
Cache-Oblivious Algorithms

EXTENDED ABSTRACT SUBMITTED FOR PUBLICATION. FOCS99

Matteo Frigo Charles E. Leiserson Harald Prokop Sridhar Ramachandran
MIT Laboratory for Computer Science, 545 Technology Square, Cambridge, MA 02139
{athena,cel,prokop,sridhar}@supertech.lcs.mit.edu

SIAM REVIEW
Vol. 46, No. 1, pp. 3–45

Recursive Blocked Algorithms and Hybrid Data Structures for Dense Matrix Library Software*



Erik Elmroth†
Fred Gustavson‡
Isak Jonsson†
Bo Kågström†

Combine NT with ...

Optimal data/computation layout (on Cell, GPU, multicore,...)?

Improving Memory Hierarchy Performance for Irregular Applications*

John Mellor-Crummey†, David Whalley‡, Ken Kennedy†

† Department of Computer Science, MS 132
Rice University
6100 Main
Houston, TX 77005
{johnmc,ken}@cs.rice.edu

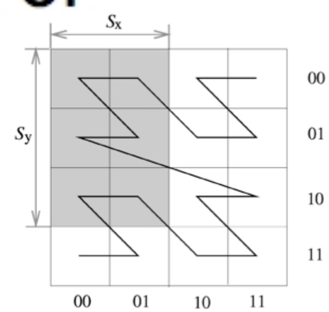
‡ Computer Science Department
Florida State University
Tallahassee, FL 32306-4530
whalley@cs.fsu.edu
phone: (850) 644-3506

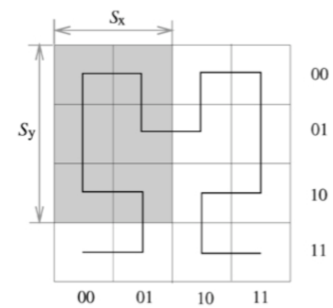
ISC99

IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 13, NO. 1, JANUARY/FEBRUARY 2001

Analysis of the Clustering Properties of the Hilbert Space-Filling Curve

Bongki Moon, H.V. Jagadish, Christos Faloutsos, *Member*, *IEEE*, and Joel H. Saltz, *Member*, *IEEE*





Metrics and Models for Reordering Transformations

Morton or Hilbert?

MSP04

Michelle Mills Strout
Mathematics and Computer Science Division
Argonne National Laboratory
Argonne, IL 60439 USA
mstrout@mcs.anl.gov

Paul D. Hovland
Mathematics and Computer Science Division
Argonne National Laboratory
Argonne, IL 60439 USA
hovland@mcs.anl.gov

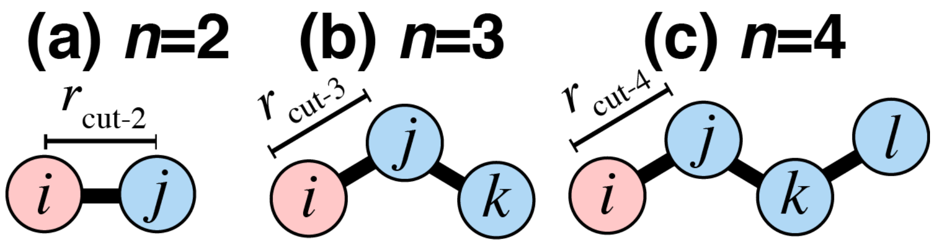
Hypergraph

G.M. Morton, "A computer oriented geodetic data base & a new technique in file sequencing," *IBM Tech. Report* ('66)

