Co-design of Large Scale Climate Data Analytics for Emerging Supercomputing Architectures

Forrest M. Hoffman[1], Sarat Sreepathi[1], Richard T. Mills[2] Jitendra Kumar[1], and William W. Hargrove[3]

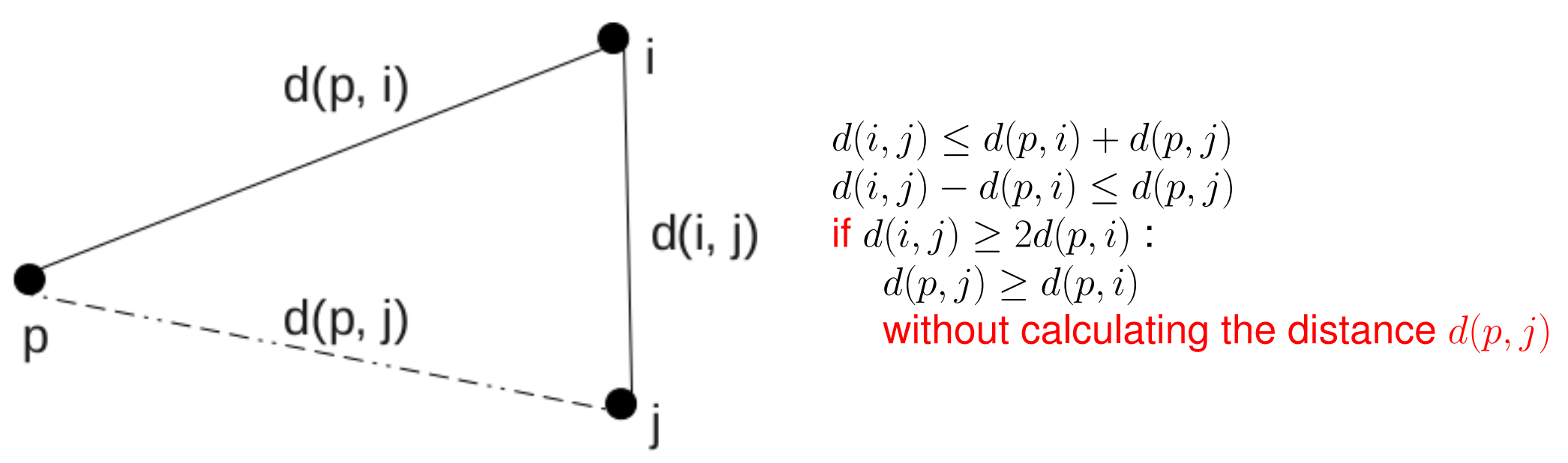
[1]Oak Ridge National Laboratory (ORNL) [2]Intel Corp. and [3]USDA Forest Service

Introduction

- Volumes of climate and other Earth science data are rapidly growing as model resolutions increase and observing networks and satellites collect data at higher spatial and temporal resolutions.
- New data analytics approaches are required on high performance computing platforms to synthesize and analyze these data.
- We examine some of these approaches and demonstrate their utility for climate, remotely-sensed vegetation phenology, and LiDAR data sets.
- Described here is a potential co-design effort to
  - develop and extract key analytics methods useful in climate research,
  - optimize these methods for existing Leadership Computing platforms using large climate data sets, and
  - develop benchmark problems for co-design of future data analytics platforms.

Accelerated *k*-means Clustering

- We have two implementations of accelerated *k*-means clustering, following two parallel programming models
  - A master-worker (MW) model: Central master assigns "aliquots" of work to workers. This model facilitates dynamic load balancing but has memory and performance scalability limits because of the single central process.
  - Fully distributed (FD): All processes use a static distribution of work. This model is very scalable, but has no dynamic load balancing.
- We "accelerate" the *k*-means process using two techniques described by Phillips (doi:10.1109/IGARSS.2002.1026202):
  - Use the triangle inequality to eliminate unnecessary point-to-centroid distance computations based on the previous cluster assignments and the new inter-centroid distances.
  - Reduce evaluation overhead by sorting inter-centroid distances so that new candidate centroids c_j are evaluated in order of their distance from the former centroid c_i. Once the critical distance $2d(p, c_i)$ is surpassed, no additional evaluations are needed, as the nearest centroid is known from a previous evaluation.



