Best Practices: Application Profiling at the HPCAC High Performance Center

Pak Lui

Stanford Conference 2017

February 07-08, 2017
Munger Conference Center - Paul Brest Hall
Stanford University
<table>
<thead>
<tr>
<th>Application</th>
<th>Application</th>
<th>Application</th>
<th>Application</th>
<th>Application</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abaqus</td>
<td>COSMO</td>
<td>HPCC</td>
<td>Nekbone</td>
<td>RFD tNavigator</td>
<td></td>
</tr>
<tr>
<td>ABySS</td>
<td>CP2K</td>
<td>HPCG</td>
<td>NEMO</td>
<td>SNAP</td>
<td></td>
</tr>
<tr>
<td>AcuSolve</td>
<td>CPMD</td>
<td>HYCOM</td>
<td>NWChem</td>
<td>SPECFEM3D</td>
<td></td>
</tr>
<tr>
<td>Amber</td>
<td>Dacapo</td>
<td>ICON</td>
<td>Octopus</td>
<td>STAR-CCM+</td>
<td></td>
</tr>
<tr>
<td>AMG</td>
<td>Desmond</td>
<td>Lattice QCD</td>
<td>OpenAtom</td>
<td>STAR-CD</td>
<td></td>
</tr>
<tr>
<td>AMR</td>
<td>DL-POLY</td>
<td>LAMMPS</td>
<td>OpenFOAM</td>
<td>VASP</td>
<td></td>
</tr>
<tr>
<td>ANSYS CFX</td>
<td>Eclipse</td>
<td>LS-DYNA</td>
<td>OpenMX</td>
<td>WRF</td>
<td></td>
</tr>
<tr>
<td>ANSYS Fluent</td>
<td>FLOW-3D</td>
<td>miniFE</td>
<td>OptiStruct</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ANSYS Mechanical</td>
<td>GADGET-2</td>
<td>MILC</td>
<td>PAM-CRASH / VPS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BQCD</td>
<td>Graph500</td>
<td>MSC Nastran</td>
<td>PARATEC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BSMBench</td>
<td>GROMACS</td>
<td>MR Bayes</td>
<td>Pretty Fast Analysis</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CAM-SE</td>
<td>Himeno</td>
<td>MM5</td>
<td>PFLOTRAN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCSM 4.0</td>
<td>HIT3D</td>
<td>MPQC</td>
<td>Quantum ESPRESSO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CESM</td>
<td>HOOMD-blue</td>
<td>NAMD</td>
<td>RADIOSS</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

35 Applications Installation Best Practices Published

- Adaptive Mesh Refinement (AMR)
- Amber (for GPU/CUDA)
- Amber (for CPU)
- ANSYS Fluent 15.0.7
- ANSYS Fluent 17.1
- BQCD
- CASTEP 16.1
- CESM
- CP2K
- CPMD
- DL-POLY 4
- ESI PAM-CRASH 2015.1
- ESI PAM-CRASH / VPS 2013.1
- GADGET-2
- GROMACS 5.1.2
- GROMACS 4.5.4
- GROMACS 5.0.4 (GPU/CUDA)
- Himeno
- HOOMD Blue
- LAMMPS
- LAMMPS-KOKKOS
- LS-DYNA
- MrBayes
- NAMD
- NEMO
- NWChem
- Octopus
- OpenFOAM
- OpenMX
- PyFR
- Quantum ESPRESSO 4.1.2
- Quantum ESPRESSO 5.1.1
- Quantum ESPRESSO 5.3.0
- WRF 3.2.1
- WRF 3.8

For more information, visit: [http://www.hpcadvisorycouncil.com/subgroups_hpc_works.php](http://www.hpcadvisorycouncil.com/subgroups_hpc_works.php)
Agenda

• Overview of HPC Applications Performance
• Way to Inspect, Profile, Optimize HPC Applications
  – CPU, memory, file I/O, network
• System Configurations and Tuning
• Case Studies, Performance Comparisons, Optimizations and Highlights
• Conclusions
HPC Application Performance Overview

- **To achieve scalability performance on HPC applications**
  - Involves understanding of the workload by performing profile analysis

- **Tune for the most time spent (either CPU, Network, IO, etc)**
  - Underlying implicit requirement: Each node to perform similarly

- **Run CPU/memory/network tests or cluster checker to identify bad node(s)**
  - Comparing behaviors of using different HW components