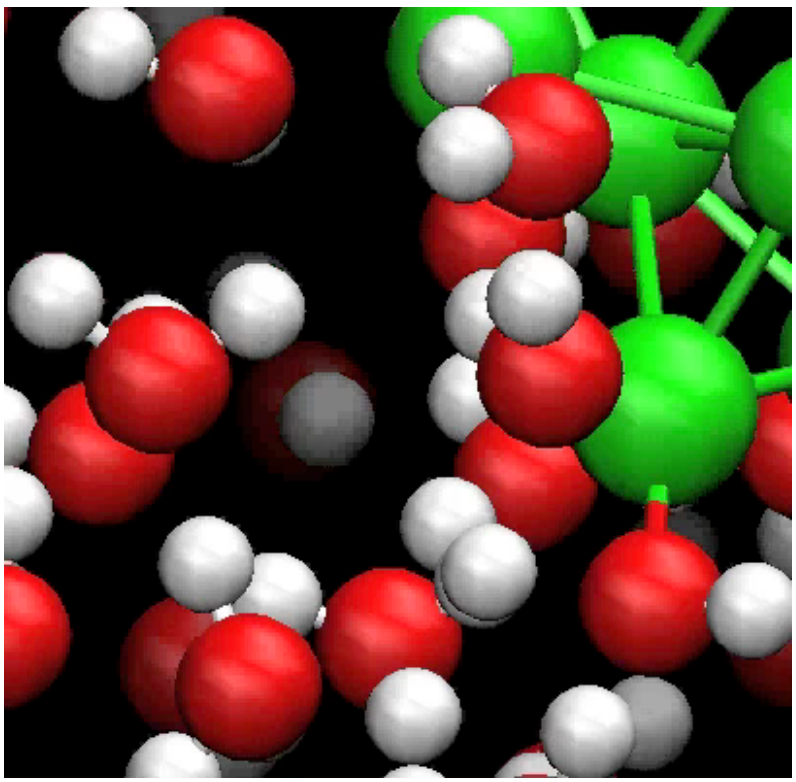
Reactive Molecular Dynamics (RMD)

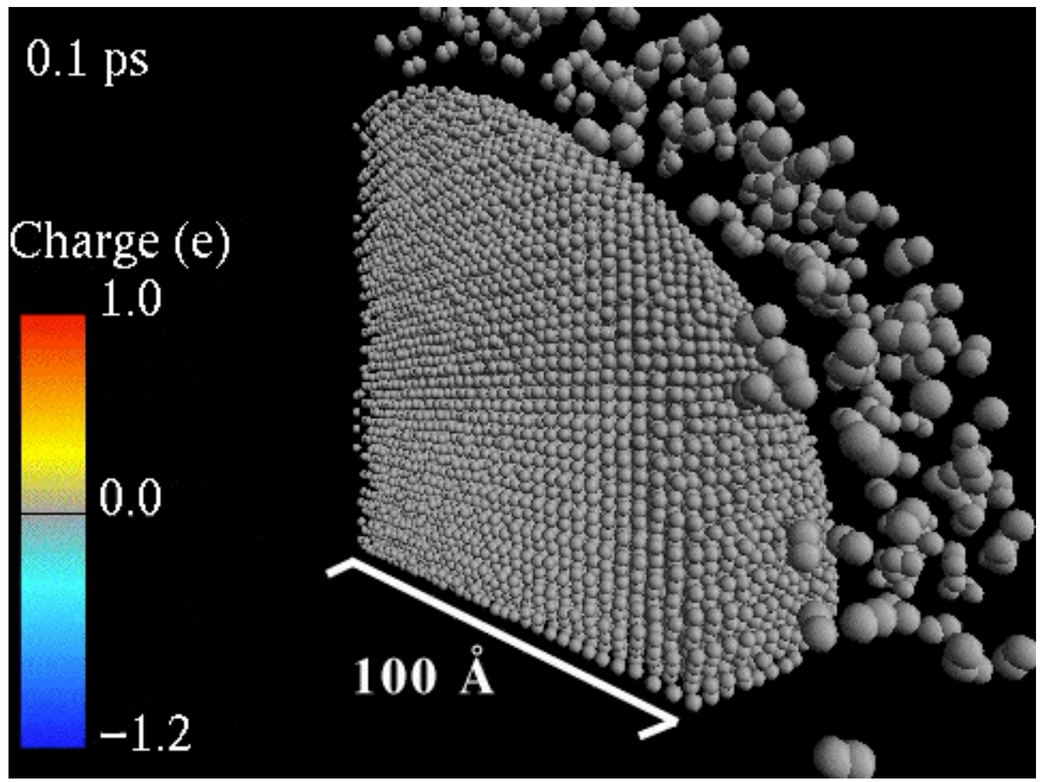
- **Dynamic *n*-tuple computation: *n* ≤ 4 explicitly; ≤ 6 through bond order**

