

Performance Evaluation of the Cray XT3

Configured with Dual Core Opteron Processors

Richard F. Barrett
(865) 241-1512
rbarrett@ornl.gov

Sadaf R. Alam
(865) 241-1533
alamr@ornl.gov

Jeffrey S. Vetter
(865) 356-1649
vetter@ornl.gov

Oak Ridge National Laboratory
One Bethel Valley Lane, MS 6173
Oak Ridge, TN 37830

Categories and Subject Descriptors B.8.2 [Performance and Reliability] Performance Analysis and Design Aids, and D.1.3 [Programming techniques] Concurrent programming.

General Terms Algorithms, Measurement, Performance.

Keywords Performance characterization, multi-core processor, AMD Opteron, micro-benchmarking, scientific applications.

1. Introduction

The move by major microprocessor vendors toward processors containing multiple homogeneous processor cores is arguably the most important trend in contemporary computer architectures. The fundamental question for HPC scientific computing is whether multiple cores per processor can provide performance commensurate with expectations. Although a hybrid programming model that uses threads for parallelism within a node (e.g. using OpenMP) and message-passing for parallelism among nodes (most often using MPI) is often proposed as the best way to use systems with multi-core processors, the traditional MPI programming model is likely to remain important for portability reasons and the huge base of existing scientific applications.

Previously, we reported on the performance of micro-benchmarks and scientific application on the Cray XT3 system as configured with single core processors [S.R. Alam '07]. That platform, located at Oak Ridge National Laboratory (ORNL), has recently been upgraded to dual-core processors. It provides the opportunity to reevaluate this large-scale architecture, now with 50 TFLOPS of peak performance. In particular, we evaluate the performance of peta-scale targeted applications from scientific fields including astrophysics, climate, combustion, fusion, and materials science. In addition to runtime performance statistics and characteristics of these applications, we present micro-kernel, computational kernel, and inter-process communication benchmark results that aid in understanding the performance of these particular applications as we extend our observations beyond them.

2. Cray XT3 System Overview

The XT3 is Cray's third-generation massively parallel processing

system. It follows a similar design to the successful Cray T3D and T3E systems. The XT3 installed at ORNL uses a single processor node, or processing element (PE) consisting of two processor cores.

2.1 Processing Elements

Each XT3 PE has one Opteron processor with its own dedicated memory and communication resource. The compute nodes each consist of two 2.6 GHz dual-core AMD Opteron processors, 4 GBytes of memory (shared by the two cores) and 1 MByte of L2 cache (dedicated to each core). The 5,212 compute PEs run a lightweight operating system kernel called Catamount. The relatively few service nodes run SuSE Linux.

2.2 Interconnect

Each Opteron processor is directly connected to the XT3 interconnect via a Cray SeaStar chip. This SeaStar chip is a routing and communications chip and acts as the gateway to the XT3's high-bandwidth, low-latency interconnect. The PE is connected to the SeaStar chip with a 6.4 GB/s HT link. SeaStar provides six high-speed network links to connect to neighbors in a 3D mesh topology. Each of the six links has a peak bandwidth of 7.6 GB/s with sustained bandwidth of around 4 GB/s according to Cray. However, the SeaStar implementation in the ORNL XT3 limits the rate at which the Opteron can inject data onto the interconnect. In the XT3, the interconnect carries all message passing traffic as well as I/O traffic to the system's Lustre parallel file system.

2.3 Software

The XT3 uses a lightweight kernel operating system on its compute PEs, a user-space communications library, and a hierarchical approach for scalable application start-up. For scalability and performance predictability, each instance of the Catamount kernel runs only one single-threaded process and does not provide services like demand-paged virtual memory that could cause unpredictable performance behavior. Unlike the compute PEs, service PEs (i.e., login, I/O, network, and system PEs) run a full SuSE Linux distribution to provide a familiar and powerful environment for application development and for hosting system and performance tools.

Optimized compilers for Fortran, C and C++ are provided by the Portland Group. In addition, cross-compile gnu compiler suite is also available.

The XT3 uses the Portals data movement layer for flexible, low-overhead inter-node communication. Portals provide

connectionless, reliable, in-order delivery of messages between processes. For high performance and to avoid unpredictable changes in the kernel's memory footprint, Portals deliver data from a sending process' user space to the receiving process' user space without kernel buffering. Portals support both one-sided and two-sided communication models.

Cray provides a Message Passing Interface (MPI) communication library based on MPICH version 1.2 that uses Portals for data transfer. We used this implementation in all of our parallel benchmark and application experiments on the XT3.

