Performance Evaluation of NWChem Ab-Initio Molecular Dynamics (AIMD) Simulations on the Intel® Xeon Phi™ Processor

E.J. Bylaska[1], M. Jacquelin[2],
B. de Jong[2], J.R. Hammond[3],
and **M. Klemm**[3]

[1]Pacific North-west National Lab,
[2]Lawrence Berkeley National Lab, [3]Intel

Legal Disclaimer & Optimization Notice

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.

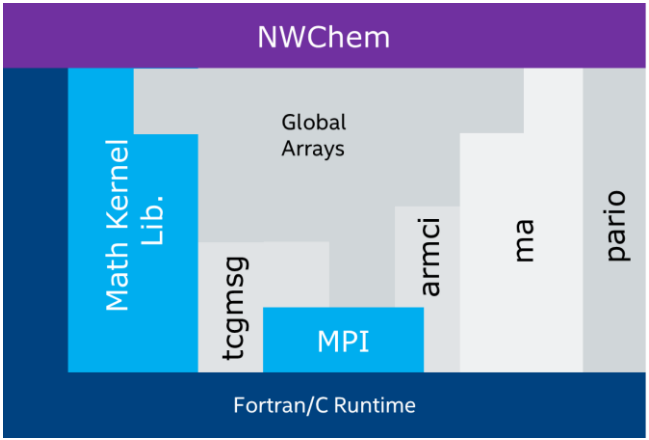
Optimization Notice

Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Notice revision #20110804

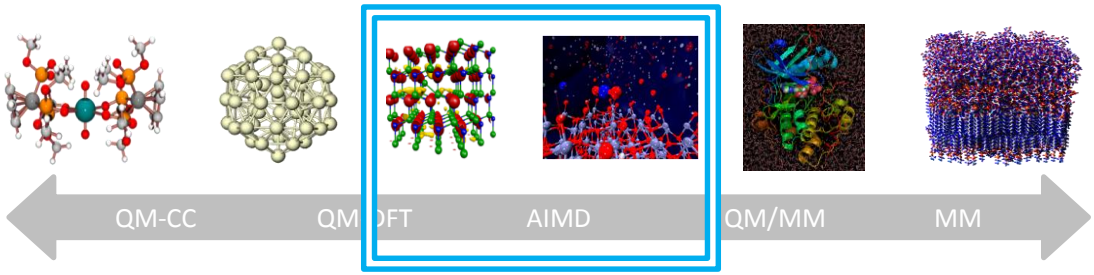
NWChem Fun Facts

- URL: www.nwchem-sw.org



Language	# files	LOC
Fortran 77	17147	4823003
C	857	242622
Bourne Shell	39	164184
HTML	182	55168
make	398	44373
TeX	115	39884
C/C++ Header	377	31217
Fortran 90	58	30525
Python	95	22571
m4	126	21894
C++	67	13489
Java	66	12311

Introduction: Plane Wave Methods



- 100-1000 atoms, uses plane wave basis
- Many FFTs and DGEMM operations
- “Meaty”: Lots of FLOPs, but also bandwidth sensitive

$$(-1/2)\nabla^2\Psi + V_{ext}\Psi + V_H\Psi + V_{xc}\Psi + V_{x,exact}\Psi = E\Psi$$

$$\langle \Psi_i | \Psi_j \rangle = \delta_{ij}$$

N_eN_g

$(N_aN_g + N_gLogN_g + N_eN_g) + N_aN_eN_g$

$N_eN_gLogN_g+N_eN_g+2N_gLogN_g+N_g+N_eN_g$

$N_eN_gLogN_g + N_eN_g$

$N_e(N_e+1)N_g\ LogN_g$

$N_e^2N_g + N_e^3$

N_a - number of atoms
N_e - number of electrons
N_g – size of FFT grid

Strong Scaling is Key

- 20 psec of simulation time ≈ 200,000 steps
  - 1 sec/step = 2-3 days simulation time
  - 10 sec/step = 23 days simulation time
  - 13 sec/step = 70 days simulation time
- Mesoscale phenomena at longer time scales
  - Assume 1 sec/step
  - 100 psec = 10-15 days simulation time
  - 1 nsec = 100 - 150 days simulation time
- Strong scaling required to reduce time per time step as much as possible
  - At least below 1sec/step