$$\mathbf{f}_i^{(n)} = -\sum_{\forall(\mathbf{r}_0,\ldots,\mathbf{r}_{n-1})\in\Gamma^{(n)}} \frac{\partial}{\partial \mathbf{x}_i}\Phi_n(\mathbf{x}_0,\ldots,\mathbf{x}_{n-1})\Big|_{(\mathbf{x}_0,\ldots,\mathbf{x}_{n-1})=(\mathbf{r}_0,\ldots,\mathbf{r}_{n-1})}$$



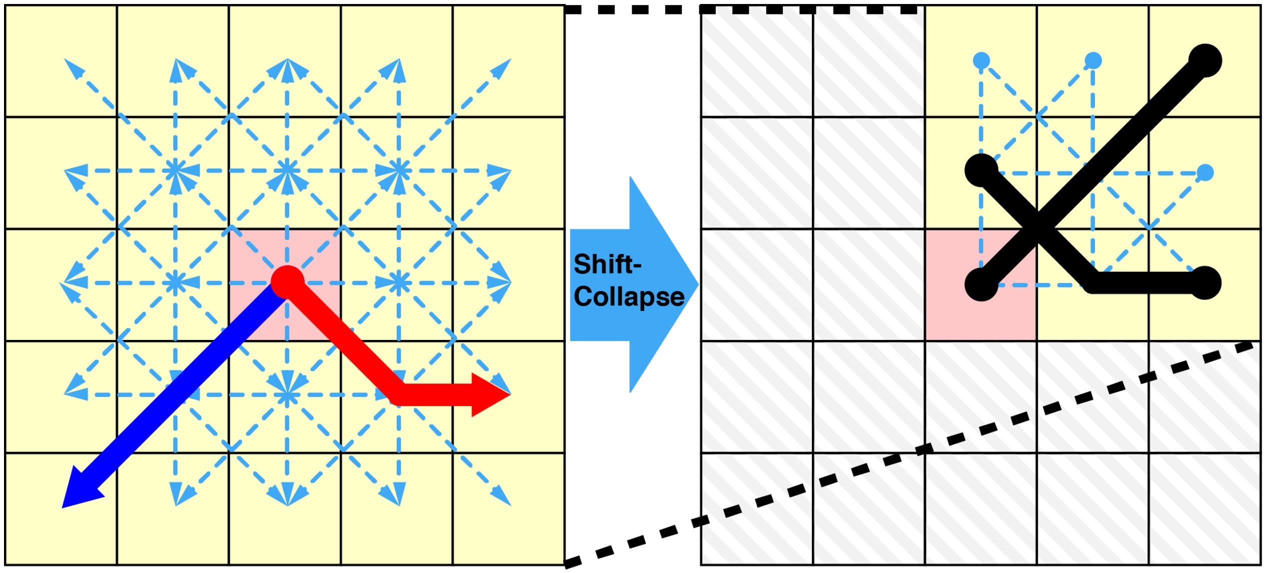
K. Nomura *et al.*, *Comp. Phys. Commun.* **192**, 91 ('15)



