THE UNIVERSITY of TENNESSEE
NATIONAL LEADERSHIP IN HPC

Performance Metrics and Application Experiences on a Cray CS300-AC™ Cluster Supercomputer Equipped with Intel® Xeon Phi™ Coprocessors

Vincent C. Betro, Ph.D.
Computational Scientist
National Institute for Computational Sciences/
Application Acceleration Center of Excellence
vbetro@tennessee.edu

R. Glenn Brook, Ph.D.
Chief Technology Officer
Director, Application Acceleration Center of Excellence
Joint Institute for Computational Sciences
glenn-brook@tennessee.edu

A NEW VINTAGE OF COMPUTING CUG 2013
Napa Valley, California • May 6–9

Outline

- Introduction to AACE/Beacon
- Introduction to the Intel Xeon Phi Coprocessor
- Discussion of codes ported and optimized for the Phi
  - ENZO
  - H3D
  - Boltzmann-BGK
  - Gyro
- Discussion of codes ported but not optimized for the Phi
- Conclusions

Application Acceleration Center of Excellence (AACE)

- Established by the University of Tennessee in 2011 alongside the National Institute for Computational Sciences (NICS) at Oak Ridge National Laboratory (ORNL)
- Tasked with exploring the application of emerging computing technologies to simulation in science and engineering
- *An essential element of a sustainable software infrastructure for simulation in science and engineering*

Strategic Engagement with Intel

- Multi-year agreement with Intel to jointly pursue:
  - Development of next-generation, high-performance computing solutions based on the Intel® Many Integrated Core (MIC) architecture
  - Design of scientific applications emphasizing a sustainable approach for both performance and productivity
- AACE receives early access to Intel technologies and provides application testing, performance results, and expert feedback to Intel
  - Help guide further development efforts by Intel
  - Prepare scientific community for next generation HPC technologies such as the Intel® Xeon Phi™ coprocessor

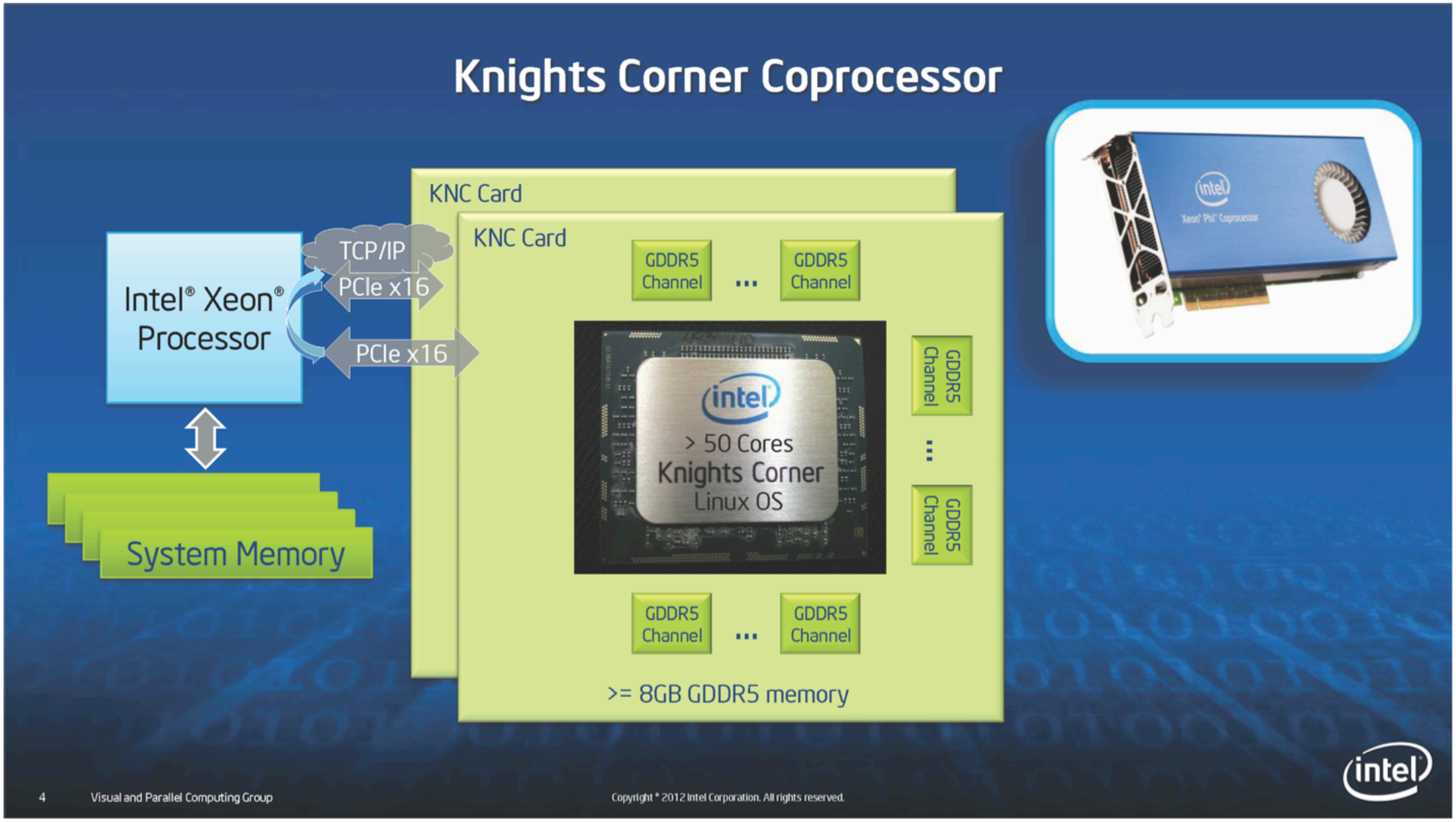
The Beacon Project

- Funded by NSF to port and optimize scientific codes to the Intel® Xeon Phi™ coprocessor
- State-funded expansion focuses on energy efficiency, big data applications, and industry
- Example Codes: PSC, H3D, OMEN, ENZO, MADNESS, NWCHEM, Amber, MILC, and MAGMA

