Optimizing Molecular Dynamics and Stencil mini-applications for Intel's Knights Landing

IXPUG, SC'16 LIGHTNING TALK

KAVITHA CHANDRASEKAR*, LAXMIKANT V. KALE

UNIVERSITY OF ILLINOIS AT URBANA CHAMPAIGN

Charm++ Runtime System

Supports an asynchronous message-driven execution model

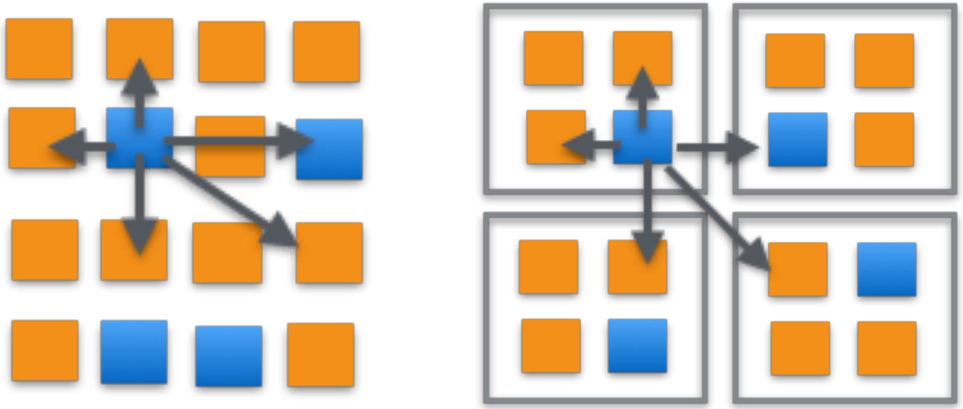
Overdecomposition of work units (or chares) on PEs (Processing Elements)

Migratability of work units

Load balancing framework

Support for several applications

- NAMD
- OpenAtom
- ChaNGa



Over-decomposition of work into Charm++ objects

Mapping of Charm++ objects to processors

Charm++ RTS view of over-decomposition

LeanMD – Molecular Dynamics Mini-application

Molecular Dynamics simulation application

Atoms in the simulation are divided spatially into k-cells roughly the size of the cutoff distance

- 3-away strategy

Uses PINY MD physics engine for its molecular dynamics calculations

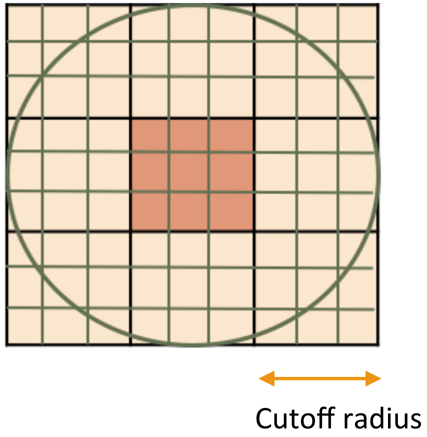
Computation bound

- Computation of interacting forces

Communication involved

- Exchange of atoms based on calculated position

Load imbalance from exchange of atoms



LeanMD work units

Stencil3D Application

Stencil application is representative of communication pattern with neighboring chares in 2D or 3D grid

- Example is MIMD Lattice computation

3D chare array of doubles

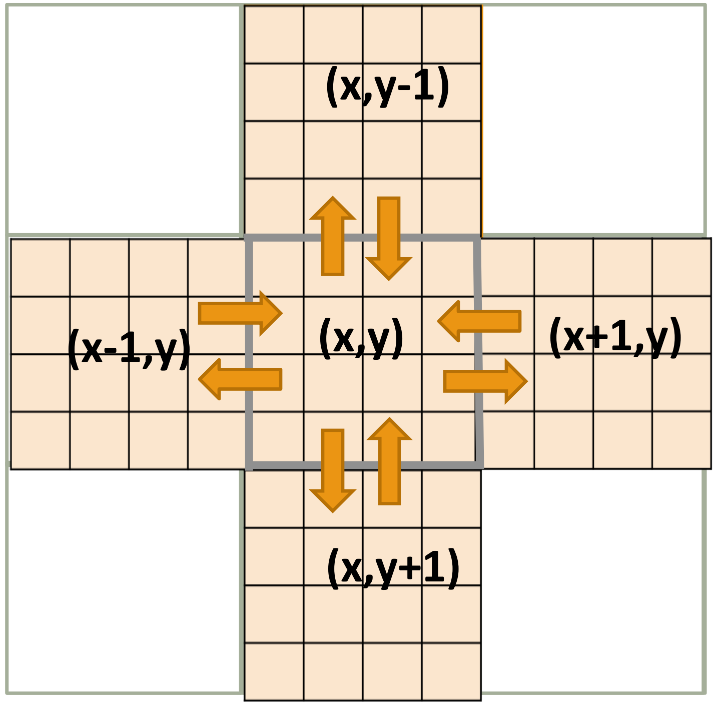
Regular communication pattern with 7 neighbor chares