$d(i,j) \leq d(p,i) + d(p,j)$
$d(i,j) - d(p,i) \leq d(p,j)$
if $d(i,j) \geq 2d(p,i)$:
$d(p,j) \geq d(p,i)$
without calculating the distance $d(p,j)$

Figure 1: *The triangle inequality is used to eliminate unnecessary distance calculations.*

- We also improve cluster quality by moving or "warping" clusters that become empty to locations in data space where points that are farthest from their current cluster centroids reside.

Baseline Performance Characterization

We collected performance data with our clustering code for a baseline scenario using the LiDAR dataset for the Great Smoky Mountains National Park (GSMNP), 'tnnc_gsmnp_vertical_profiles' (1.7 GB).

- We utilized the Oxbow toolkit and Performance Analytics Data Store (PADS) infrastructure for this application characterization.
- This kind of data is invaluable to pursue effective co-design through objective assessment and aids in adaptation to emerging architectural features.

Computational Profiling

The computational profile of application execution is described by the mix of executed micro-operations. Figure 2 shows the instruction mix.

- Obtained by decoding the x86 assembler instructions and grouping them into coarser categories like memory, control, floating point and integer arithmetic.
- Obtained using a tool based on Intel's PIN, a dynamic binary instrumentation tool.
- The data is useful to ascertain if there is potential for improved performance. For instance, we identified an opportunity for improved performance by better utilization of floating point single-instruction-multiple-data (SIMD) operations (see **Improving computational intensity** discussion of BLAS level 2-3 formulation below).

