OPTIMIZATION OF LATTICE QCD WITH CG AND MULTI-SHIFT CG ON INTEL XEON PHI COPROCESSOR

Hirokazu Kobayashi (Intel),
Yoshifumi Nakamura (RIKEN AICS),
Shinji Takeda (Kanazawa University),
Yoshinobu Kuramashi (University of Tsukuba / RIKEN AICS)

Outline of the Presentation

- Motivation
- Implementation of Vectorized CG
- Performance Evaluation of Hopping Term and CG on 1 cards
- Performance Evaluation on Multi Cards
- Performance Evaluation of Multi shift CG
- Conclusion

Motivation

Develop High Performance Lattice QCD Implementation on Xeon Phi

Optimize Wilson Clover Fermion Operator

- $D = 1 + C - \kappa \underline{\sum_{\mu=1}^{4}\left(\left(1 - \gamma_\mu\right)U_{+\mu}(n)\delta_{n,m+\hat{\mu}} + \left(1 + \gamma_\mu\right)U_{-\mu}(n)\delta_{n,m-\hat{\mu}}\right)}$

  - $C = \frac{i}{2}\kappa cSW \sigma_{\mu\nu}F_{\mu\nu}(n)\delta_{m,n}$ (Clover Term)

Hopping Term

Implementation of Vectorized CG for Xeon Phi™ Coprocessor

- Run Native mode on MIC
- Double Precision CG Solver
- Full Intrinsics Implementation Kernel
- MPI & OpenMP Parallelism (OpenMP in a card, MPI among cards)
- Overlapping of MPI Communication and Computation
- Support Normal and Compressed Gauge
- Gauge is rearranged to Linear Access Pattern
- Clover Term is fused with Hopping Term
- Software Prefetch
- Streaming Store
- Linea Algebra in CG is Fused

Data Layout

Array of Structure of Array (AOSOA) Layout along X Direction

- X direction size must be multiple of 16 (even odd pre-conditioning is used)
- No constraints for other directions

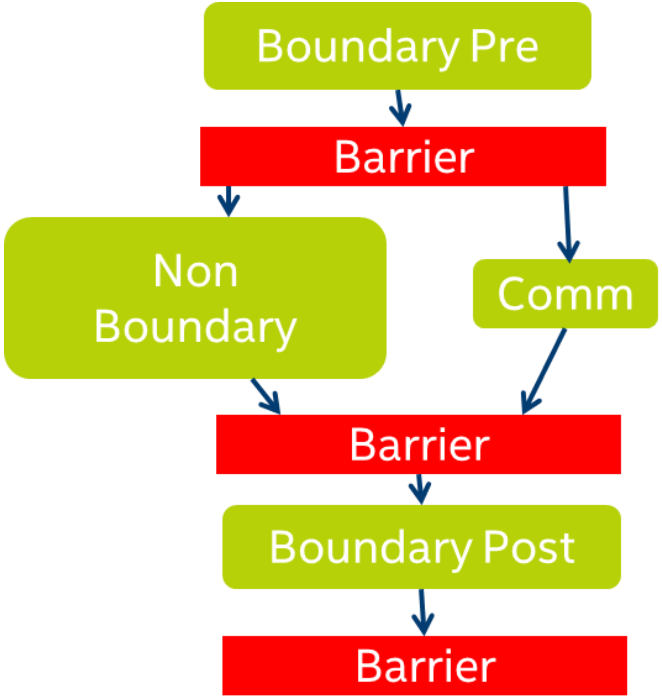
- Storage of Quark Fields
  - double SC[3][4][2][8]
- Storage of Gauge Fields
  - double su3[8][3][3][2][8]; (uncompressed)
  - double su3[8][3][2][2][8]; (compressed)

OpenMP & MPI Implementation of Hopping Term

3 OMP Synchronizations in Hopping Term

One Thread is dedicate for Communication

- Overlapped with Non Boundary Processing



Machine & Software Configuration

Element	Configuration
Host	Xeon-E5 2697 v3 (HSW) 2.6GHz 14core x 2socket
Coprocessor	Xeon Phi 7120A(1.238GHz, 61 core)
HCA	Mellanox FDR IB
MPSS	Version 3.3.3
Compiler	Intel Compiler 15.0.2
MPI	Intel MPI 5.0.3