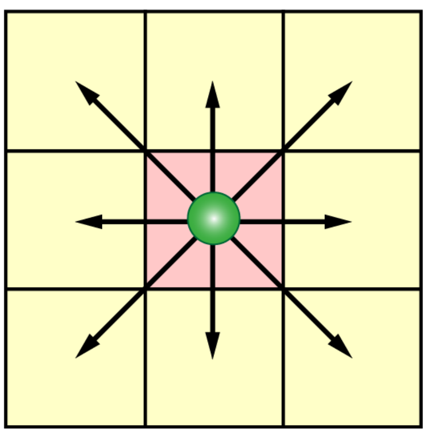


Shift-Collapse (SC) Algorithm

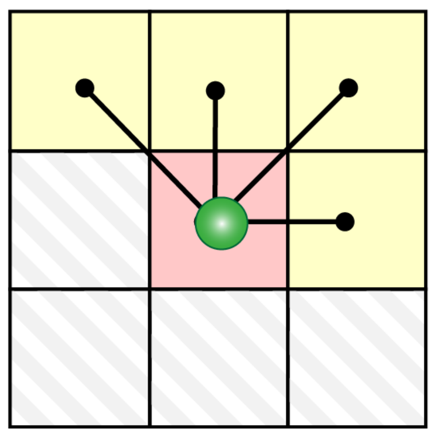
- **Generalization of Shaw's eighth-cell method (non-owner-compute method on high-latency cluster) for pair computation to general dynamic range-limited *n*-tuples**



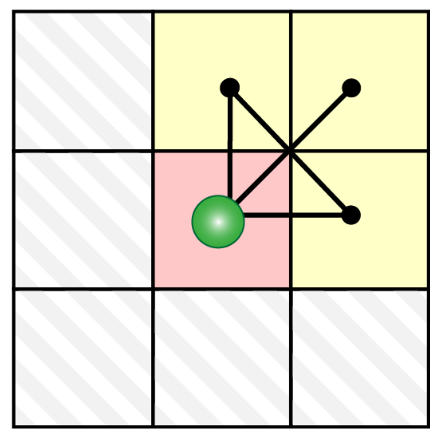
M. Kunaseth *et al*., *IEEE/ACM Supercomputing* (*SC13*)



Full-shell (FS) method
[*e.g*. Rappaport, '88]



Half-shell (HS) method
[*e.g*. Rappaport, '88]