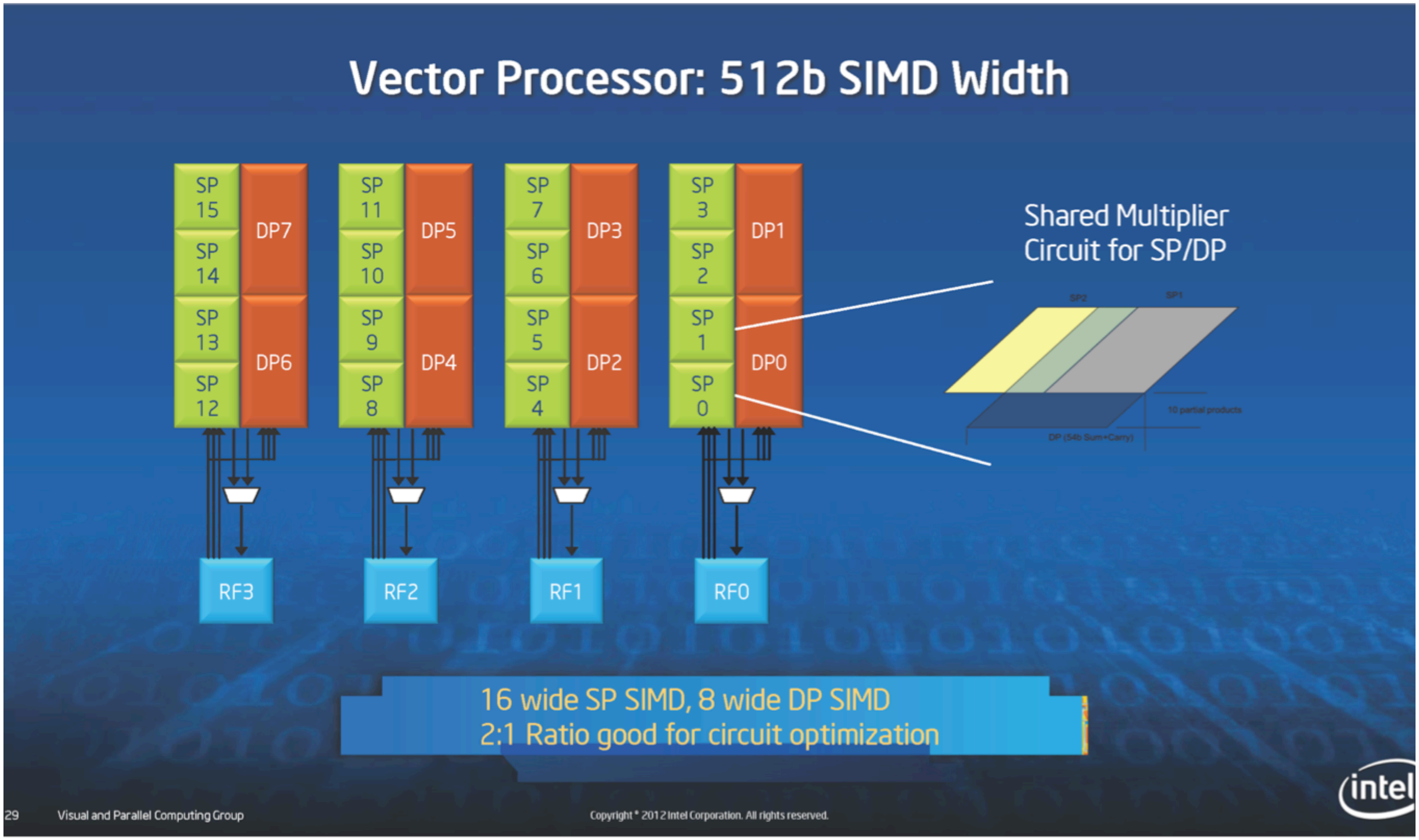
Beacon – Phase 1 Cray Xtreme-X Supercomputer	
Nodes	2 service, 16 compute
Interconnect	FDR IB Fat Tree
CPU model	Intel Xeon E5-2670
CPUs per node	2 8-core, 2.6 GHz
RAM per node	64 GB
SSD per node	80 GB
Intel® Xeon Phi™ coprocessors per node	2 x pre-production 50+ cores, 8 GB GDDR5 RAM

Beacon – Phase 2 Cray Xtreme-X Supercomputer Peak Performance: 210.1 TFLOP/s	
Nodes	4 service, 6 I/O, 48 compute
Interconnect	FDR IB Fat Tree
CPU model	Intel Xeon E5-2670
CPUs per node	2 8-core, 2.6GHz
RAM per node	256 GB
SSD per node	2 x 480 GB (compute), 16 x 300 GB (I/O)
Intel® Xeon Phi Coprocessors per node	4 x 5110P 60-core, 1.053GHz 8 GB GDDR5 RAM

The Intel Xeon Phi Coprocessor



Vectorization on the Intel Xeon Phi



Comparing Results from Intel Xeon Phi

In order to have an apples to apples comparison on the Xeon Phi, we need to consider the following:

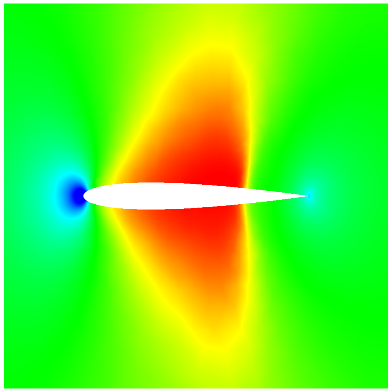
- Compare 2 Sandybridges to 1 Xeon Phi
    - 125W x 2 vs 255 W
    - 256 bit vector x 2 vs 512 bit vector
    - Run same code on both
- Doing some quick math, this means that we have similar power and vector processing, except that the Xeon is ~2.5 times the clock speed
- So, how do we make the Xeon Phi worth our while?
    - Many more threads
    - Programming versatility

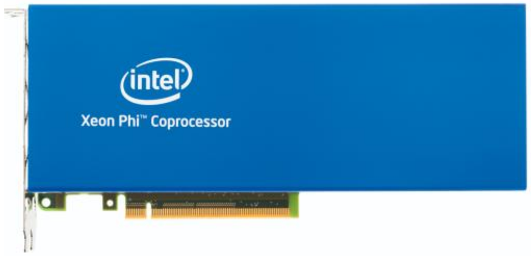
Intel Xeon Phi Programming Paradigms

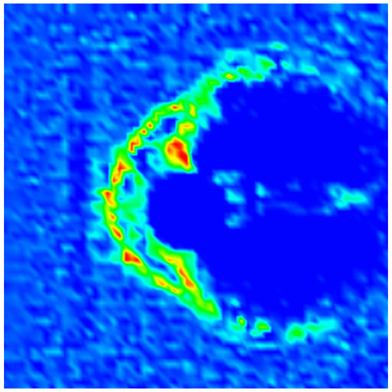
- Native mode
- Offload mode
- Heterogeneous Execution
- MPI/OpenMP/Hybrid

Codes Investigated by AACE

- Atmospheric modeling – HOMME-CAM (ported)
- Astrophysics – Enzo (ported and partially optimized)
- Computational Fluid Dynamics (CFD) – BGK-Boltzmann Solver (ported and optimized)
- Earthquake modeling – AWP-ODC (ported)
- Magnetospheric Physics – H3D (ported and partially optimized) and PSC (ported)
- Tokamak Plasmas – Gyro (ported)
- Agent Based Modeling – Transims and ASCAPE (ported)