Hopping Term Performance on 1 Cards

Lattice size	Normal Gauge	Compressed Gauge
32x32x32x12	75 GFLOPS (0.87)	86 GFLOPS (1.00)
32x32x32x24	80 GFLOPS (0.86)	93 GFLOPS (1.00)
32x32x32x32	80 GFLOPS (0.86)	93 GFLOPS (1.00)

Compressed Gauge increase Performance 14%

Software Prefetch Effect Evaluation on Hopping Term (Compressed Gauge)

Lattice Size	Without Prefetch	With Prefetch
32x32x32x12	63 GFLOPS (0.73)	86 GFLOPS (1.00)
32x32x32x24	69 GFLOPS (0.74)	93 GFLOPS (1.00)
32x32x32x32	69 GFLOPS (0.74)	93 GFLOPS (1.00)

Appropriate SW Prefetch increase Performance 26%

- KNC has Mem to L2 HW Prefetchter, but no L2 to L1 Prefetchter
- Stencil data is hard to predict the access pattern for HW Prefetcher

Streaming Store Effect Evaluation on Hopping Term (Compressed Gauge)

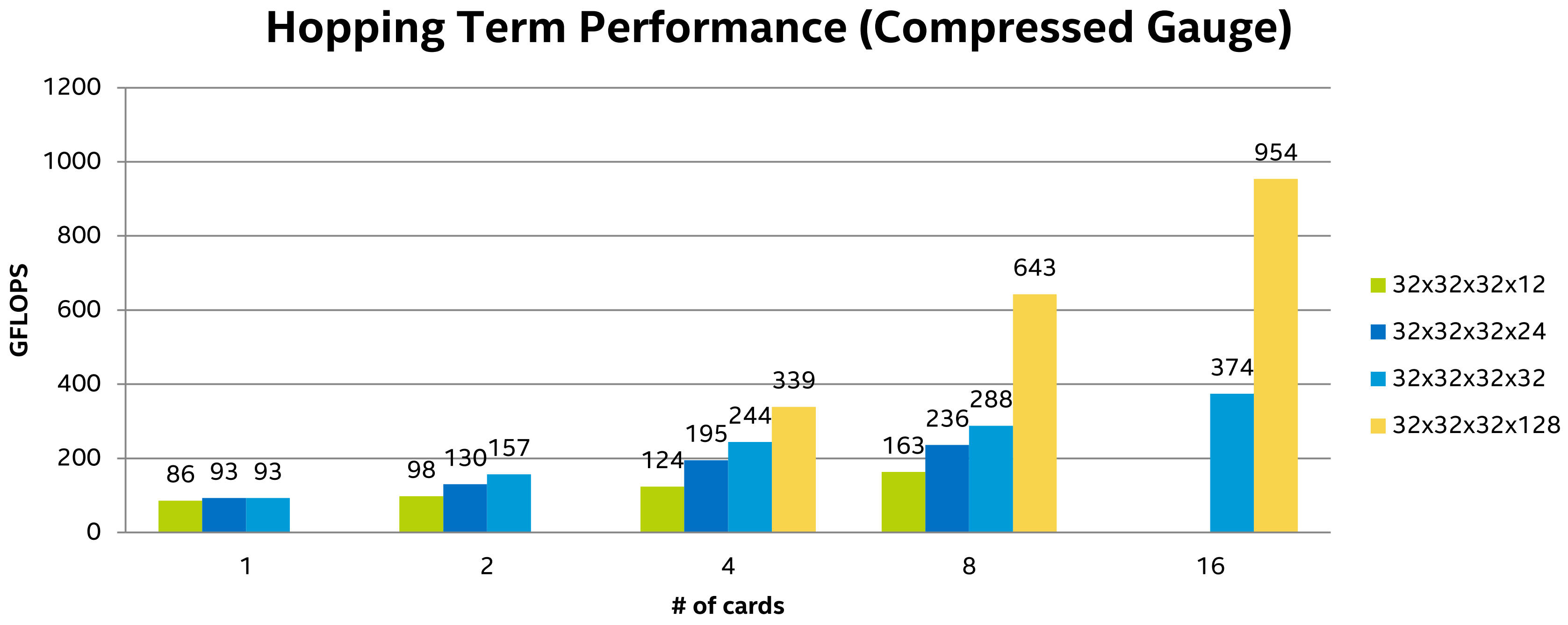
Lattice Size	Without SS	With SS
32x32x32x12	79 GFLOPS (0.91)	86 GFLOPS (1.00)
32x32x32x24	85 GFLOPS (0.91)	93 GFLOPS (1.00)
32x32x32x32	85 GFLOPS (0.91)	93 GFLOPS (1.00)