Plane Wave Discretization

$$H\psi_i(r) = -\tfrac{1}{2}\nabla^2 + V_L(r) + (1-\alpha)V_X[\rho](r) + V_C[\rho](r)$$
$$+ V_{NL} + V_H[\rho](r) - \alpha\sum_j K_{ij}(r)\psi_i(r)$$

Matrix multiplication in reciprocal space, SUMMA

$$\nabla^2 V_{H,X,C}(r) = -4\pi\rho(r) \qquad \nabla^2 K_{ij}(r) = -4\pi\psi_i(r)\psi_j(r)$$

N_e 3D-FFT Poisson $(N_e+1)N_e$ 3D-FFT

$$\rho(r) = \sum_{i=1}^{N}|\psi_i(r)|^2 \qquad \int_W \psi_i(r)\psi_j(r)dr = \delta_{ij}$$

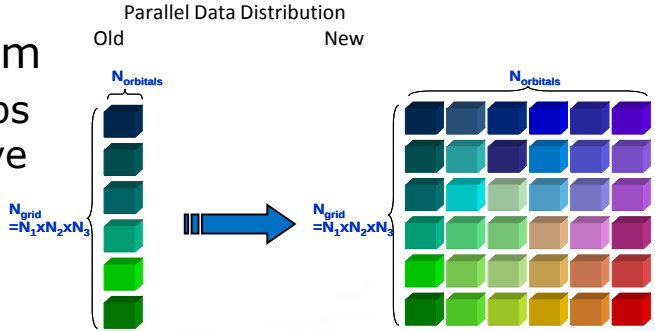
Density Orthogonality (matrix multiplication, SUMMA)

Plane Wave AIMD Computational Hotspots

- Three major hotspots with different performance behavior
- Applying V_H and V_{xc}, involving calculation of $2N_e$ 3D FFTs
- Non-local Pseudopotentials, V_{NL} with matrix multiplications of tall-and-skinny matrices
- Enforcing Orthogonality, with matrix multiplications of tall-and-skinny matrices

Plane Wave Data Distribution

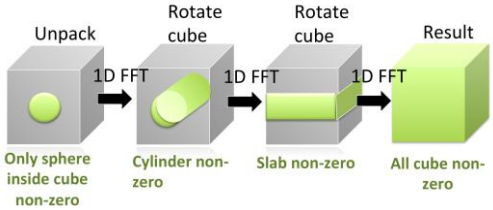
- Smart data distribution and communication algorithms enable DFT and hybrid-DFT to scale to large numbers of processors
- Multiple levels of parallelism
  - k-points through subgroups (fewer of them as we move to larger systems)
  - Parallel distribution of grid points and orbitals



Gygi et al. Supercomputing '06

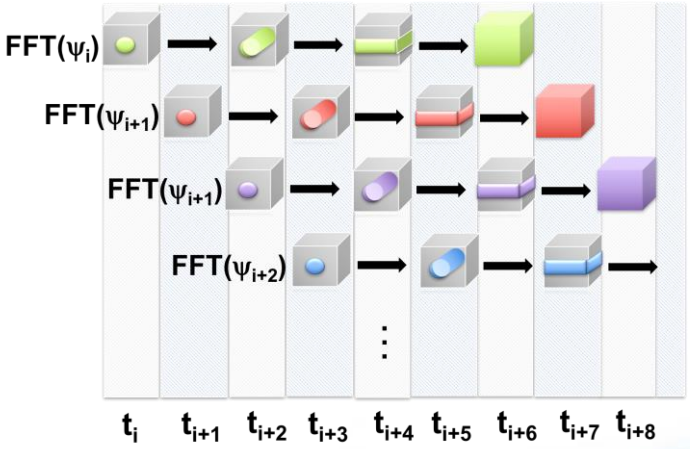
3D FFTs

- Performed at each step
  - 2 N_e 3D FFTs for DFT
  - Plus (N_e+1)*N_e 3D FFTs for hybrid DFT
- In reciprocal space, sphere of radius E_{cut} is stored
- Forward FFTs:
  1. Unpack sphere into a 3D cube (z,x,y)
  2. Perform $n_x \times n_y$ 1D FFTs along the z-dimension
  3. Rotate the cube ($z; x; y$) → ($y; z; x$)
  4. Perform $n_z \times n_x$ 1D FFTs along the y-dimension
  5. Rotate the cube ($y; z; x$) → ($x; y; z$)
  6. Perform $n_y \times n_z$ 1D FFTs along the x-dimension
- Backward FFTs: reverse above steps