Computation performed on received data

- Temperatures are updated on receiving boundary data from neighbors

Communication/Memory bound

No load imbalance



Five point communication for 2D stencil

Experimental Setup

Charm++ in SMP mode

- Reduces message copy
- Ability to map workers to cores

Single KNL node (Stampede 1.5)

- Quadrant mode
- Cache Mode (except for comparison of Cache and Flat mode)
- 256 cores were used out of 272 cores for mapping work units to PEs

Performance Tools

- IPC, MCDRAM and DDR4 accesses – perf stat
- Energy numbers – RAPL counters

LeanMD – Performance Analysis

Load balancing using Greedy heuristics

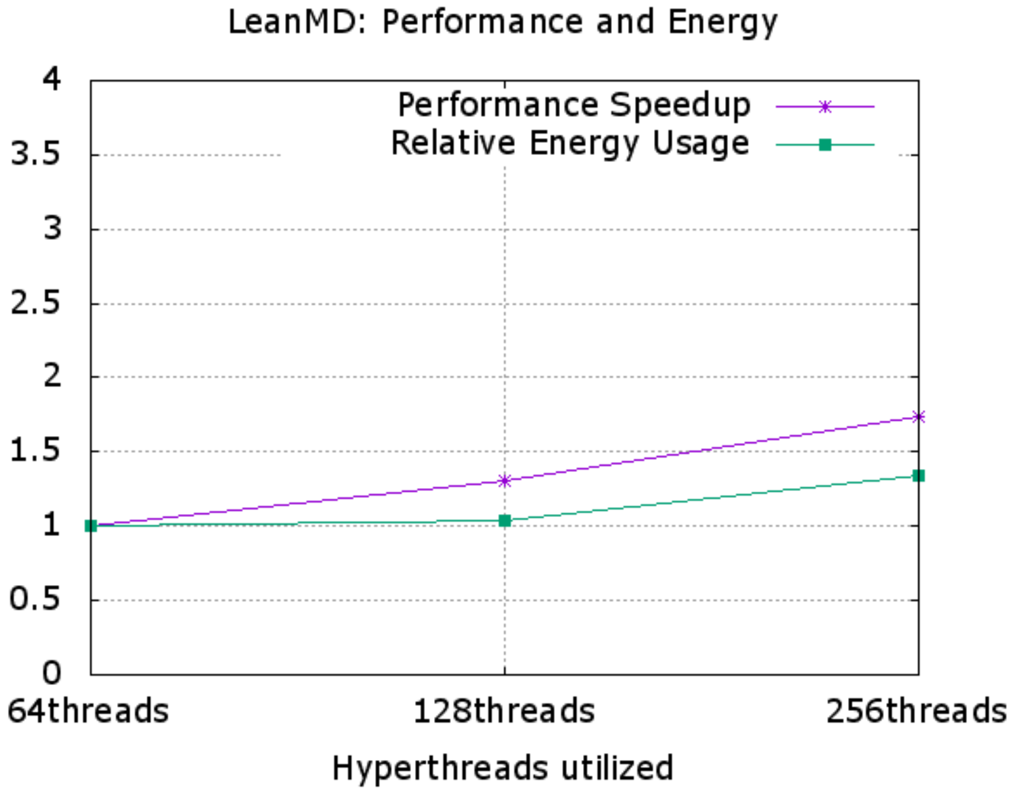
Speedup (Higher the better)

Energy Usage (Lower the better)

Input size:

- Cell and Compute Dimensions
- 16 X 16 X 16 cells

CPU Pinning: No significant degradation was observed without CPU pinning



Stencil3D - Performance analysis

Large: Stencil3d

- 2048 X 2048 X 1024
- Block dimensions: 256 X 128 X 128

Optimal configuration for smaller working set sizes was also observed to be 128 threads (2 hyperthreads per core)

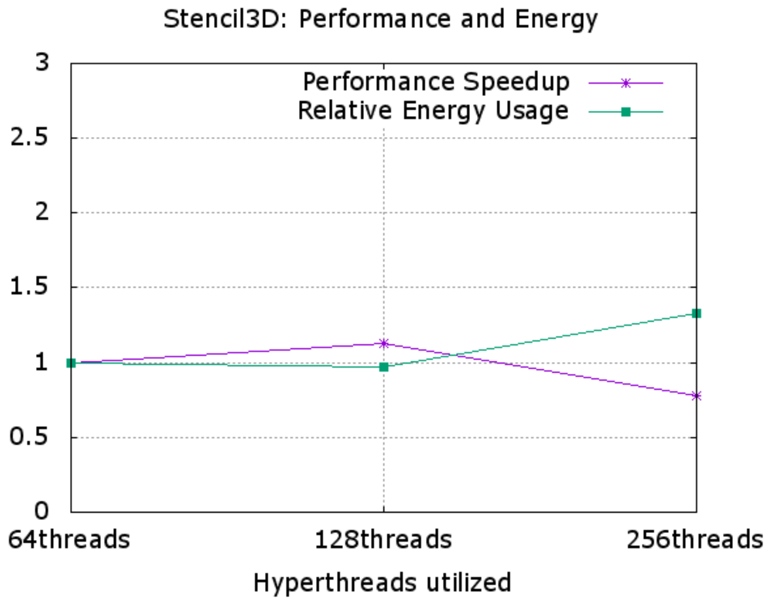
Factors for lack of performance improvement