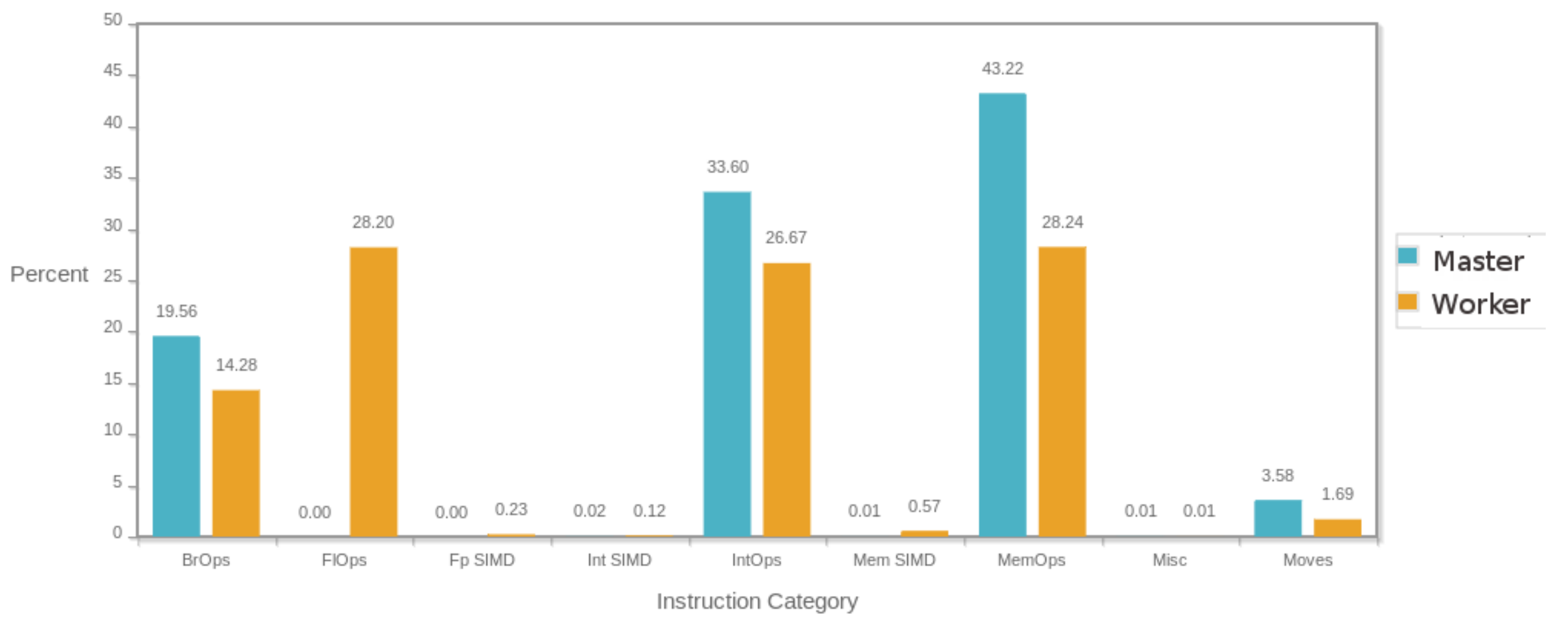


Figure 2: *The instruction mix for the clustering code running on 16 processors on a 1.7 GB dataset. The blue bar corresponds to the master process that primarily handles communication, explaining the lack of any floating point operations. The orange bar represents worker processes that exclusively handle the computation, as reflected in floating point operations.*

Communication Behavior

We used a communication profiling tool (mpiP) to capture the volume of data transferred between MPI ranks and visualized the results to understand the communication topology (Figure 3). It is clearly evident that we are using a master-worker protocol because all communication is point-to-point between the first process and all other processes.