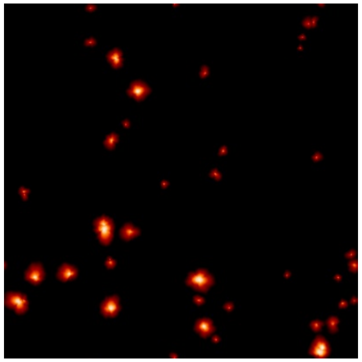


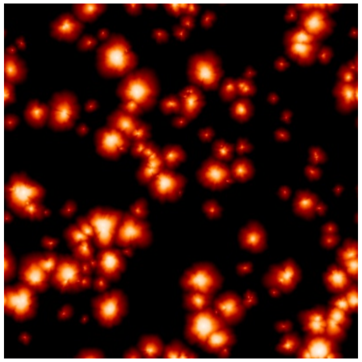


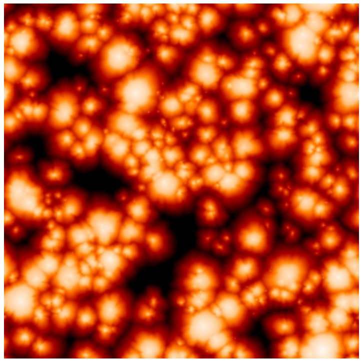


Enzo

- Community code for computational astrophysics and cosmology
- More than 1 million lines of code
- Uses powerful adaptive mesh refinement
- Highly vectorized with a hybrid MPI + OpenMP programming model
- Utilizes HDF5 and HYPRE libraries
- Multiple MPI tasks per coprocessor and many threads per MPI task



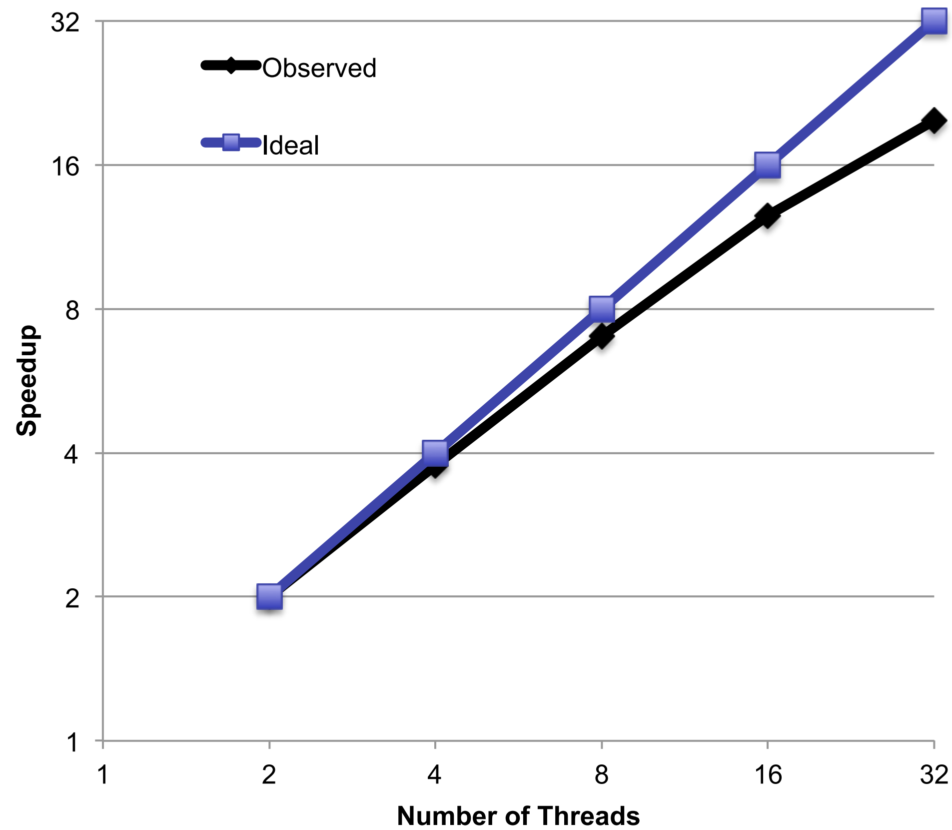




Enzo was ported and optimized for the the Intel® Xeon Phi™ Coprocessor by
Dr. Robert Harkness
harkness@sdsc.edu