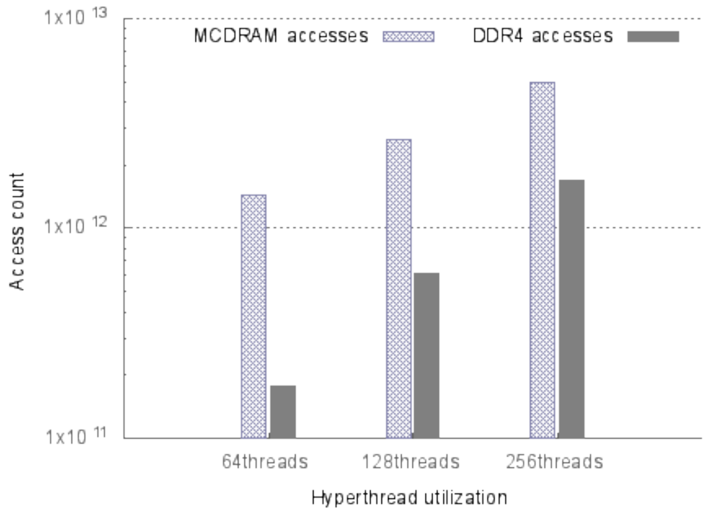
- Working set size
  - Potentially causes increased eviction from MCDRAM
- Higher memory streams in parallel
- Overhead from higher number of worker threads

Scope for energy efficient execution with 2 hyperthreads per core

CPU-pinning: 10%-30% performance degradation without CPU-pinning

- Lack of benefits from L2 locality, context switches, thread migrations
- Load imbalance due to variation in core frequencies



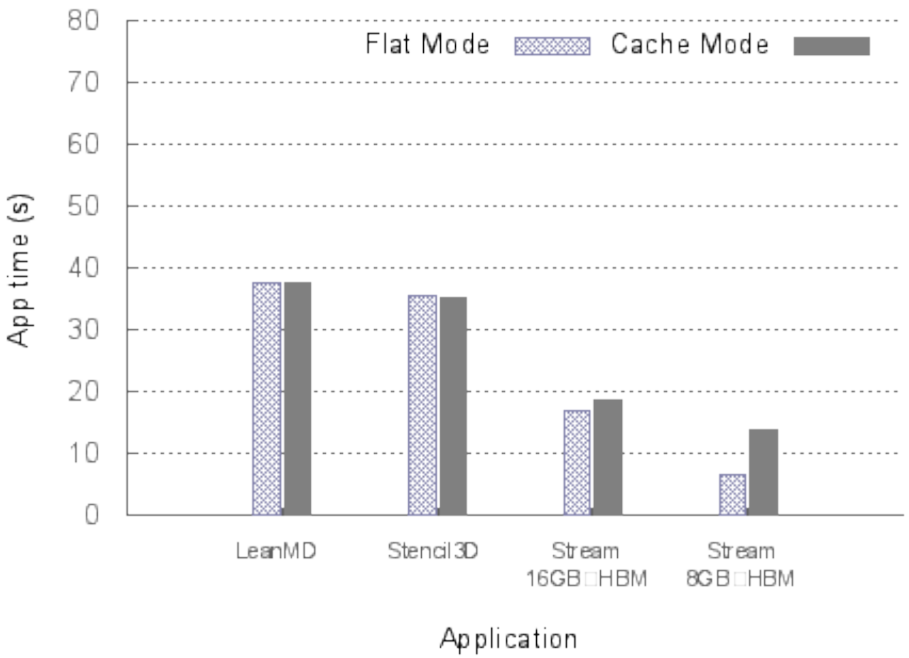


Flat mode vs. Cache mode

LeanMD and Stencil3D do not show any significant performance difference when run on Cache mode vs Flat mode

Input data size fits in MCDRAM in both cases

Stream is the only benchmark that benefits from Flat mode for input data size half the size of MCDRAM



Conclusion and Future work

HPC applications like LeanMD and Stencil3D can benefit in terms of performance and energy when tuned appropriately on Knights Landing

- Depending on working set size there might be thrashing of HBM in Cache mode
- Memory bound applications might not benefit from using all 4 hyperthreads
- Overdecomposition factor is a key influencer in performance depending on the number of worker PEs
- Variation in core frequencies can lead to load imbalance making CPU pinning or similar techniques necessary

Automation of tuning for application characteristics within the Charm++ framework by adjusting knobs like

- Hyperthreading
- CPU affinity
- Scheduling on nodes with preferred HBM usage mode