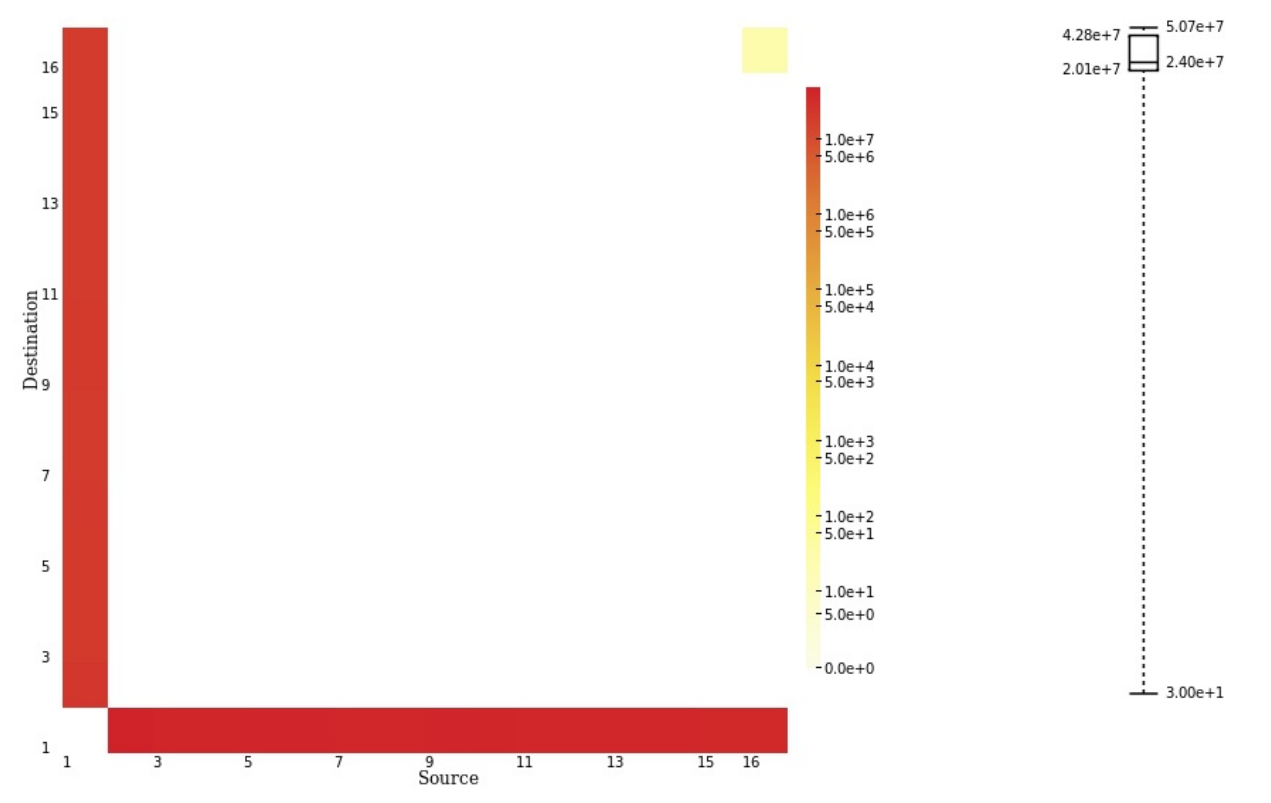


Figure 3: *Communication volume for clustering code using 16 MPI processes. The axes show the ranks of the sender and receiver process respectively. The box plot on the right shows the distribution of MPI message sizes. Note: the faint yellow box on the upper right of the plot is an artifact of the profiling tool.*

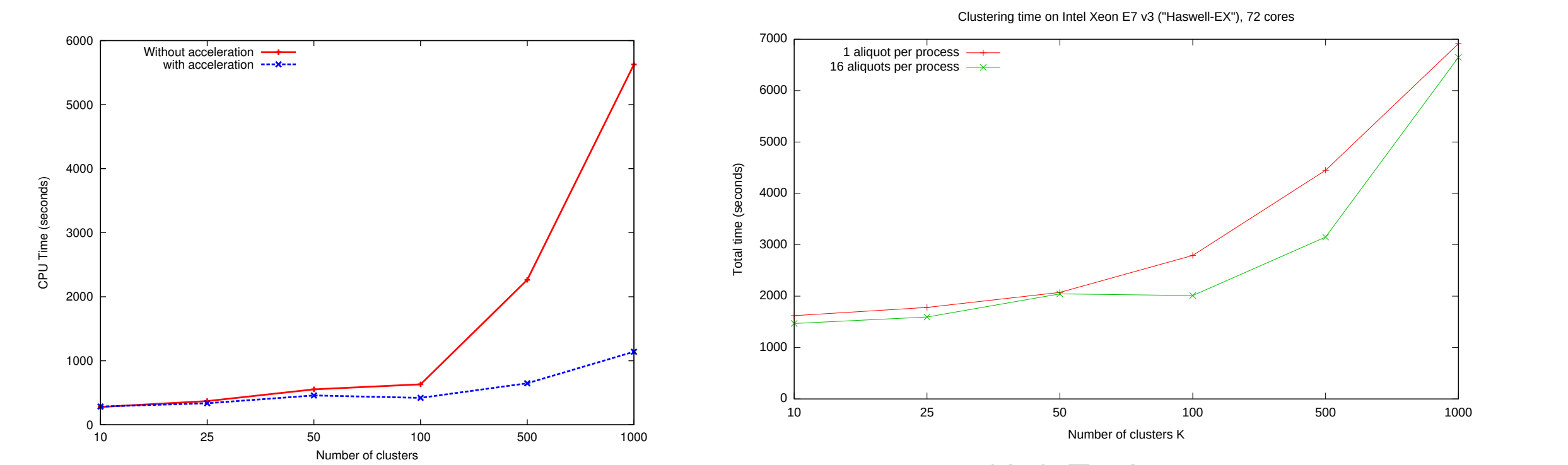
Memory Behavior

We instrumented the kernel of our application using PAPI hardware counters for obtaining detailed memory performance data. The kernel achieves a read bandwidth of 122 MB/s and a write bandwidth of 58.9 MB/s. These results are for the baseline code with no in-memory data rearrangement to optimize memory performance.