The primary math library on the XT3 is the AMD Core Math Library (ACML). It incorporates BLAS, LAPACK and FFT routines, and is optimized for high performance on AMD-based platforms

3. Evaluation overview

The focus of this poster is on the performance effects of the dual core processor based machine, in comparison to single core processors, so our work is designed to expose those effects. The processor on the XT3 can be used in either single- or dual- core mode simply by adding a flag to execution launch.

We examine and report on the performance of applications from science areas such as fusion energy, climate modeling, and biology. Only the results from a fusion simulation are included here, that shows modest performance degradation in the two execution modes. Computational biology applications, on the other hand, have shown slow downs of over 40% in the dual-core mode runtimes as compared to the single-core mode runtimes.

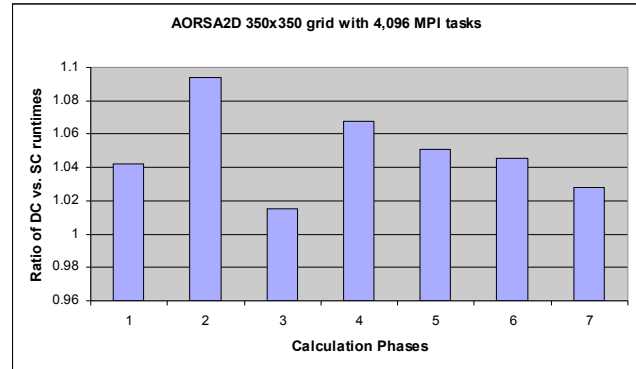
In order to help us understand the performance of these complex applications, we include performance results from micro-benchmarks, such as those found in the HPC Challenge suite [J. Dongarra '05] as well as the Intel MPI benchmark [Intel '07]. These results are going to be presented in the final poster. Like the application results, the benchmark results show performance losses ranging from less than 1% to over 50% in the dual-code runtimes as compared to the single-core runtimes.

3.1 Fusion energy

AORSA is a Fortran based application program used to simulate the behavior of rf heating of a plasma in a fusion device such as a tokamak [E.F. Jaeger '06]. Operating in Fourier space, the problem is formulated using Maxwell's Equation, converted to real space, where a dense, linear system is solved. From this AORSA computes the various electric fields, etc.

Figure 1 compares the performance of the code phases in single core and dual core mode when operating on a 350x350 grid, executing across 4,096 MPI processes. This results in an order 232,812 linear system. Cray has applied special effort to improving the dual core performance of the ScaLAPACK library, so it is interesting to note that the strongest dual core performance occurs within this operation (the ScaLAPACK dense linear solver **PZGESV**, component 3). Because the time spent in this routine constitutes the majority of overall runtime, AORSA experiences less than 3% performance degradation when running in dual core mode (the last bar).

Figure 1: This graph compares the performance of AORSA using dual or single core mode. The x-axis represents the code phases, with phase 7 representing total time. A y-value of 1.0 would indicate no change in performance.



4. Conclusions and Plans

Our evaluation includes runtime configurations ranging from one to several thousand processors. Our work shows that most applications evaluated map easily and successfully to the dual core XT3 system, maintaining near 90% of single process performance and scalability. Some applications show a small degradation, 20-30%, while the performance of one important code decreased significantly. Further, we use this information to speculate on approaches for improving those codes that did not perform as anticipated, and discuss issues that may adversely impact performance when a Cray XT4, configured for quad-core processors, is installed at ORNL.

Acknowledgements

The authors would like to thank the staff of the National Center for Computational Sciences (NCCS). This research used resources of the Center for Computational Sciences at Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.

References

- [1] S.R. Alam, R.F. Barrett, M.R. Fahey, J.A. Kuehn, O.E. Messer, R.T. Mills, P.C. Roth, J.S. Vetter, and P.H. Worley, "An Evaluation of the ORNL XT3," *International Journal of High Performance Computing Applications*, to appear, 2007.
- [2] J. Dongarra and P. Luszczek, "Introduction to the HPCChallenge Benchmark Suite," *Computer Science Department Tech Report:05-544*, 2005.
- [3] Intel, *Intel MPI Benchmarks*, <http://www.intel.com>, 2007.
- [4] E.F. Jaeger, L.A. Berry, S.D. Ahern, R.F. Barrett, D.B. Batchelor, M.D. Carter, E.F. D'Azevedo, R.D. Moore, R.W. Harvey, and J.R. Myra, "Self-consistent full-wave and Fokker-Planck calculations for ion cyclotron heating in non-Maxwellian plasmas," *Physics of Plasmas*, 13(5):56101-, 2006.