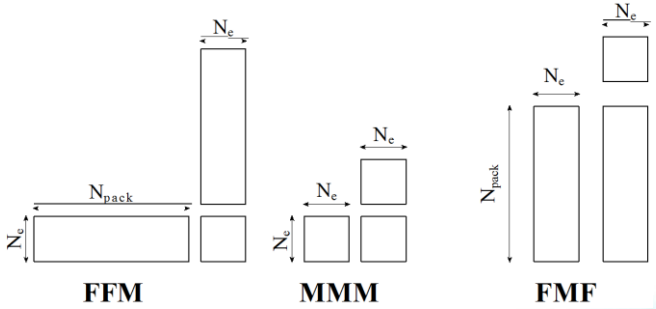
3D FFTs – Pipelined Implementation

- Performed at each step
  - 2 N_e 3D FFTs for DFT
  - Plus $(N_e+1)*N_e$ 3D FFTs for hybrid DFT
- In reciprocal space, sphere of radius E_{cut} is stored
- 3D FFTs are pipelined
  - Overlap communication and computation
  - Latency reduction
  - N^2 1D FFTs per stage execute in parallel

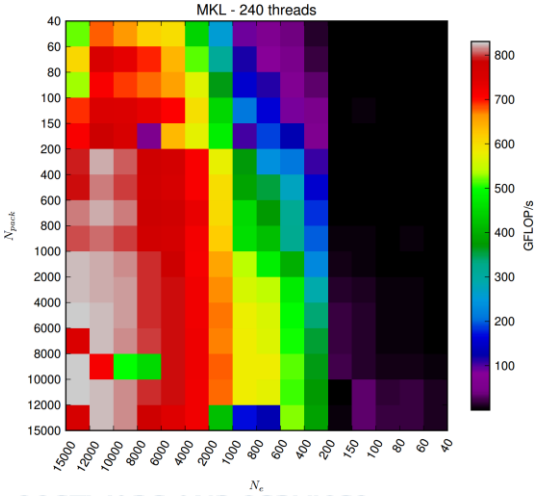


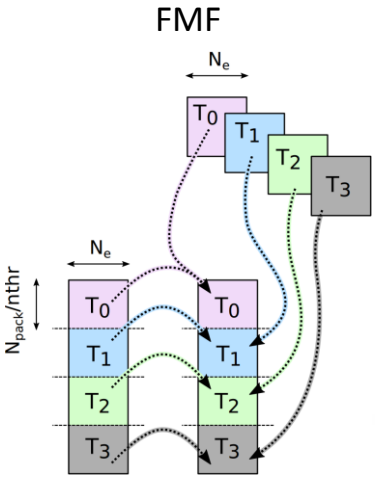
Lagrange Multiplier

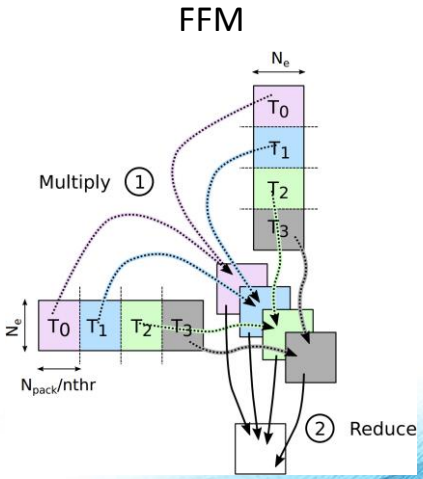
- Sequence of matrix products of shape F or M
  - F: N_{pack} x N_e or N_e x N_{pack} matrix (tall & skinny)
  - M: N_e x N_e matrix
  - In general: N_{pack} >> N_e