Parallel Performance

Accelerated *k*-means code

- In 2011, we used ∼1024 AMD Opteron cores on a machine like Jaguar, the Cray XT5 at ORNL, for our analyses.
- In 2015, we can do larger analyses on a single compute node of Intel's Endeavor cluster with Intel® Xeon® E7-8890 v3 ("Haswell-EX") processors.
  - AVX2 instruction set: 256-bit (8 single precision `float`s) vector registers with dual-issue fused multiply-add
  - Four 18 core (36 thread) CPUs; over 500 GB DRAM



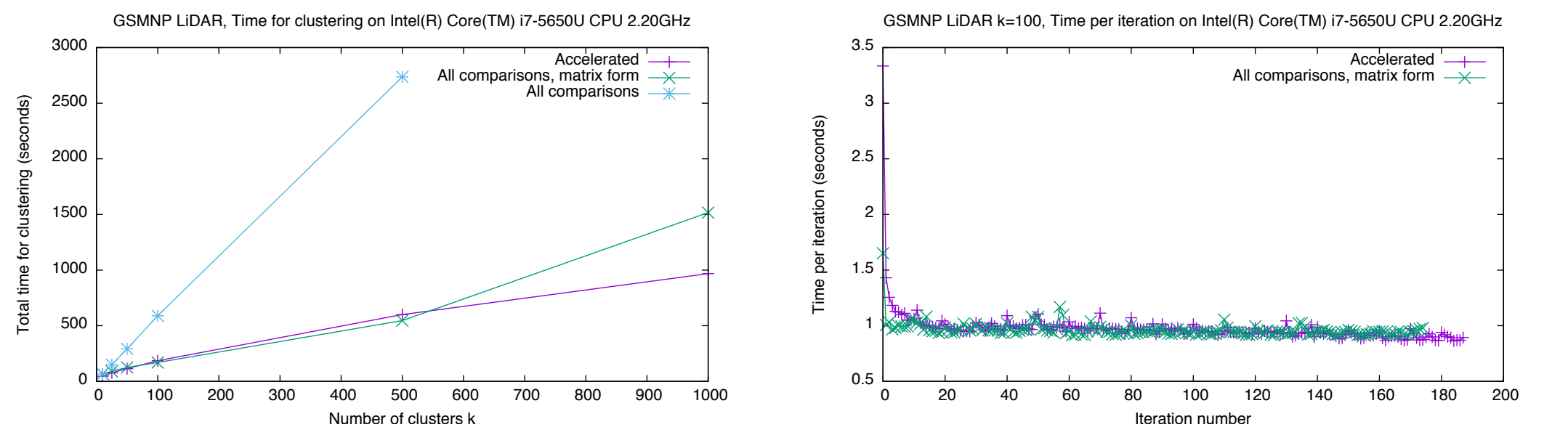
(a) Jaguar (b) Endeavor

Figure 4: *Times to cluster different versions of the 2000–2009 ForWarn phenology data set on (a) 1024 cores of the Jaguar Cray XT5, ca. 2011 at ORNL and (b) a single 72-core "Haswell-EX" node on Intel's Endeavor cluster. The data set used on Jaguar is the 16 day product, while the one on Endeavor is the 8 day product and is therefore twice as large (251 GB in single precision).*

- With acceleration, an equal distribution of observation vectors among processes does not guarantee load balance. Figure 4b illustrates the benefit of using smaller aliquots to enable dynamic load balancing in the MW clustering code.

Improving computational intensity

- We have recently realized that it is possible to achieve greater computational intensity of the observation–centroid distance calculations by expressing the calculation in matrix form:
  - For observation vector x_i and centroid vector z_j, the squared distance between them is $D_{ij} = \|x_i - z_j\|^2$.
  - Via binomial expansion, $D_{ij} = \|x_i\|^2 + \|z_j\|^2 - 2x_i \cdot z_j$
  - The matrix of squared distances can thus be expressed as $D = \overline{x}\mathbf{1}^\intercal + \mathbf{1}\overline{z}^\intercal - 2X^\intercal Z$, where X and Z are matrices of observations and centroids, respectively, stored in columns, $\overline{x}$ and $\overline{z}$ are vectors of the sum of squares of the columns of X and Z, and $\mathbf{1}$ is a vector of all 1s.
- The above expression for D can be calculated in terms of a level-3 BLAS operation (xGEMM), followed by two rank-one updates (xGER, a level-2 operation).
- Level 2 and 3 BLAS operations admit very computationally efficient implementations, and libraries such as Intel® MKL provide highly optimized versions.
- We have experimented with using the above, matrix formulation for the distance calculations and have found that it is dramatically faster than the straightforward loop over vector distance calculations when many distance comparisons must be made.
- For architectures that employ a high level of fine-grained parallelism with wide SIMD lanes, increasing the computation intensity has an especially high payoff in terms of improved performance. See Figure 6.
- Using the matrix formulation for distance comparisons in early *k*-means iterations is straightforward; a more complicated approach we will explore is using the matrix formulation in combination with the acceleration techniques described above, in which only a subset of observation–centroid distances are calculated.