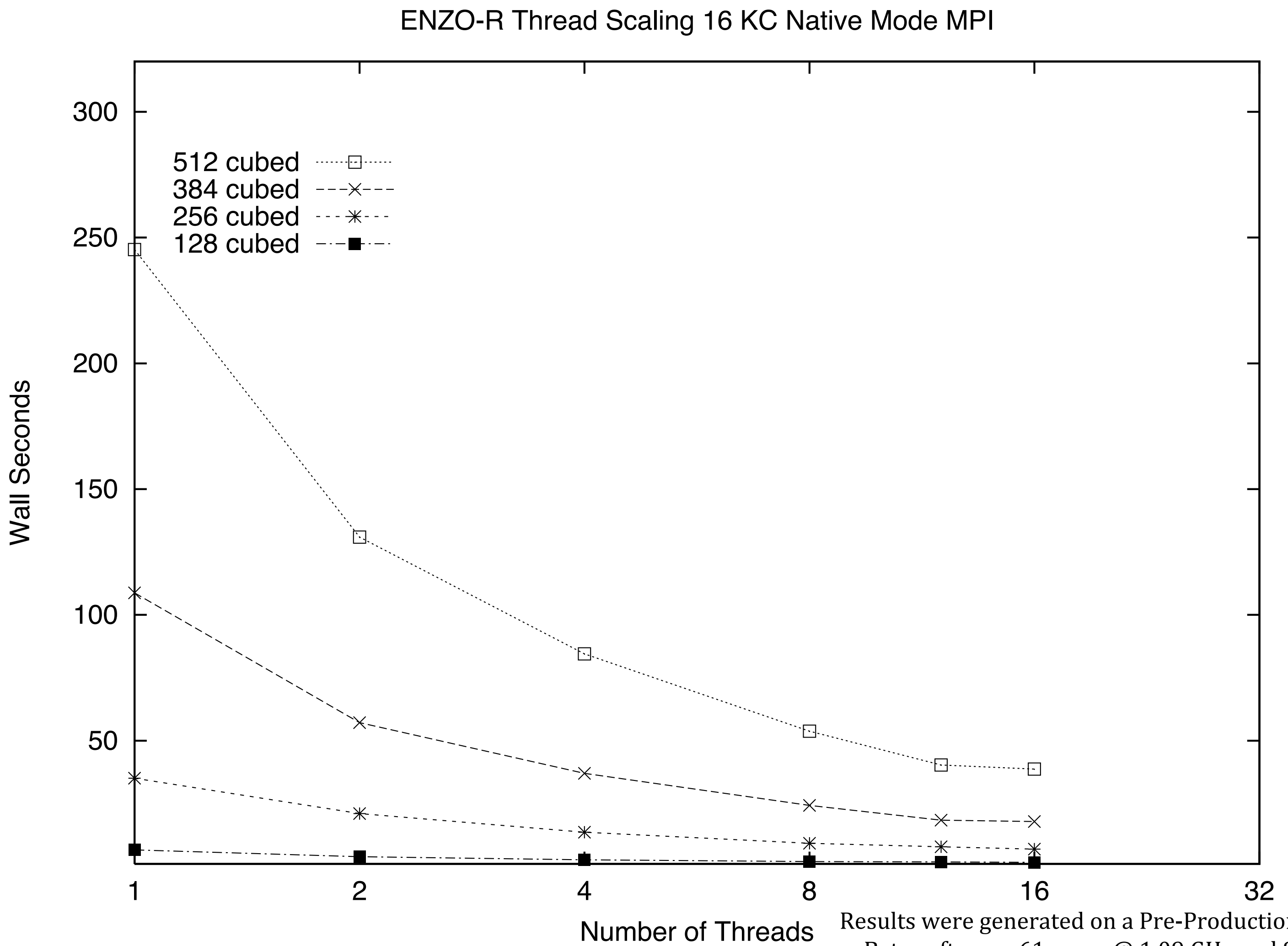
Preliminary Scaling Study: Native

- ENZO-C
- 128^3 mesh (non-AMR)
- pure MPI
- native mode
- Fortran, C, C++



Results were generated on the Intel® Knights Ferry software development platform

Multi-KNC Scaling Study: Native



- ENZO-R
- N^3 mesh
- Decomposed into 4x4x4 blocks
- Native mode on 16 KNC
- 4 MPI ranks per KNC
- 1,2,4,8,12,16 threads per rank

Results were generated on a Pre-Production Intel® Xeon Phi™ coprocessor
Beta software, 61 cores @ 1.09 GHz and 8 GB of GDDR5 RAM @ 2.75 GHz

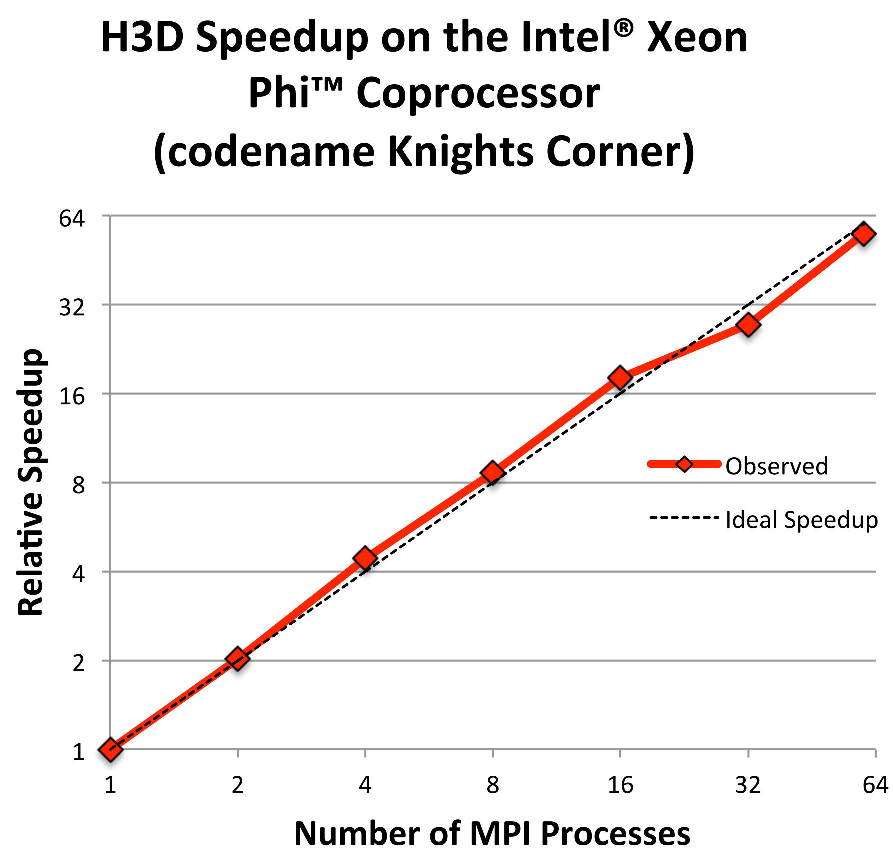
Hybrid3d (H3D)

- Provides breakthrough kinetic simulations of the Earth's magnetosphere
- Models the complex solar wind-magnetosphere interaction using both electron fluid and kinetic ions
  - Unlike magnetohydrodynamics (MHD), which completely ignores ion kinetic effects
- Contains the following HPC innovations:
  1. multi-zone (asynchronous) algorithm
  2. dynamic load balancing
  3. code adaptation and optimization to large number of cores

Hybrid3d (H3D) was provided for porting to the the Intel® Xeon Phi™ Coprocessor by
Dr. Homa Karimabadi
hkarimabadi@ucsd.edu

Hybrid3d (H3D) Performance

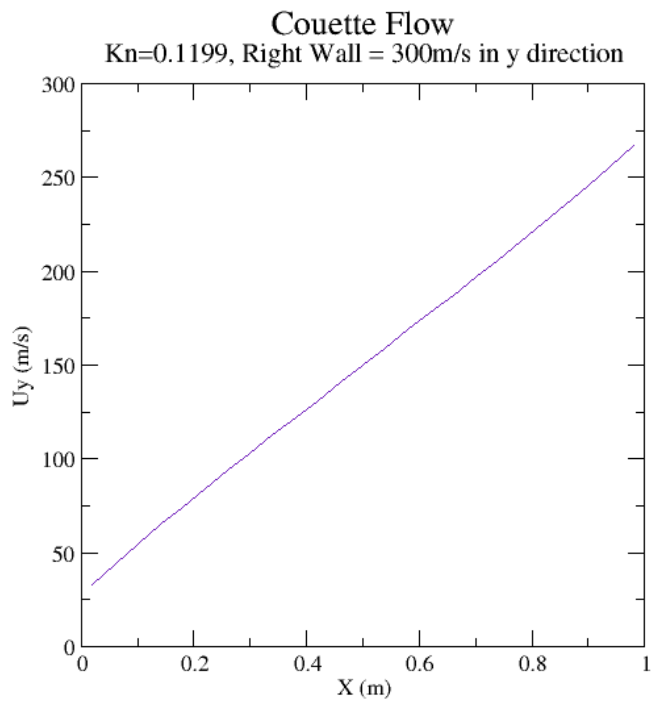
Optimizations were provided by Intel senior software engineer Rob Van der Wjingaart.