Lagrange Multiplier – Parallelization







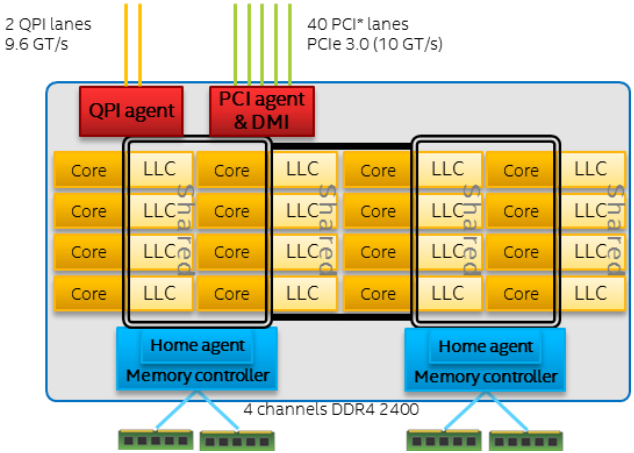
SOFTWARE AND SERVICES

Experimental Setup – NERSC Cori

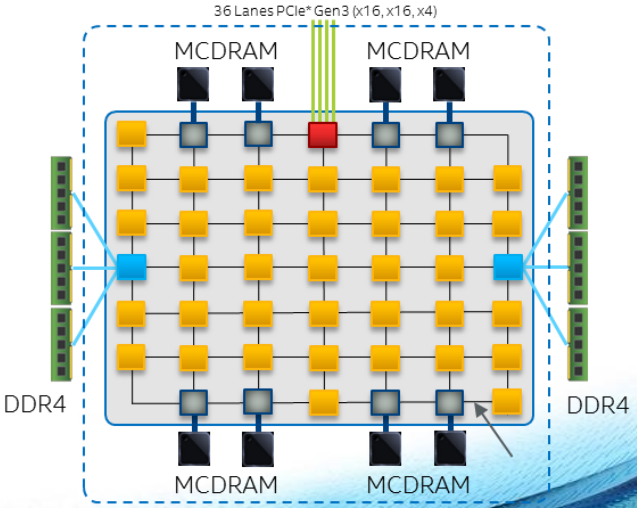
- "Haswell", HSW
  - Cray* XC40
  - 2S Intel® Xeon® E5-2698v3 processors
  - 32 cores, no Hyper-Threading
  - 2.3 GHz clock frequency
  - 128 GB of DDR4 at 2133 MHz
  - Cray* Aries* w/ Dragonfly
- "Knights Landing", KNL
  - Cray* XC40
  - Intel® Xeon Phi™ 7250 processors
  - 68 cores w/ 4 hardware threads
  - 1.4 GHz clock frequency
  - 96 GB of DDR4 at 2400 MHz
  - Cache mode
  - Quadrant cluster mode
  - Cray* Aries* w/ Dragonfly

