BSMBench
Performance Benchmark and Profiling

February 2017

Note

- **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

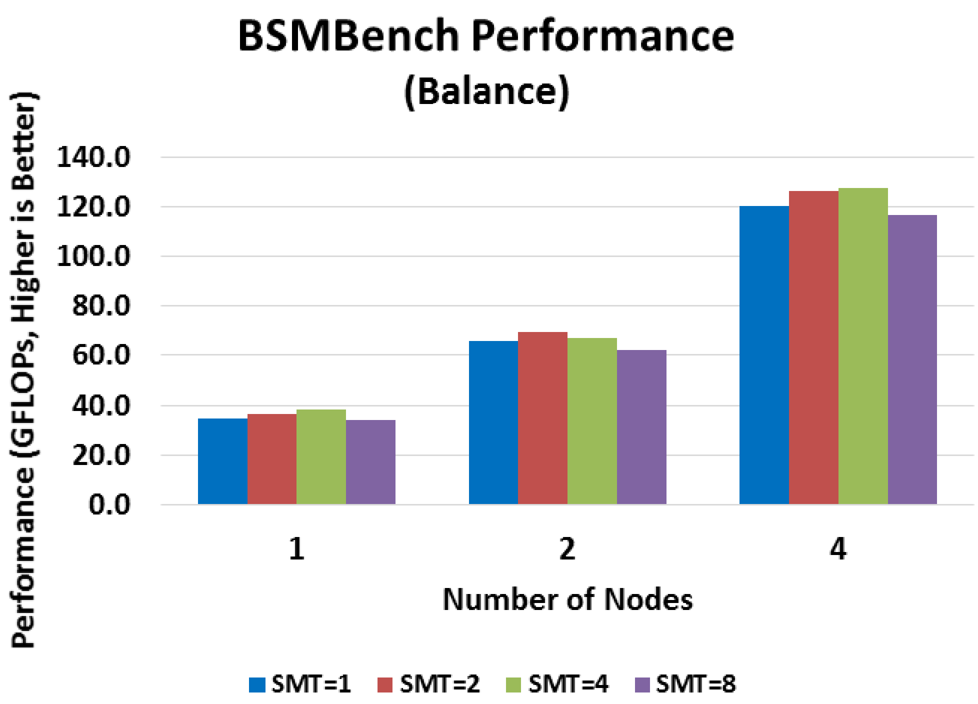
- **The presented research was done to provide best practices**
  - BSMBench performance benchmarking
    - MPI Library performance comparison
    - Interconnect performance comparison
    - Compilers comparison
    - Optimization tuning
- **The presented results will demonstrate**
  - The scalability of the compute environment/application
  - Considerations for higher productivity and efficiency