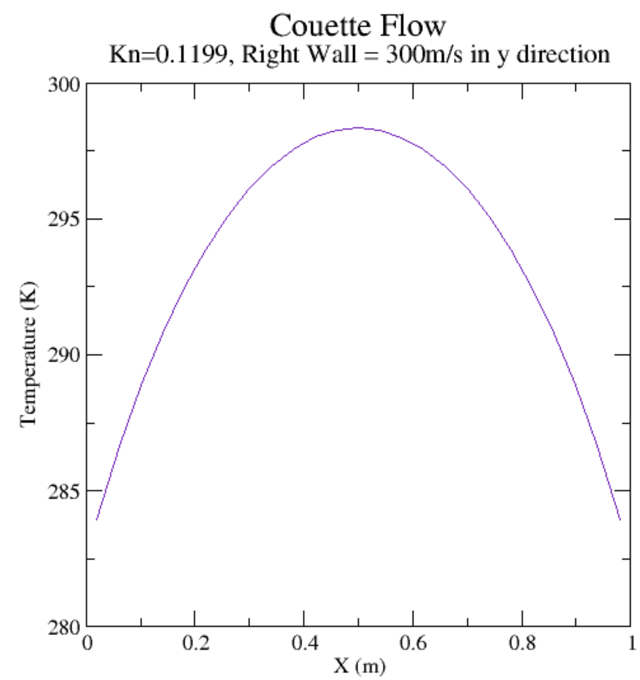


Results were generated on a Pre-Production Intel® Xeon Phi™ coprocessor with B0 HW and Beta SW
61 cores @ 1.09 GHz and 8 GB of GDDR5 RAM @ 2.75 GHz

Computational Fluid Dynamics (CFD)

Steady-state solution of a Couette flow using the Boltzmann equation with BGK collision approximation





The above CFD solvers were developed for the Intel® Xeon Phi™ Coprocessor by

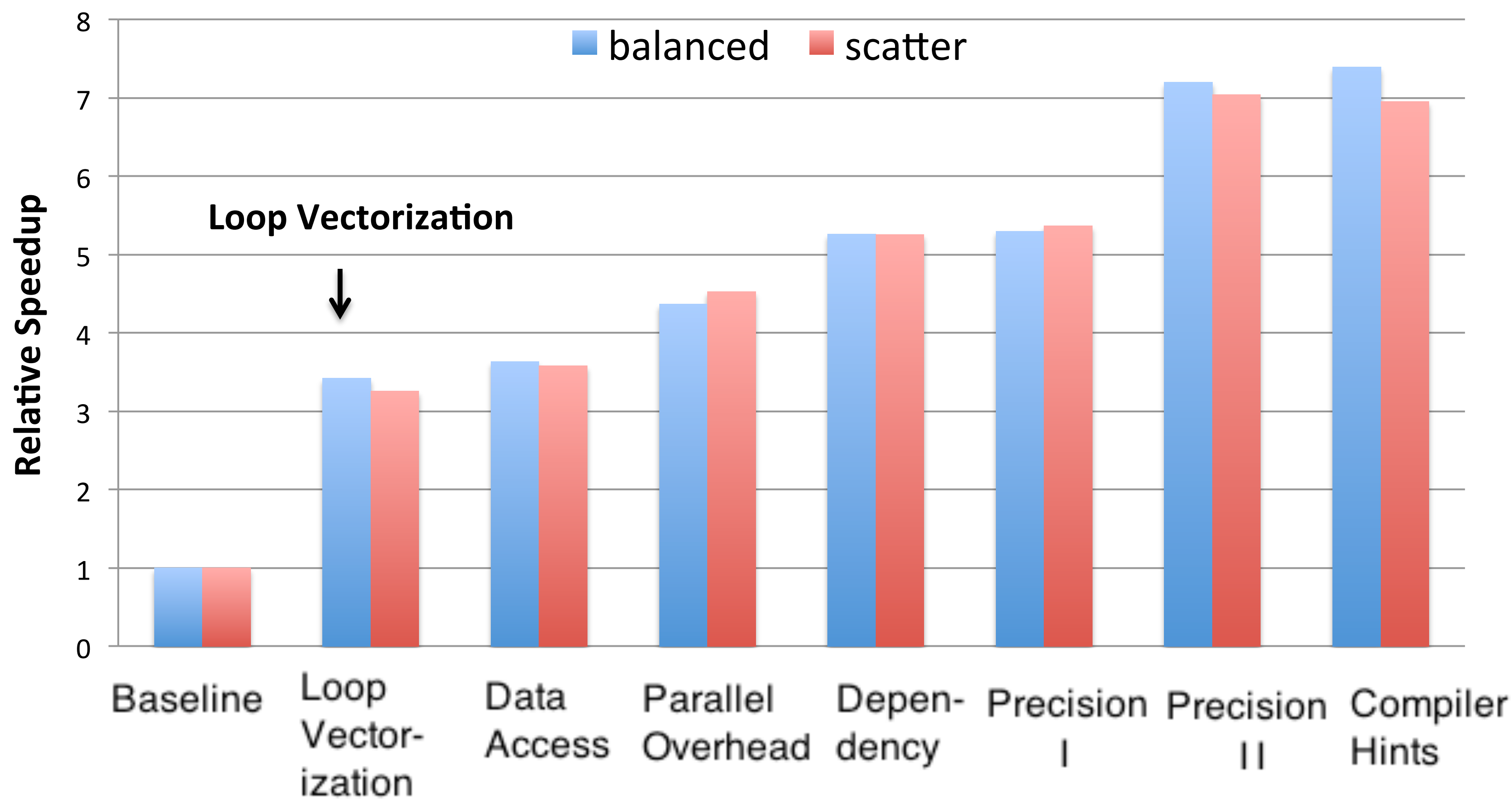
Ryan C. Hulguin
ryan-hulguin@tennessee.edu

Impact of Various Optimizations on the Model Boltzmann Equation Solver

- Optimized by Intel software engineer Rob Van der Wjingaart
- Base-line solver – all loops were vectorized except for one