Experimental Setup – Processor Architectures

- Intel® Xeon® E5-2697v3



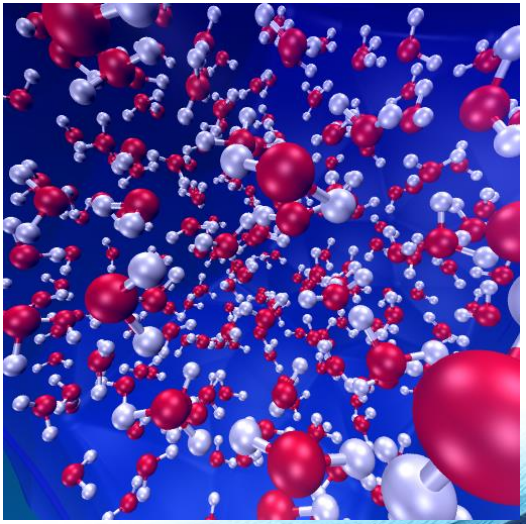
- Intel® Xeon Phi™ 7250



SOFTWARE AND SERVICES

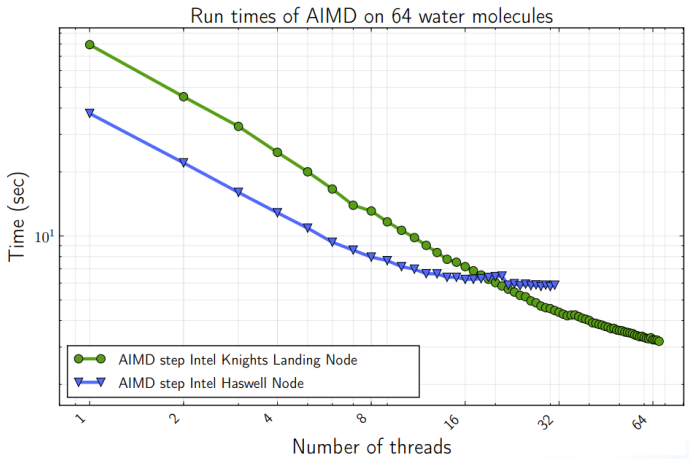
Experimental Setup – Benchmarks

- water64:
  - 64 water molecules in a box
  - test intra-node strong scaling
- water256:
  - 256 water molecules
  - test cluster strong scaling
  - N_e=2056
  - N_g=5,832,000 (180^3)
  - N_{pack}=437,000



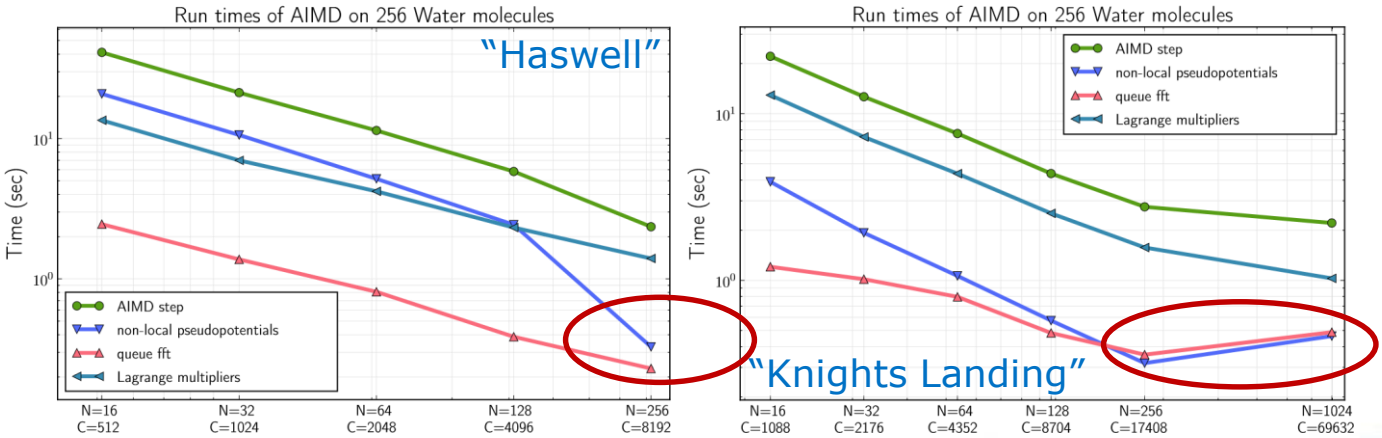
Intra-node Performance

- Insight into performance without fabric effects
- Xeon node saturates at about 16 cores, reaching memory bandwidth limits
- Xeon Phi node keeps strong scaling due to the on-package cache memory
- 1.8x speed-up of KNL over HSW node



Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests are measured using specific computer systems, components, software, operations, and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. System configuration: Cray* XC40 system, 2S Intel® Xeon® E5-2698v3 processor, Intel® Hyper-Threading technology disabled, 128 GB of DDR4 (8x 16 GB, 2133 MHz), Cray* Aries interconnect with Dragonfly topology; Cray* XC40 system Intel® Xeon Phi™ 7250 processors, 96 GB of DDR4 (6x 16GB, 2400 MHz), quadrant cluster mode, MCDRAM in cache mode, Cray* Aries interconnect with Dragonfly topology.