Streaming Store increase Performance 9%

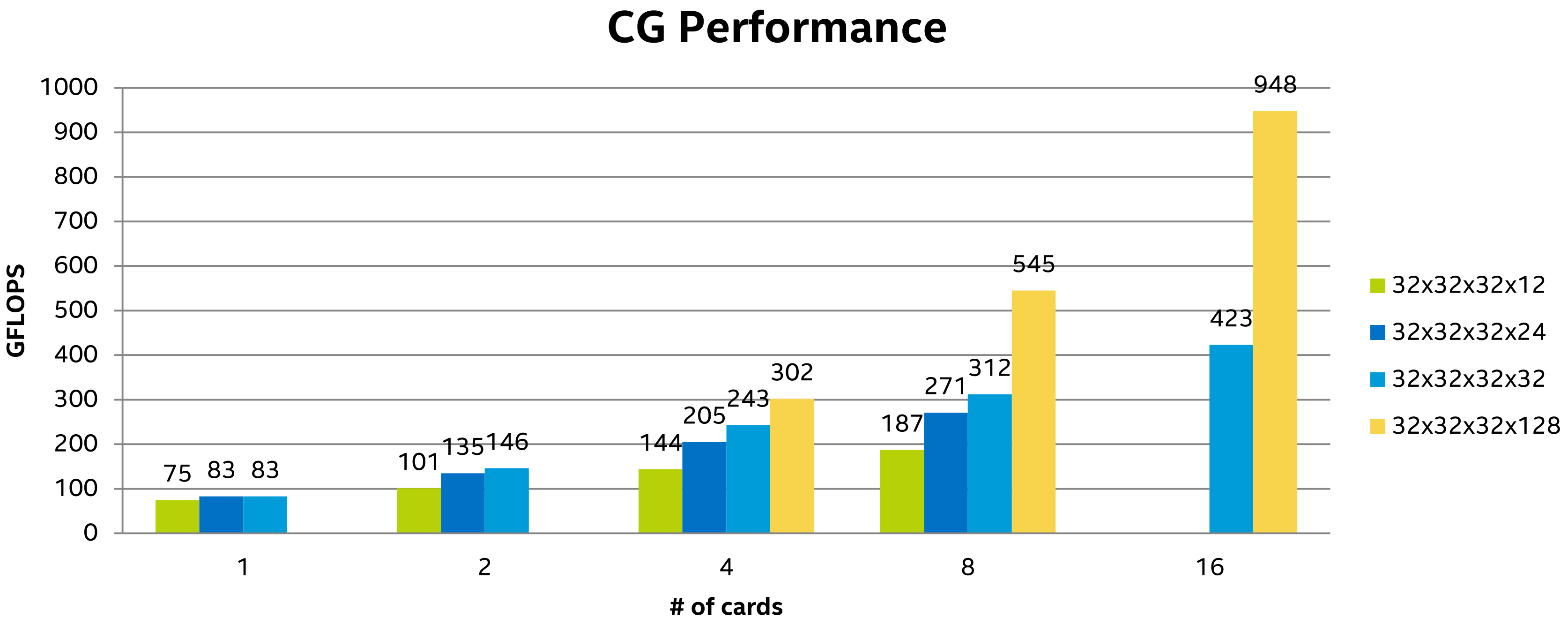
CG Performance on 1 Cards

Lattice Size	Normal Gauge	Compressed Gauge
32x32x32x12	68 GFLOPS	75 GFLOPS
32x32x32x24	75 GFLOPS	83 GFLOPS
32x32x32x32	77 GFLOPS	83 GFLOPS