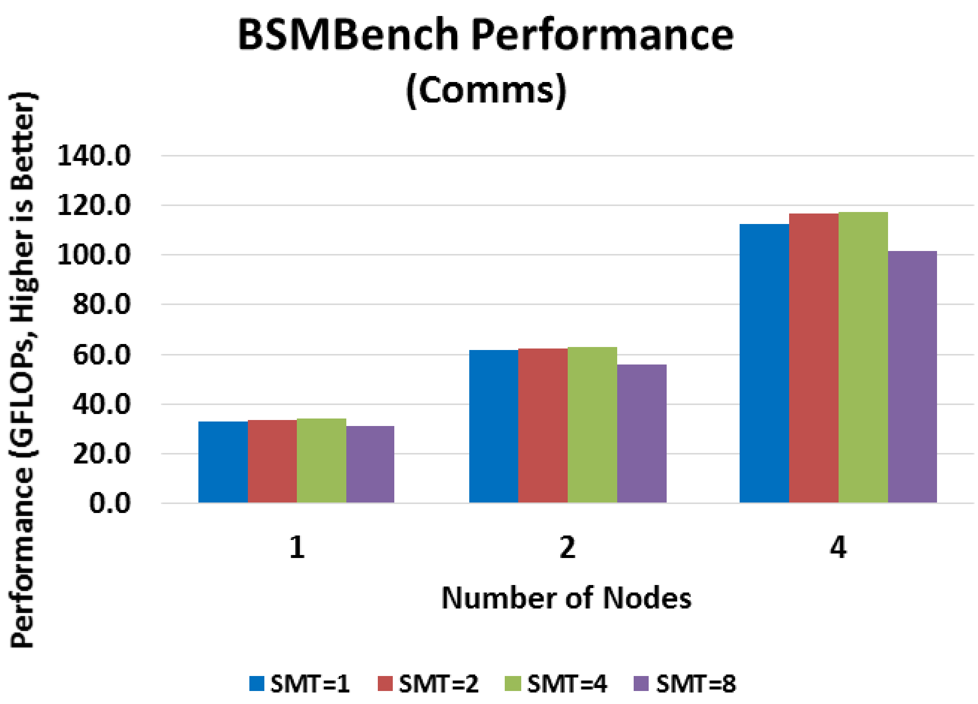
Test Cluster Configuration

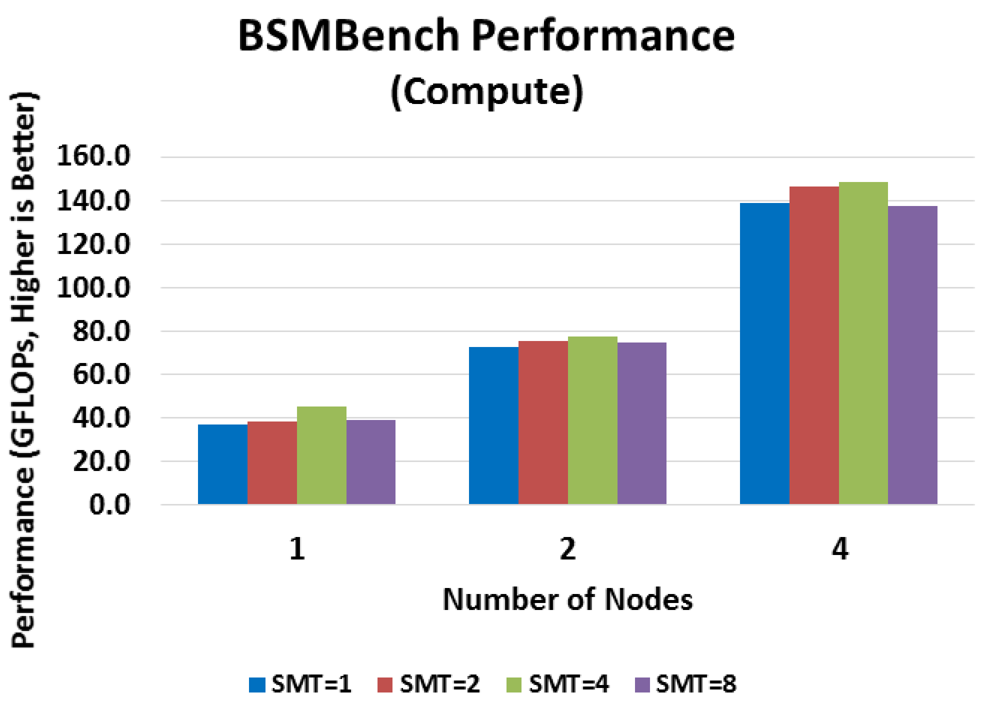
- **IBM OperPOWER 8-node “Telesto” cluster**
- **IBM Power System S822LC (8335-GTA)**
  - IBM: Dual-Socket 10-Core @ 3.491 GHz CPUs, Memory: 256GB memory, DDR3 PC3-14900 MHz
- **Wistron OpenPOWER servers**
  - Wistron: Dual-Socket 8-Core @ 3.867 GHz CPUs. Memory: 224GB memory, DDR3 PC3-14900 MHz
- **OS: 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: Mellanox HPC-X MPI Toolkit v1.8, IBM Spectrum MPI 10.1.0.2**
- **Application: BSMBench Version 1.0**
- **MPI Profiler: IPM (from Mellanox HPC-X)**