(a) (b)

Figure 5: *Timings for clustering the GSMNP LiDAR dataset using a single worker process on an Intel® Core™ i7-5650U CPU operating at 2.20 GHz. (a) Total timings for k-means clustering using the acceleration techniques; doing all distance comparisons but forming the distance matrix using BLAS operations provided by Intel® MKL; and doing all distance comparisons without the benefit of the matrix formulation and BLAS. (b) Timings per iteration for k=100 when using the acceleration technique compared to the matrix formulation for the distance calculations. In early iterations, where many distance comparisons are required, the matrix formulation offers better performance.*

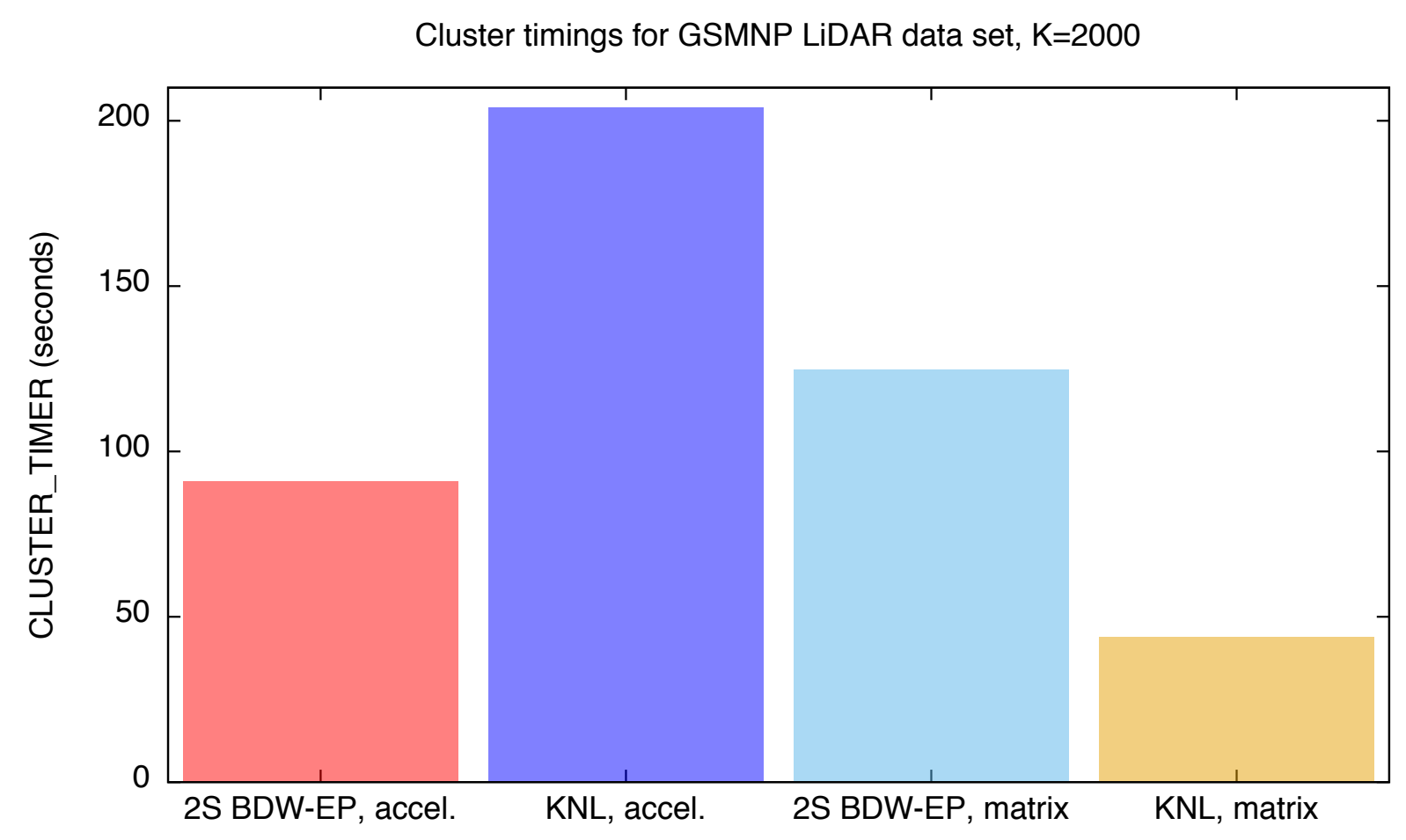


Figure 6: *A comparison of the matrix formulation and the "acceleration"-based approach on a recent high-end Intel® Xeon® server-based machine (dual-socket E5-2697 v4 "Broadwell-EP", referred to as "2S BDW-EP") and a second-generation Intel® Xeon Phi™ 7250 node (68 core "Knights Landing", referred to as "KNL"), clustering the GSMNP LiDAR dataset. The original "accelerated" formulation used by the code operates vector-by-vector and does not fully make use of the wide SIMD units on KNL. When employing the matrix formulation, however, KNL is so fast that, even at high values of k (2000 in this case), relying entirely on it without switching to the accelerated version gives the fastest performance out of all the cases, even though this means doing far more distance calculations than in the accelerated case. On the 2D BDW-EP system, the matrix formulation also results in much better use of the SIMD lanes (which have half the width of the KNL ones), but not so much as to make relying entirely on the matrix formulation most effective. (Note that relying on a vector-by-vector approach without the acceleration technique is so slow that we do not present timings for this case.)*

Future Work

- Optimize process and thread mapping in hybrid MPI-OpenMP mode
- Co-design for deep memory hierarchies

  Emerging supercomputing platforms are expected to have deeper memory levels and more diversity in memory technologies (volatile and non-volatile).
  - Identify and map suitable data structures in high bandwidth memory (e.g., Multi-channel DRAM in Xeon Phi™ KNL).