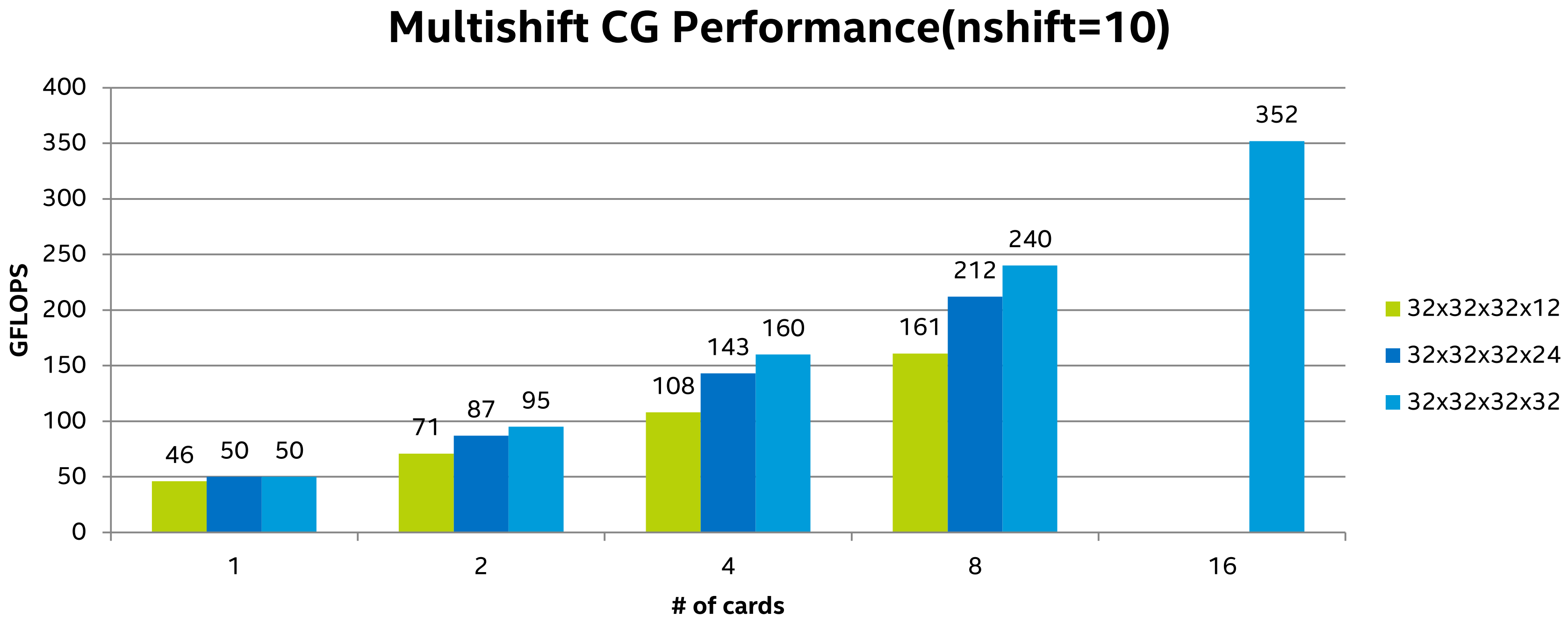
Hopping Term Performance on Multi-cards



CG Performance on Multi-cards



Multishift CG Performance



Conclusion

Our Implementation scales up to 16 KNC for 32x32x32x128 lattice size

- Scalability depends on lattice size.
  - Network bandwidth limit the small lattice size performance

SW Prefetch, Streaming Store and Compressed Gauge increase Performance

Multi shift CG scales in small lattice size

Optimization Notice

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

intel®

Software