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



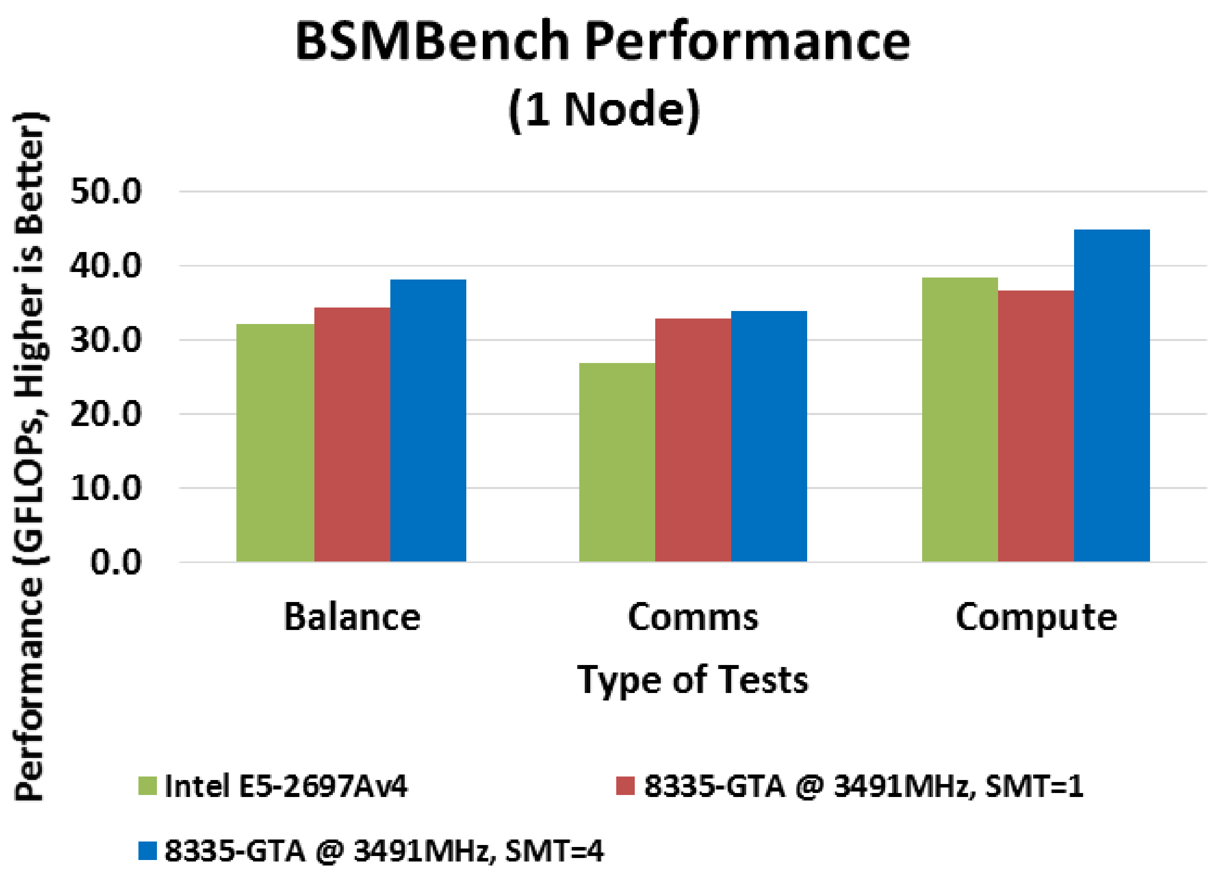


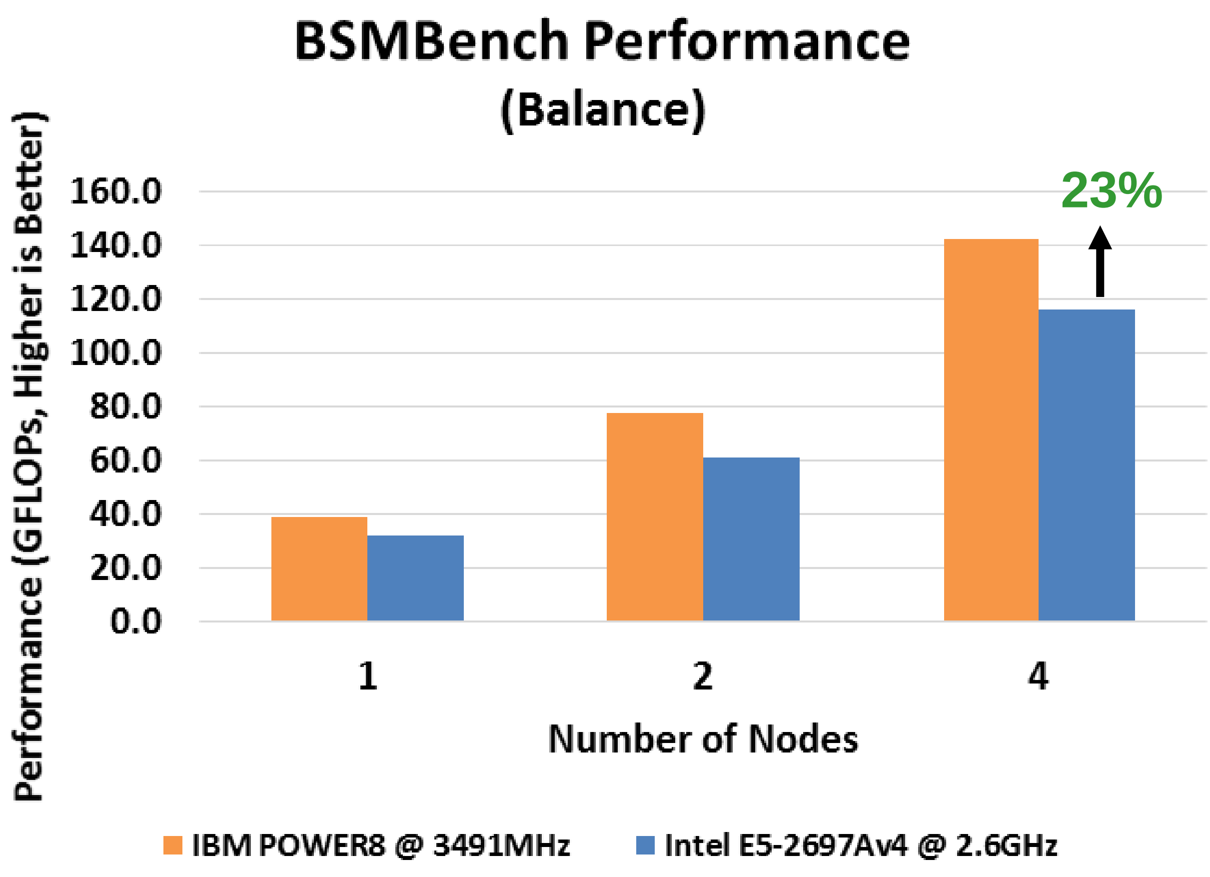


Higher is better

BSMBench Performance – CPU Architecture

- **IBM architecture demonstrates higher performance versus x86**
  - Performance gain on a single node is approximately 20% for Communications 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