- Set I — Loop Vectorization
  - Stack variable pulled out of the loop
  - Class member turned into a regular structure
- Set II — Data Access
  - Arrays linearized using macros
  - Align data for more efficient access
- Set III — Parallel Overhead
  - Reduce the number of parallel sections
- Set IV — Dependency
  - Remove reduction from computational loop by saving value into a private variable
- Set V — Precision
  - Use medium precision for math function calls (-fimf-precision=medium)
- Set VI — Precision
  - Use single precision constants and intrinsics
- Set VII — Compiler Hints
  - Use #pragma SIMD instead of #pragma IVDEP

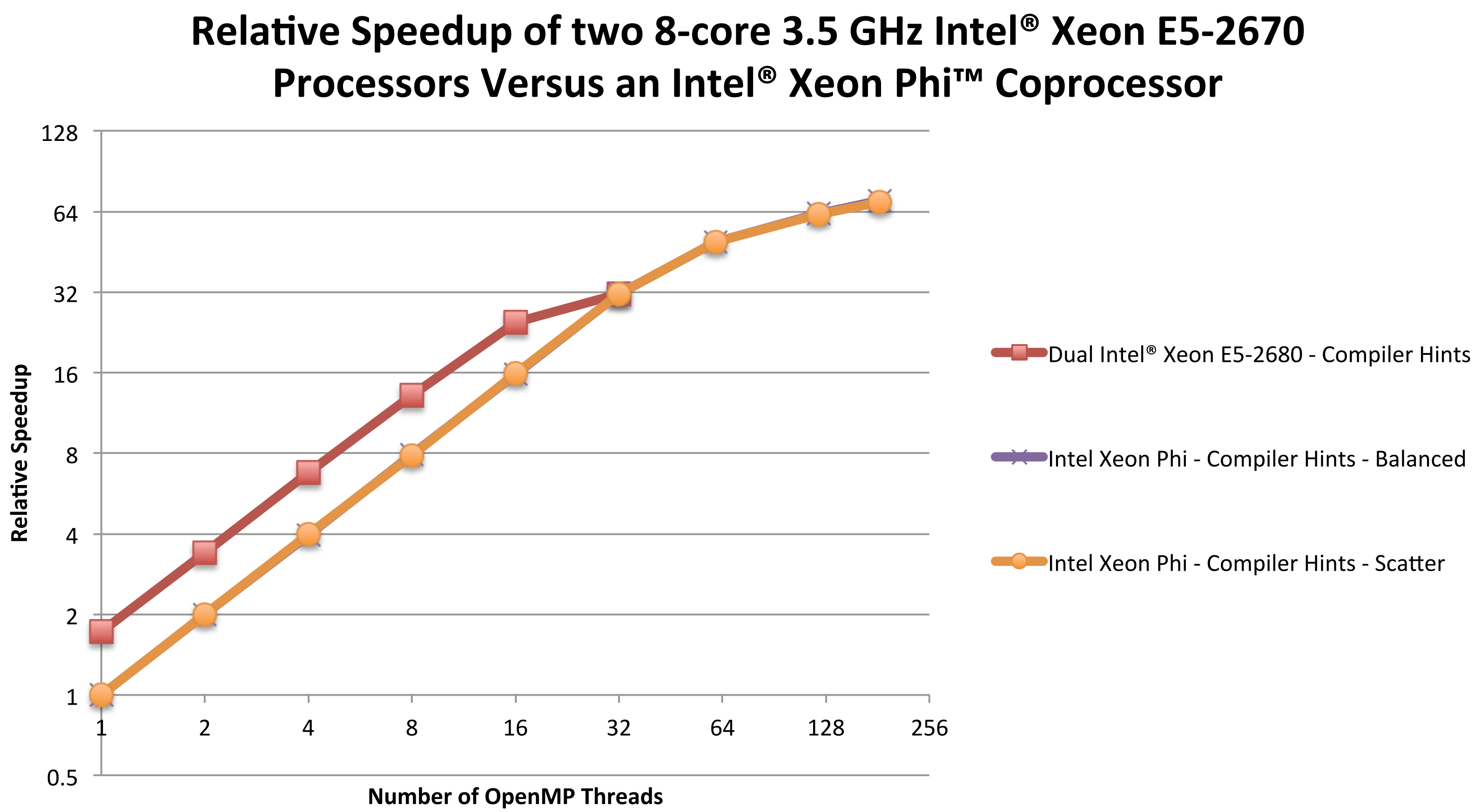
Optimization Results from the Model Boltzmann Equation Solver



Results were generated on a Pre-Production Intel® Xeon Phi™ coprocessor with B0 HW and Beta SW

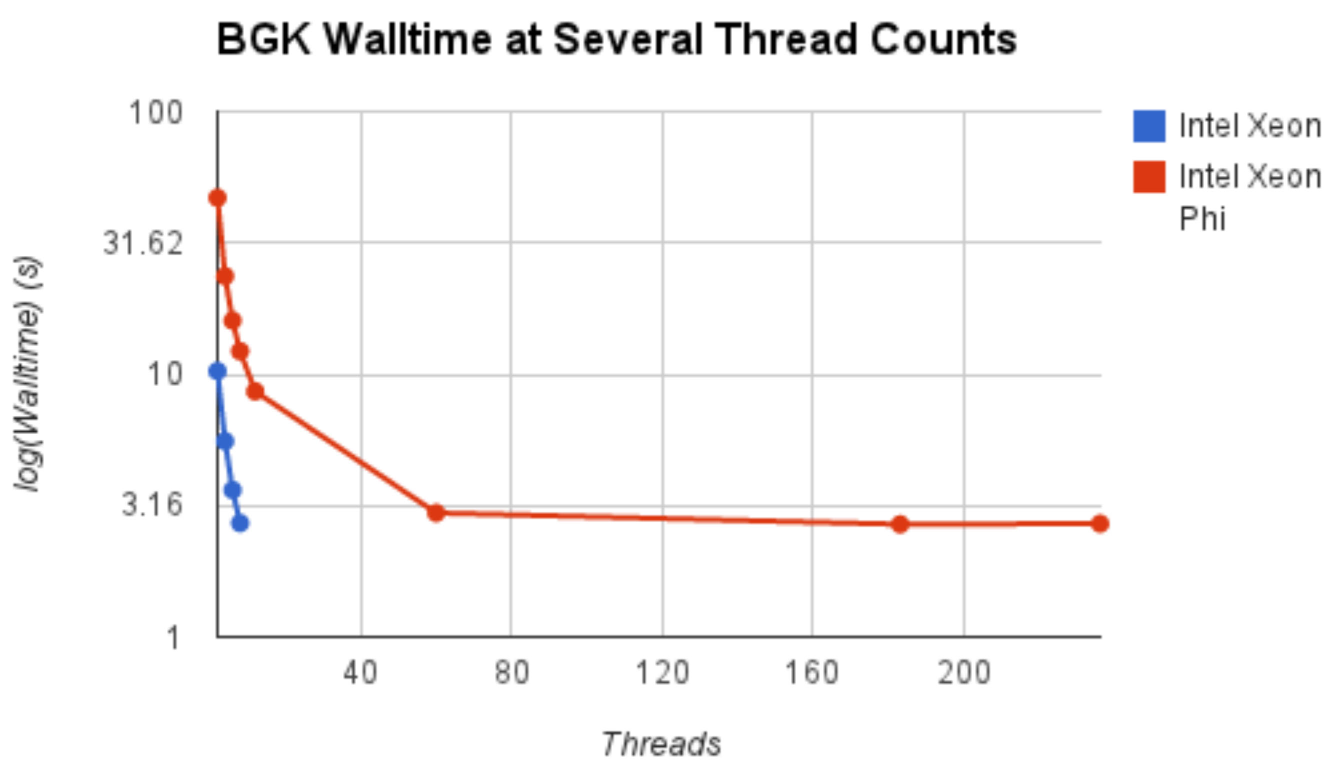
61 cores @ 1.09 GHz and 8 GB of GDDR5 RAM @ 2.75 GHz

