Understanding MPI Load Imbalance with Intel® Trace Analyzer and Collector

Introduction

Many MPI applications are written with the implicit assumption that they will run on homogenous systems. The problem domain is decomposed into equal chunks of work assigned to each MPI rank. Load imbalance results when one rank takes longer to complete its chunk than the others, but given a well written application running on a homogeneous system, load imbalance is typically minimal.

A system based on Intel® Xeon® processors and Intel® Xeon Phi™ coprocessors is heterogeneous. On a core-to-core basis, a single Intel Xeon core is more computationally powerful than a single Intel Xeon Phi core. Given a typical MPI scenario, where equal chunks of work are assigned to each MPI rank and one MPI rank is run per core, running the job on both Intel Xeon processors and Intel Xeon Phi coprocessors will result in severe load imbalance.

One way to address load imbalance is dynamically allocate chunks of work based on the relative computational power of each MPI rank. However, few MPI applications already do this, and the application source code would likely need significant changes to do this.

A more common scenario is an application that combines MPI with a threading model such as OpenMP. Since the chunk of work is the same for all MPI ranks regardless of how many threads are associated with the rank, the basis for balancing a heterogeneous platform is to assign more threads per rank to the ranks running on the relatively slower nodes. This paper describes how to choose the number of MPI ranks and OpenMP threads when running on a system with both Intel Xeon processors and Intel Xeon Phi coprocessors to minimize the amount of load imbalance.

Intel® Trace Analyzer and Collector

Intel® Trace Analyzer and Collector (ITAC) is a powerful tool for understanding MPI application behavior, quickly finding bottlenecks, and achieving high performance for parallel cluster applications. ITAC provides low overhead tracing of MPI applications, coupled with a graphical user interface to visualize the resulting data. The ITAC event timeline view can quickly and easily show load imbalance. More information can be found at www.intel.com/go/itac.

Understanding Load Imbalance

The miniFE workload from Sandia National Lab (https://software.sandia.gov/mantevo/packages.html) is used to illustrate how to understand load imbalance. The workload was run a 4 node cluster, where each node is composed of 2 Intel Xeon processors and 2 Intel Xeon Phi coprocessors. To make the traces easier to view, the trace data was aggregated at the node level in the ITAC GUI (Advanced ->
Process Aggregation). Note that each Intel Xeon Phi card appears as separate node in the trace since each card has its own operating system and IP address. NxM is used to refer to the number of MPI ranks and OpenMP threads per rank, e.g., 8x4 means 8 MPI ranks and 4 OpenMP threads per rank.

First, the workload was run 16 MPI ranks and 1 OpenMP thread per rank on the host and 8 MPI ranks with 28 OpenMP threads per rank on each coprocessor. Figure 1 shows the ITAC trace of this configuration. There is MPI synchronization at the end of each iteration. The red/blue imbalance shows that the coprocessors complete their chunks of work (shown in blue) in approximately half the time as the hosts. The coprocessors then wait in the MPI synchronization (shown in red) until the host catches up.

![Image](image1.png)

**Figure 1** 4 iterations of miniFE using 16x1 on the host and 8x28 on each coprocessor. Red regions represent MPI and blue regions represent application code. The area outlined in yellow represents a single iteration.

The configuration was then adjusted to run 16 MPI ranks and 1 OpenMP thread per rank on the host and 24 MPI ranks with 8 OpenMP thread per rank on each coprocessor. Figure 2 shows the ITAC trace of this configuration. The load imbalance is improved compared to the first configuration, but there is still a significant amount of imbalance. Here the situation is reversed: the hosts are now completing their
chunks of work faster than the coprocessors and then wait inside the MPI synchronization for the coprocessors to catch up.

Figure 2 4 iterations of miniFE using 16x1 on the host and 24x8 on each coprocessor.

The configuration was further adjusted to run 16 MPI ranks and 1 OpenMP thread per rank on the host and 16 MPI ranks and 12 OpenMP ranks on each of the cards. Figure 3 shows the ITAC trace of this configuration. This configuration achieves near ideal load imbalance: the amount of work assigned to each MPI rank is balanced against the relative computational rates of the Intel Xeon processors and Intel Xeon Phi coprocessors.
Figure 3 4 iterations of miniFE using 16x1 on the host and 16x12 on each coprocessor.

Note that this particular balanced configuration is specific to this application. However, this methodology can be applied to any hybrid MPI/OpenMP application to determine the configuration that achieves the best load balance on a system with Intel Xeon processors and Intel Xeon Phi coprocessors.

About the Author
Scott McMillan is a senior software engineer in Intel’s Software and Services Group. During his time at Intel, he has worked on cluster system management, parallel application scaling, HPC/Cloud, and the MPI-3 standard, among others. He is the creator of Intel® Cluster Checker. Scott holds a Ph.D. in chemical engineering from Northwestern University.
Notices

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

A "Mission Critical Application" is any application in which failure of the Intel Product could result, directly or indirectly, in personal injury or death. SHOULD YOU PURCHASE OR USE INTEL'S PRODUCTS FOR ANY SUCH MISSION CRITICAL APPLICATION, YOU SHALL INDEMNIFY AND HOLD INTEL AND ITS SUBCONTRACTORS, SUBCONTRACTORS AND AFFILIATES, AND THE DIRECTORS, OFFICERS, AND EMPLOYEES OF EACH, HARMLESS AGAINST ALL CLAIMS COSTS, DAMAGES, AND EXPENSES AND REASONABLE ATTORNEYS' FEES ARISING OUT OF, DIRECTLY OR INDIRECTLY, ANY CLAIM OF PRODUCT LIABILITY, PERSONAL INJURY, OR DEATH ARISING IN ANY WAY OUT OF SUCH MISSION CRITICAL APPLICATION, WHETHER OR NOT INTEL OR ITS SUBCONTRACTOR WAS NEGLIGENT IN THE DESIGN, MANUFACTURE, OR WARNING OF THE INTEL PRODUCT OR ANY OF ITS PARTS.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined". Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request. Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

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, the Intel logo, VTune, Phi and Xeon are trademarks of Intel Corporation in the U.S. and other countries.

*Other names and brands may be claimed as the property of others

Copyright © 2013 Intel Corporation. All rights reserved.

Optimization Notice

**Performance Notice**

For more complete information about performance and benchmark results, visit
www.intel.com/benchmarks

**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 SSE3 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