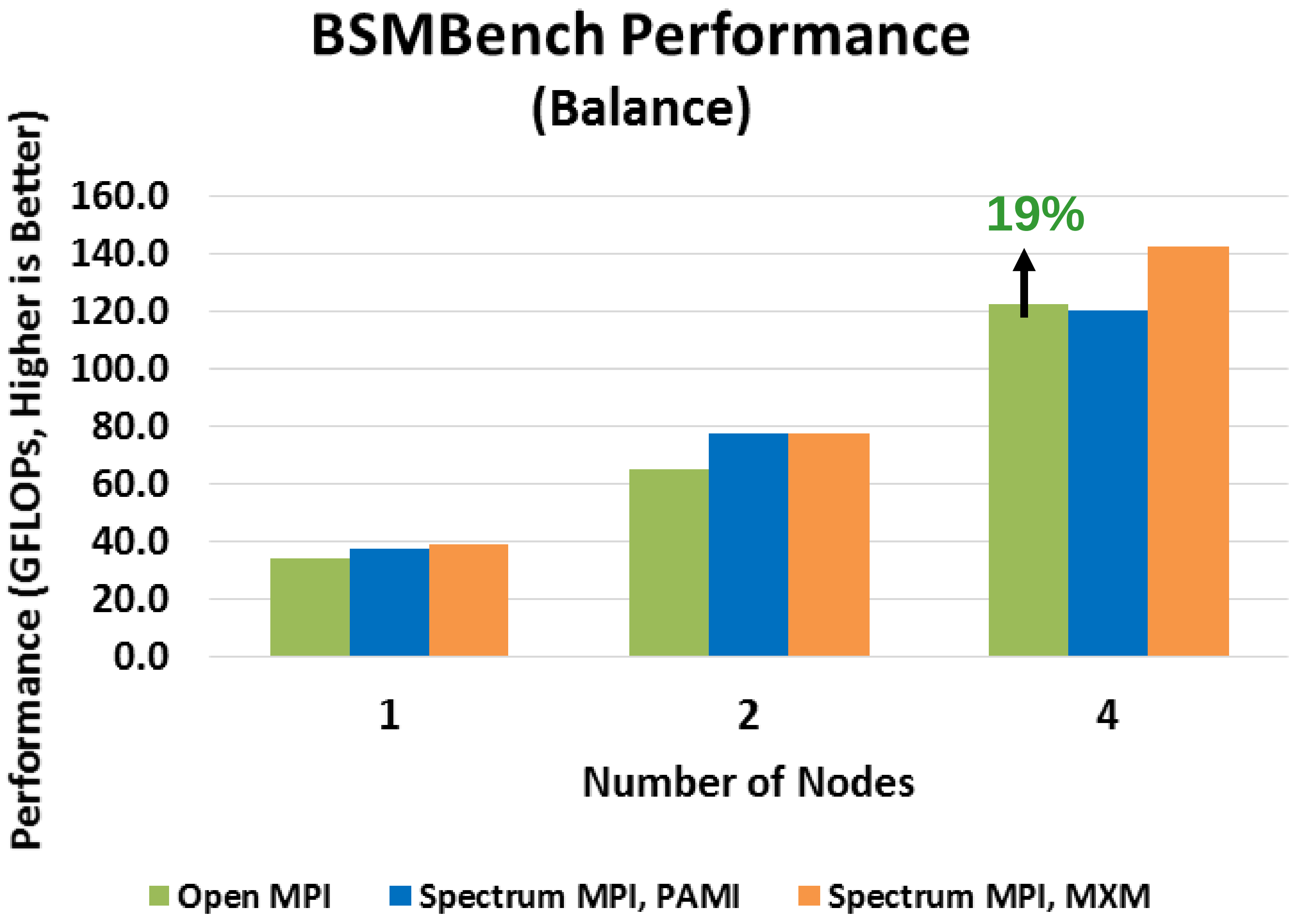




Higher is better

BSMBench Performance – MPI Libraries

- **Spectrum MPI (IBM) 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

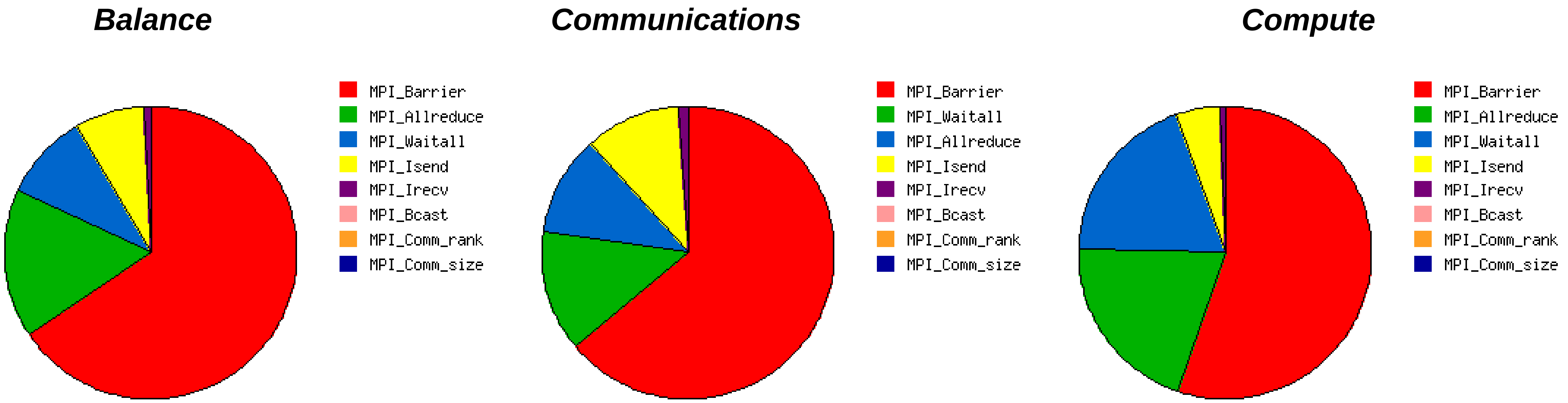


Higher is better

32 MPI Processes / Node

BSMBench Profiling – % of MPI Calls

- **For the most time consuming MPI calls (as % of wall time):**
  - Balance: MPI_Barrier (26%), MPI_Allreduce (6%), MPI_Waitall (5%), MPI_Isend (4%)
  - Comms: 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%)



32 Nodes / 1024 Processes

BSMBench Summary

- **Benchmark for BSM Lattice Physics**
  - Utilizes both compute and network communications
- **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 Power provides higher performance versus x86**
  - By 20% on a single node basis, 32 cores per node used for Intel, versus 16 cores used per node for IBM
  - By 23% on 4 nodes cluster testing
- **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
- **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

Thank You

All trademarks are property of their respective owners. All information is provided “As-Is” without any kind of warranty. The HPC Advisory Council makes no representation to the accuracy and completeness of the information contained herein. HPC Advisory Council undertakes no duty and assumes no obligation to update or correct any information presented herein