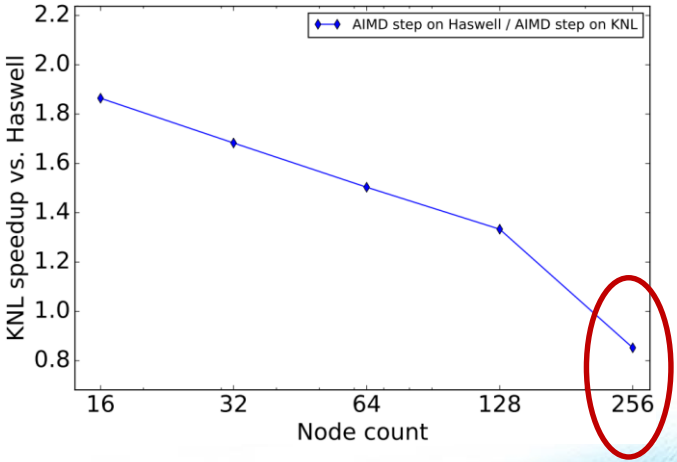
Performance



Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests are measured using specific computer systems, components, software, operations, and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. System configuration: Cray* XC40 system, 2S Intel® Xeon® E5-2698v3 processor, Intel® Hyper-Threading technology disabled, 128 GB of DDR4 (8x 16 GB, 2133 MHz), Cray* Aries interconnect with Dragonfly topology; Cray* XC40 system Intel® Xeon Phi™ 7250 processors, 96 GB of DDR4 (6x 16GB, 2400 MHz), quadrant cluster mode, MCDRAM in cache mode, Cray* Aries interconnect with Dragonfly topology.

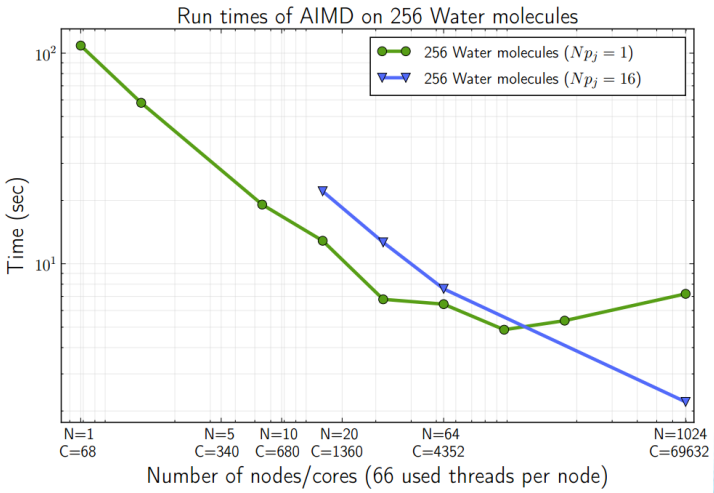
Relative Performance – HSW vs KNL

- Strong scaling regime
- Interconnect latency becomes visible
- Less occupancy of the network
- KNL seems to suffer from this more than HSW does



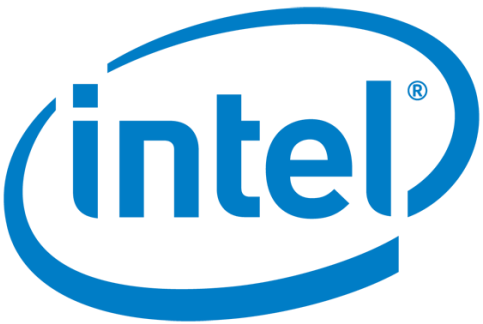
Performance – Effect of the Processor Grid

- Processor grid is a tradeoff
- 2D processor grid: $N_p = N_{pi} * N_{pj}$
- Large N_{pj} favors FFTs and non-local pseudopotentials
- Lagrange multiplier suffers from large N_{pi}
- Balancing N_{pi} and N_{pj} is required
  - problem size
  - number of ranks



The Last Slide…

- Hybridization of NWChem's AIMD code with MPI/OpenMP
- Plain library approaches were not good enough due to special requirements of the AIMD kernels
- Experiments show that AIMD can scale to O(1k) KNL nodes
- Future Work:
  - Re-use ideas of this work for hybrid DFTs and band-structure code
  - Investigate relationship between N_{pi} and N_{pj}
  - Apply all this to a "real" scientific problem ☺


intel®