- **Which pinpoint bottlenecks in different areas of the HPC cluster**

- **A selection of HPC applications will be shown**
  - To demonstrate method of profiling and analysis
  - To determine the bottleneck in SW/HW
  - To determine the effectiveness of tuning to improve on performance
Ways To Inspect and Profile Applications

• **Computation (CPU/Accelerators)**
  – Tools: gprof, top, htop, perf top, pstack, Visual Profiler, etc
  – Tests and Benchmarks: HPL, STREAM

• **File I/O**
  – Bandwidth and Block Size: iostat, collectl, darshan, etc
  – Characterization Tools and Benchmarks: iozone, ior, etc

• **Network Interconnect and MPI communications**
  – Tools and Profilers: perfquery, MPI profilers (IPM, TAU, etc)
  – Characterization Tools and Benchmarks:
    – Latency and Bandwidth: OSU benchmarks, IMB
The following research was performed under the HPC Advisory Council activities
  - Compute resource - HPC Advisory Council Cluster Center

The following was done to provide best practices
  - BSMBench performance overview
  - Understanding BSMBench communication patterns
  - Ways to increase BSMBench productivity

For more info please refer to
  - https://gitlab.com/edbennett/BSMBench
BSMBench

- Open source supercomputer benchmarking tool
- Based on simulation code used for studying strong interactions in particle physics
- Includes the ability to tune the ratio of communication over computation
- Includes 3 examples that show the performance of the system for
  - Problem that is computationally dominated (marked as Communications)
  - Problem that is communication dominated (marked as Compute)
  - Problem in which communication and computational requirements are balanced (marked as Balance)
- Used to simulate workload such as Lattice Quantum ChromoDynamics (QCD), and by extension its parent field, Lattice Gauge Theory (LGT), which make up a significant fraction of supercomputing cycles worldwide
- For reference: technical paper published at the 2016 International Conference on High Performance Computing & Simulation (HPCS), Innsbruck, Austria, 2016, pp. 834-839
Objectives

• The presented research was done to provide best practices
  – BSMBench performance benchmarking
    • MPI Library performance comparison
    • Interconnect performance comparison
    • Compilers comparison
    • Optimization tuning

• The presented results will demonstrate
  – The scalability of the compute environment/application
  – Considerations for higher productivity and efficiency
Test Cluster Configuration

• Dell PowerEdge R730 32-node (1024-core) “Thor” cluster
  – Dual-Socket 16-Core Intel E5-2697Av4 @ 2.60 GHz CPUs (BIOS: Maximum Performance, Home Snoop)
  – Memory: 256GB memory, DDR4 2400 MHz, Memory Snoop Mode in BIOS sets to Home Snoop
  – OS: RHEL 7.2, M MLNX_OFED_LINUX-3.4-1.0.0.0 InfiniBand SW stack

• Mellanox ConnectX-4 EDR 100Gb/s InfiniBand Adapters

• Mellanox Switch-IB SB7800 36-port EDR 100Gb/s InfiniBand Switch

• Intel® Omni-Path Host Fabric Interface (HFI) 100Gbps Adapter

• Intel® Omni-Path Edge Switch 100 Series

• Dell InfiniBand-Based Lustre Storage based on Dell PowerVault MD3460 and Dell PowerVault MD3420

• Compilers: Intel Compilers 2016.4.258

• MPI: Intel Parallel Studio XE 2016 Update 4, Mellanox HPC-X MPI Toolkit v1.8

• Application: BSMBench Version 1.0

• MPI Profiler: IPM (from Mellanox HPC-X)
BSMBench Profiling – % of MPI Calls

- **Major MPI calls (as % of wall time):**
  - Balance: MPI_Barrier (26%), MPI_Allreduce (6%), MPI_Waitall (5%), MPI_Isend (4%)
  - Communications: MPI_Barrier (14%), MPI_Allreduce (5%), MPI_Waitall (5%), MPI_Isend (2%)
  - Compute: MPI_Barrier (14%), MPI_Allreduce (5%), MPI_Waitall (5%), MPI_Isend (1%)

![Pie charts showing balance, communications, and compute categories of MPI calls.]