  - Develop techniques to effectively utilize non-volatile memory in contrast to traditional memory (e.g., Burst-buffers).
  - Understand trade-offs of various clustering modes (e.g., sub-NUMA clustering) for Xeon Phi™.

Conclusions

- Data analytics methods like those described above are increasingly important for climate-related studies and the growing body of Earth science data.
- We have shown that choices of optimal solution method are sensitive to memory bandwidth, cache sizes, and core clock speeds in addition to problem size and other problem-specific parameters.
- Standalone implementations of key analytics algorithms and benchmark problems could inform the design of future Leadership Computing platforms and software design necessary to meet the needs of climate researchers.

Acknowledgments

U.S. DEPARTMENT OF ENERGY
Office of Science

This research was supported by the Office of Biological and Environmental Research in the U.S. Department of Energy Office of Science and the U.S. Department of Agriculture Forest Service. This research used resources of the Oak Ridge Leadership Computing Facility at Oak Ridge National Laboratory, which is managed by UT-Battelle, LLC, for the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.

Intel Notices and Disclaimers

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.

The products and services described may contain defects or errors known as errata which may cause deviations from published specifications. Current characterized errata are available on request.

Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or go to: http://www.intel.com/design/literature.htm

Intel, Intel Xeon, Intel Xeon Phi™ are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States or other countries. *Other brands and names may be claimed as the property of others.

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.

Want a copy of this poster to read later?
Scan this QR code with your smartphone!

Advancing Cross-cutting Ideas for Computational Climate Science (AXICCS 2016) Workshop (September 12–13, 2016), Hilton Washington DC/Rockville Hotel & Executive Meeting Center, Rockville, Maryland, USA

Oak Ridge National Laboratory (ORNL) is managed by UT-Battelle, LLC, for the U.S. Department of Energy under contract DE-AC005-00OR22725.