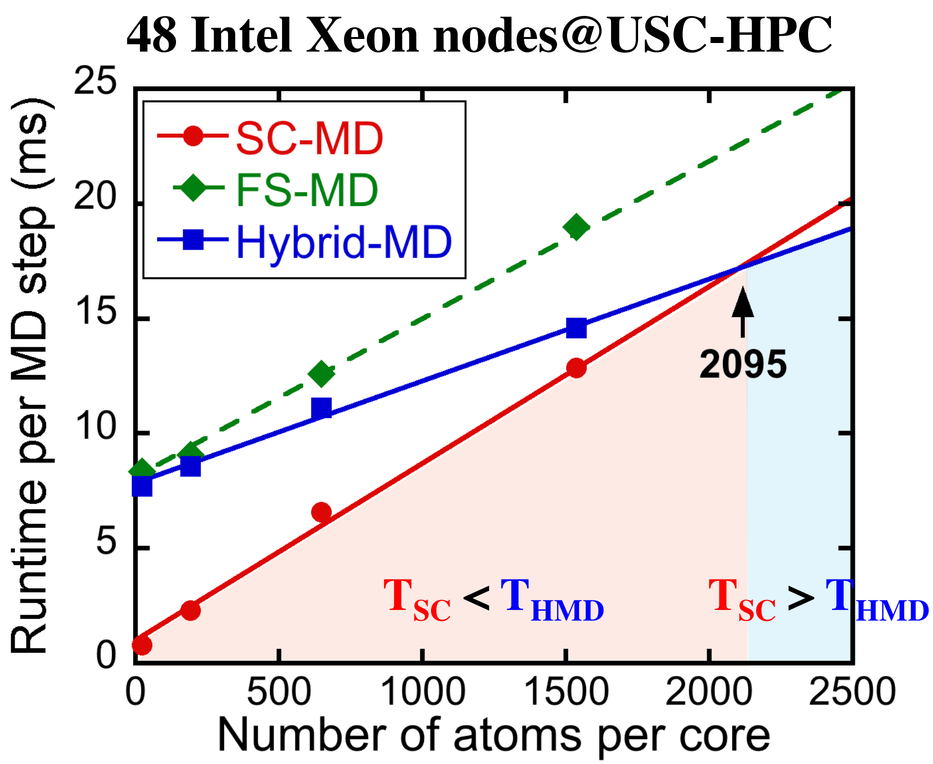


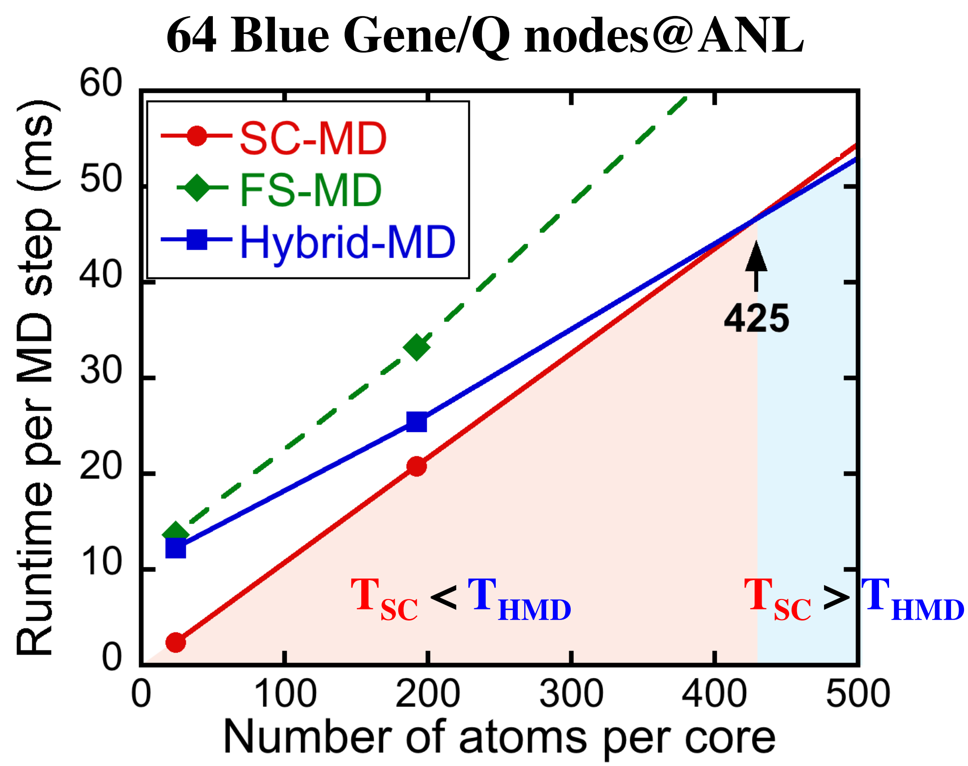
Eighth-shell (ES) method
[Bower *et al*., '06]

Shift-Collapse (SC) Performance

Runtime comparison on 48 Intel-Xeon nodes and 64 Blue Gene/Q nodes

- **SC-MD is always faster than FS-MD**
- **At the smallest grain, SC-MD is 9.7- and 5.1-fold speedups over the state-of-the-art hybrid linked-cell & neighbor list code**
- **Crossover of optimal algorithm from SC-MD to hybrid MD at larger granularity (*i.e.* N/P > 2,095 on Intel Xeon and N/P > 425)**