32 Nodes / 1024 Processes
BSMbench Profiling – MPI Message Size Distribution

- Similar communication pattern seen across all 3 examples:
  - Balance: MPI_BARRIER: 0-byte, 22% wall, MPI_Allreduce: 8-byte, 5% wall
  - Communications: MPI_BARRIER: 0-byte, 26% wall, MPI_Allreduce: 8-byte, 5% wall
  - Compute: MPI_BARRIER: 0-byte, 13% wall, MPI_Allreduce: 8-byte, 5% wall
The different communications across the MPI processes is mostly balance
- Does not appear to be any significant load imbalances in the communication layer

32 Nodes / 1024 Processes
BSMBench Performance – Interconnects

- **EDR InfiniBand delivers better scalability for BSMBench**
  - Similar performance between EDR InfiniBand and Omni-Path up to 8 nodes
  - Close to 20% performance advantage for InfiniBand at 32 nodes
  - Similar performance difference across the three different examples
BSMBench Performance – MPI Libraries

- Comparison between two commercial available MPI libraries
- Intel MPI and HPC-X delivers similar performance
  - HPC-X demonstrates 5% advantage at 32 nodes
Test Cluster Configuration

- **IBM OperPOWER 8-node “Telesto” cluster**
- **IBM Power System S822LC (8335-GTA)**
  - IBM: Dual-Socket 10-Core @ 3.491 GHz CPUs, Memory: 256GB memory, DDR3 PC3-14900 MHz
- **Wistron OpenPOWER servers**
  - Wistron: Dual-Socket 8-Core @ 3.867 GHz CPUs. Memory: 224GB memory, DDR3 PC3-14900 MHz
- **OS**: RHEL 7.2, MLNX_OFED_LINUX-3.4-1.0.0.0 InfiniBand SW stack
- **Mellanox ConnectX-4 EDR 100Gb/s InfiniBand Adapters**
- **Mellanox Switch-IB SB7800 36-port EDR 100Gb/s InfiniBand Switch**
- **Compilers**: GNU compilers 4.8.5, IBM XL Compilers 13.1.3
- **MPI**: Open MPI 2.0.2, IBM Spectrum MPI 10.1.0.2
- **Application**: BSM Bench Version 1.0
BSMBench Performance – SMT

- Simultaneous Multithreading (SMT) allows additional hardware threads for compute
- Additional performance gain is seen with SMT enabled
  - Up to 23% of performance gain is seen between no SMT versus 4 SMT threads are used

**Higher is better**
BSMBench Performance – MPI Libraries

- **Spectrum MPI with MXM support delivers higher performance**
  - Spectrum MPI provides MXM and PAMI protocol for transport/communications
  - Up to 19% of higher performance at 4 nodes / 64 cores using Spectrum MPI / MXM
BSMBench Performance – CPU Architecture

• **IBM architecture demonstrates better performance compared to Intel architecture**
  – Performance gain on a single node is ~23% for Comms and Balance
  – Additional gains are seen when more SMT hardware threads are used
  – 32 cores per node used for Intel, versus 16 cores used per node for IBM

![BSMBench Performance (Balance)](chart.png)

*Higher is better*

**Legend:**
- IBM POWER8 @ 3491MHz
- Intel E5-2697Av4 @ 2.6GHz
BSMBench Summary

• **Benchmark for BSM Lattice Physics**
  – Utilizes both compute and network communications

• **MPI Profiling**
  – Most MPI time is spent on MPI collective operations and non-blocking communications
    • Heavy use of MPI collective operations (MPI_Allreduce, MPI_Barrier)
  – Similar communication patterns seen across all three examples
    • Balance: MPI_Barrier: 0-byte, 22% wall, MPI_Allreduce: 8-byte, 5% wall
    • Comms: MPI_Barrier: 0-byte, 26% wall, MPI_Allreduce: 8-byte, 5% wall
    • Compute: MPI_Barrier: 0-byte, 13% wall, MPI_Allreduce: 8-byte, 5% wall

Fast network communication is important for scalability
BSMBench Summary

- **Interconnect comparison**
  - EDR InfiniBand demonstrates higher scalability beyond 16 nodes as compared to Omni-Path
  - EDR InfiniBand delivers nearly 20% higher performance 32 nodes / 1024 cores
  - Similar performance advantage across all three example cases

- **Simultaneous Multithreading (SMT) provides additional benefit for compute**
  - Up to 23% of performance gain is seen between no SMT versus 4 SMT threads are used

- **IBM CPU outperforms Intel by approximately 23%**
  - 32 cores per node used for Intel, versus 16 cores used per node for IBM

- **Spectrum MPI provides MXM and PAMI protocol for transport/communications**
  - Up to 19% of higher performance at 4 nodes / 64 cores using Spectrum MPI / MXM
Thank You