Model Boltzmann Equation Solver Performance



Results were generated on a Pre-Production Intel® Xeon Phi™ coprocessor with B0 HW and Beta SW
61 cores @ 1.09 GHz and 8 GB of GDDR5 RAM @ 2.75 GHz

Model Boltzmann Equation Solver Performance: Another View



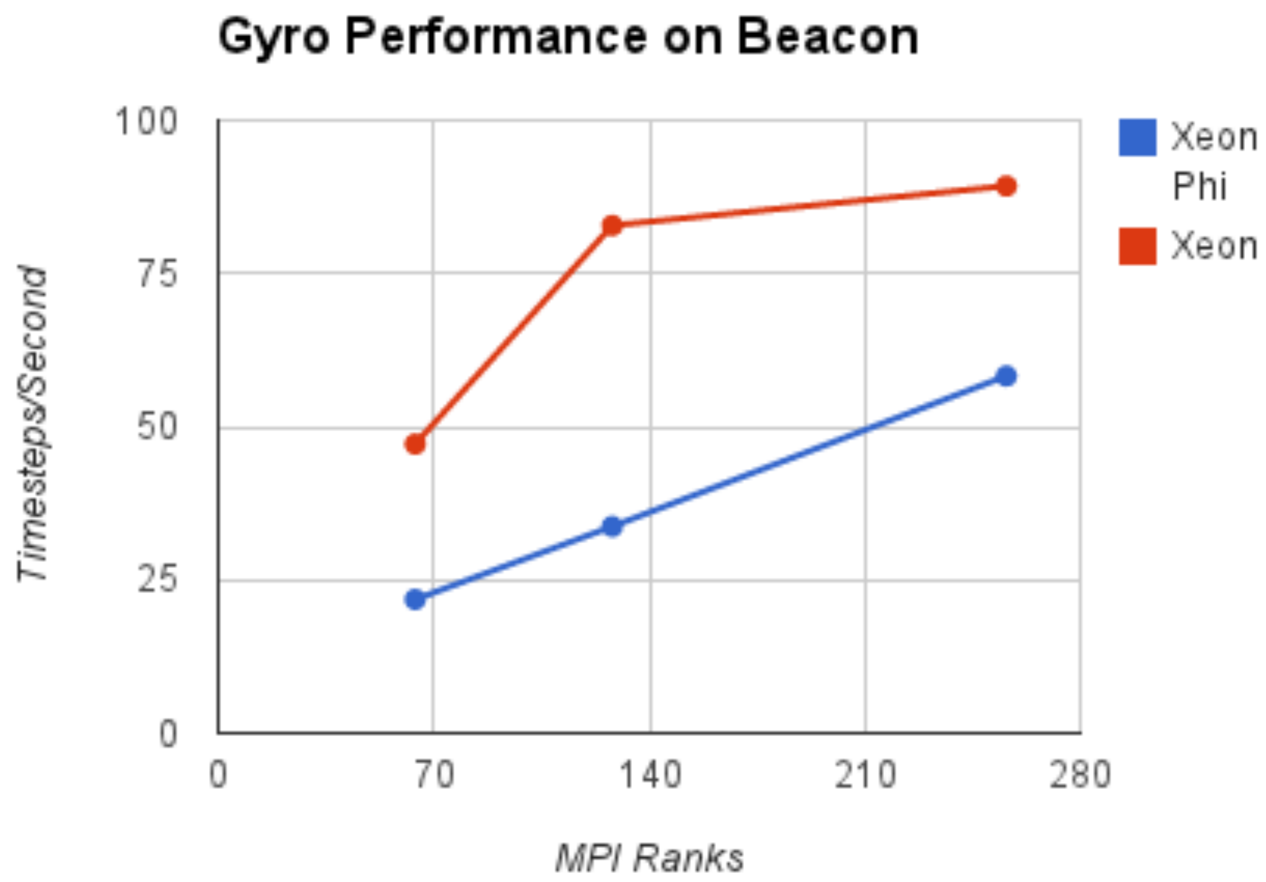
Results were generated on a 5100P Intel® Xeon Phi™ coprocessor with MPSS Gold
60 cores @ 1.053 GHz and 8 GB of GDDR5 RAM @ 2.75 GHz