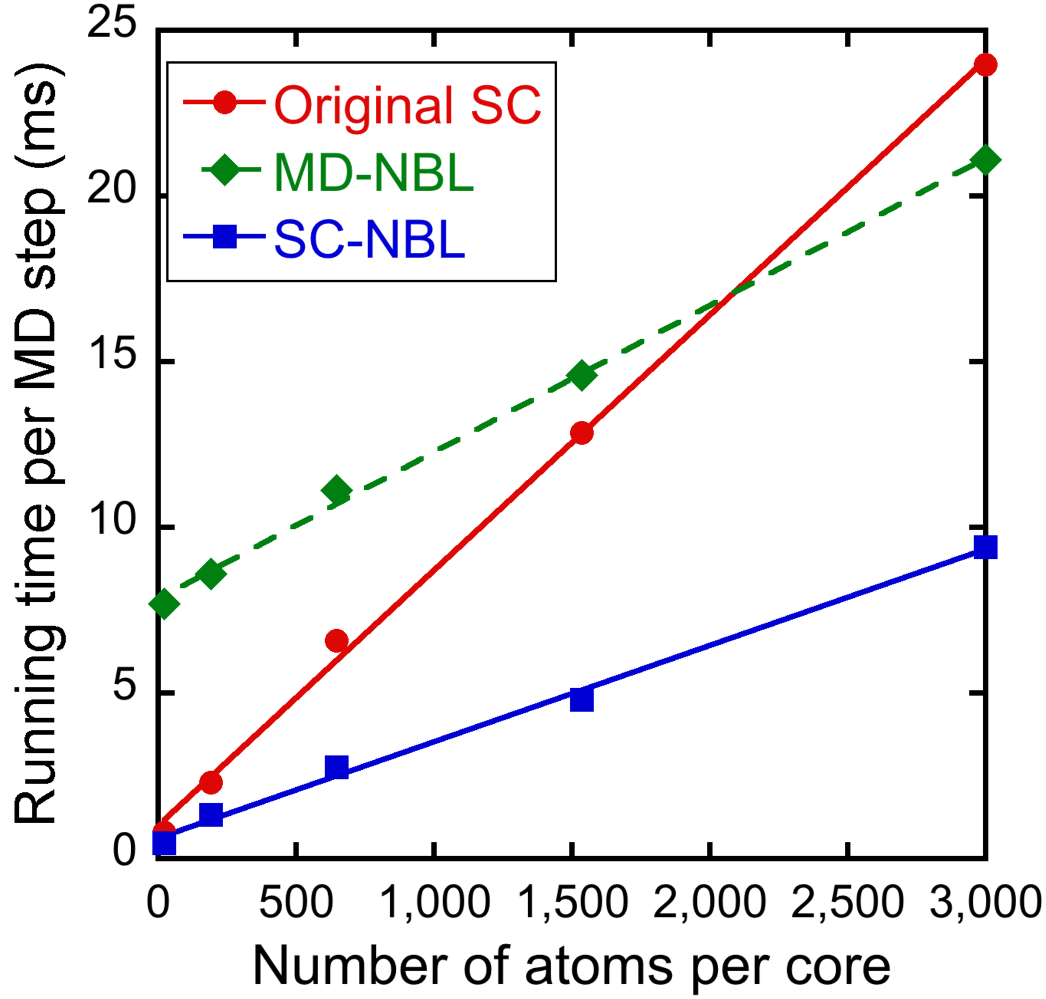




M. Kunaseth *et al*., *IEEE/ACM Supercomputing* (*SC13*)

Shift-Collapse on Neighbor List (SC-NBL)

- **Apply shift-collapse operations to the hybrid linked-cell & neighbor list code (best of both)**



"Shift/collapse on neighbor list (SC-NBL): fast evaluation of dynamic many-body potentials in molecular dynamics simulations," M. Kunaseth, S. Hannongbua, & A. Nakano, *Comput. Phys. Commun.* **235**, 88 (2019)