Gyro: Tokamak Plasmas

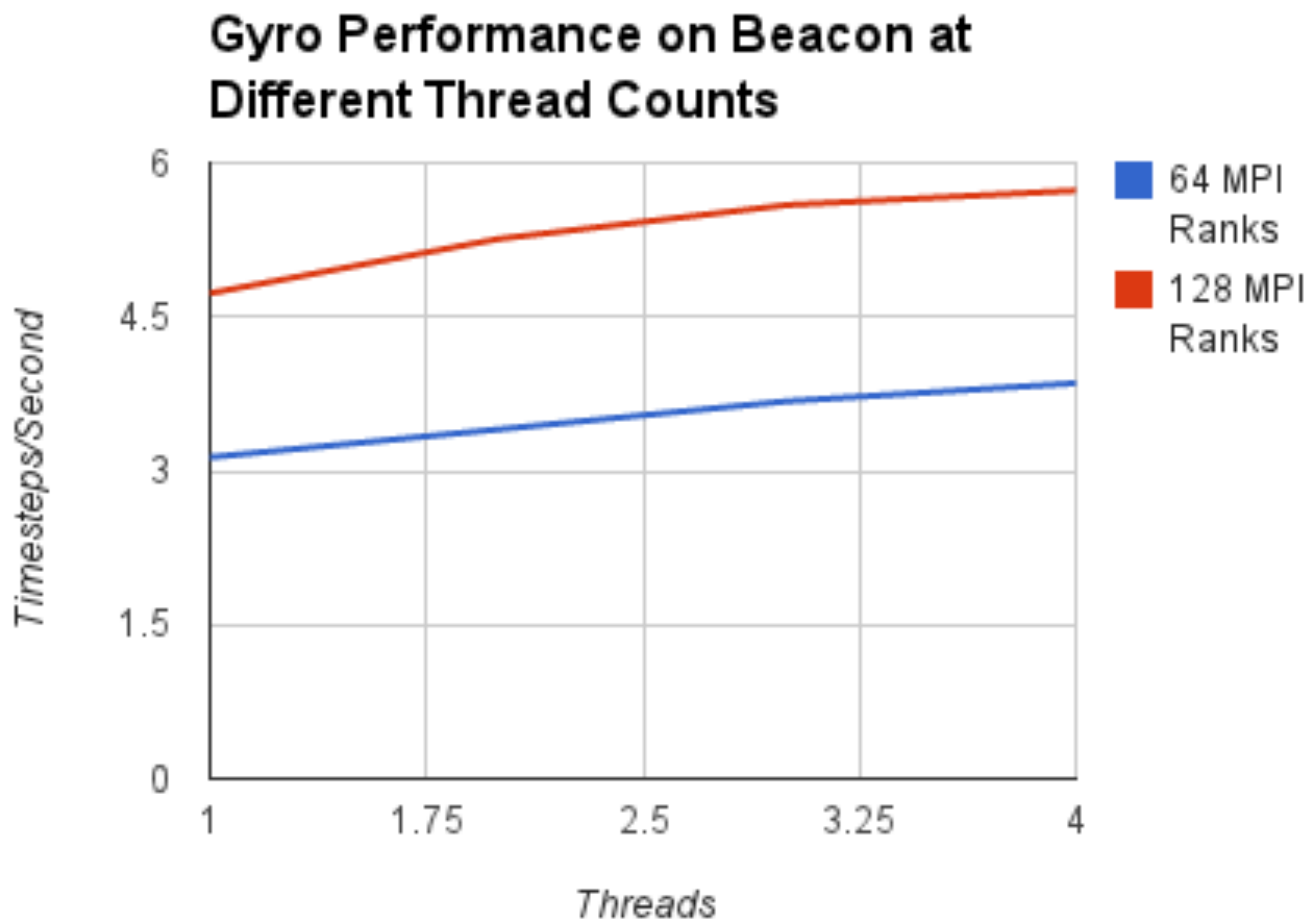
GYRO is a code for the numerical simulation of fusion tokamak microturbulence

- Computes the turbulent radial transport of particles and energy in tokamak plasmas
- Solves 5-D coupled time-dependent nonlinear gyrokinetic-Maxwell equations with gyrokinetic ions and electrons
- Developed by Jeff Candy and Ron Waltz at General Atomics
- GYRO can operate as a flux-tube (local) code, or as a global code, with electrostatic or electromagnetic fluctuations.
- Propagates system forward using a second-order, implicit-explict Runga-Kutta integrator and a fourth-order, explicit Eulerian algorithm
- Runs on a variety of machines: IBM Power, Cray XT and XE, SGI ICE and UV, Intel and Opteron Clusters

Gyro: MPI Performance



Gyro: Threaded Performance



Other codes ported to the Phi

Code	MPI Ranks	Walltime	Speed Up	% Peak
HOMME-CAM	32	82.65 s	—-	—-
HOMME-CAM	64	54.91 s	1.51	76%
PSC	16	894.41 s	—-	—-
PSC	32	679.58 s	1.32	66%
AWP-ODC	32	932 s	—-	—-
AWP-ODC	64	524 s	1.78	89%
AWP-ODC	128	287 s	3.25	81%
TRANSIMS	1	5904 s	serial	—-
ASCAPE	1	30.197 s	serial	—-

Other codes ported to the Phi

As one can see, while the Xeon Phi has a lower clock speed than the Xeon, it still exhibits speed up when running a scalable code.

However, exposing enough parallelism to beat the Xeon implementation requires some extra effort.

Conclusions

- Porting to the Intel Xeon Phi is simple
- Optimizing for the Xeon Phi takes some work, but pays dividends on all architectures
- The hybrid MPI/OpenMP model, along with vectorization, shows promise
- And….

WORLD RECORD!
"Beacon" at NICS

Intel® Xeon® + Intel Xeon Phi™
Cluster
First to Deliver
2.499 GigaFLOPS / Watt
71.4% HPL efficiency
#1 on current Green500

Open Call for Participation

- The call for NSF-supported time on Beacon is closed, but compute time on Beacon is still available at the discretion of the director.
- Individual researchers and teams are invited to submit proposals for projects investigating the impact of the Intel® Xeon Phi™ coprocessor on areas of interest to their particular field of study. Some possible areas of interest include (but are not limited to):
  - Computational modeling and simulation
  - Data analysis and visualization
  - Novel algorithms targeting the Intel® Xeon Phi™ coprocessor
  - Programming languages and tools
  - Debugging and profiling tools
  - Performance evaluation studies and tools
  - Energy-efficient computing
- Projects involving high-performance computing (HPC) applications in fields such as biology, economics, social sciences, and other non-traditional HPC domains are especially encouraged, as are projects associated with data-intensive computing and data-driven workflows.
- See http://www.jics.tennessee.edu/aace/beacon/open-call for details.

Acknowledgements

The authors would like to thank Mikhail Smelyanskiy, Karthikeyan Vaidyanathan, Ian Steiner, Joe Curley, Jim Jeffers, Pradeep Dubey, Susan Meredith, Rajesh Agny, and Russ Fromkin at Intel Labs, Alexander Heinecke from Technische Universitat Munchen, and John Lee, David Parks, Edgardo Evangelista, and Danny Tran at Cray, Inc. (formerly Appro) for their invaluable assistance in making Beacon's debut at the top of the November 2012 Green 500 a success. Additionally, the willingness of Intel's software and hardware engineers to work in concert with and listen to the needs of applications engineers at NICS and elsewhere has been invaluable. Finally, the authors wish to thank the National Science Foundation for their support of the Beacon project, along with the State of Tennessee.

Contact Information

Vincent C. Betro, Ph.D.

Computational Scientist

National Institute for Computational Sciences/

Application Acceleration Center of Excellence

vbetro@tennessee.edu

R. Glenn Brook, Ph.D.
Chief Technology Officer
Director, Application Acceleration Center of Excellence
Joint Institute for Computational Sciences
University of Tennessee & Oak Ridge National Laboratory
glenn-brook@tennessee.edu