BSMBench Installation Best Practices

HPC ADVISORY COUNCIL
NETWORK OF EXPERTISE
BEST PRACTICES

1. Introduction:

The following best practices document is provided as courtesy of the HPC Advisory Council.

2. Application Description:

Open source supercomputer benchmarking tool that is based on simulation code used for studying strong interactions in particle physics. BSMBench includes the ability to tune the ratio of communication over computation. It includes 3 examples that show the performance of the system for the problem that is computationally dominated (marked as Communications), a problem that is communication dominated (marked as Compute), and a problem in which communication and computational requirements are balanced (marked as Balance). BSMBench 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, there is a technical paper published at the 2016 International Conference on High Performance Computing & Simulation (HPCS), Innsbruck, Austria, 2016, pp. 834-839

3. Version Information:

More information about BSMBench is available at https://gitlab.com/edbennett/BSMBench

4. Prerequisites:

The instructions from this best practice have been tested on the following configuration:
Hardware:

- Dell PowerEdge R730 32-node (1024-core) "Thor" cluster.
- Dual-Socket 16-Core Intel E5-2697A v4 @ 2.60 GHz CPUs
- Mellanox ConnectX-4 EDR InfiniBand adapters
- Mellanox Switch-IB SB7800 EDR InfiniBand switch

OS and software:

- RHEL 7.2, MLNX_OFED_LINUX-3.3-1.0.4.0 InfiniBand SW stack
- Intel Compilers 2017.0.098
- MPI: Mellanox HPC-X v1.8 and IBM Spectrum MPI 10.1.0.3, Intel MPI 2017

5. Building BSMBench

5.1 Makefile changes

Make the following changes to the Makefile for Mellanox HPC-X, or Intel MPI:

```
#!/bin/bash
module purge
module load intel/compiler/2017.0.098
MPI=hpcx
#MPI=impi
#MPI=smpi
MPI=hpcx-1.8
if [ "$MPI" == "impi" ]; then
        module load intel/impi
```

```
        export I_MPI_CC=icc
        export I_MPI_CXX=icpc
        export I_MPI_FC=ifort
        export I_MPI_F90=ifort
elif [[ "$MPI" == "hpcx" || "$MPI" == "hpcx-1.8" ]]; then
        module load $MPI/icc-2016
        export OMPI_MPICC=icc
        export OMPI_MPICXX=icpc
        export OMPI_MPIFC=ifort
        export OMPI_MPIF90=ifort
elif [ "$MPI" == "smpi" ]; then
        module load ibm/spectrum_mpi/10.1.0
        export OMPI_MPICC=icc
        export OMPI_MPICXX=icpc
        export OMPI_MPIFC=ifort
        export OMPI_MPIF90=ifort
fi
rm -fr bin-$MPI
mkdir bin-$MPI
./make.sh machine-config/INTEL.cfg
mv bsmbench* bin-$MPI
```

6. Running BSMBENCH

6.1 Running BSMBENCH with Mellanox HPC-X MPI Toolkit

```
module load intel/compiler/2017.0.098
module load hpcx/icc-2017

mpirun -np 1024 -bind-to core -mca btl_sm_use_knem 1 -mca coll_fca_enable 0 -mc
a coll_hcoll_enable 1 -mca coll_hcoll_np 0 -x HCOLL_ENABLE_MCAST_ALL=1 -x HCOLL
_CONTEXT_CACHE_ENABLE=1 -mca pml yalla -mca mtl_mxm_np 0 -x MXM_TLS=ud,shm,self
 -x MXM_RDMA_PORTS=mlx5_0:1 -mca btl_openib_if_include mlx5_0:1 -x MALLOC_MMAP_
MAX_=0 -x MALLOC_TRIM_THRESHOLD_=-1 -x LD_PRELOAD=/opt/hpcx/ompi-v1.10.i2016/te
sts/ipm-2.0.2/lib/libipm.so /home/bsm/BSMBench-1.0/bin-hpcx-2016/bsmbench_compu
te -i sets/compute-1024.bsmbench -o log-i2016-compute-hpcx-32-32-1024-EDR-1hcol
l-1mxm-ud-13317-20170202-1126.txt
```

6.2 Running BSMBENCH with Intel MPI

```
module load intel/compiler/2017.0.098
module load hpcx/icc-2017

mpirun -np 1024 -genv I_MPI_PIN on -genv DAT_OVERRIDE /etc/dat.conf -genv I_MPI
_DAT_LIBRARY /usr/lib64/libdat2.so -IB -genv MV2_USE_APM 0 -genv I_MPI_FABRICS
shm:ofa -genv I_MPI_OFA_USE_XRC 1 -genv I_MPI_OFA_NUM_ADAPTERS 1 -genv I_MPI_OF
A_ADAPTER_NAME mlx5_0 -genv I_MPI_OFA_NUM_PORTS 1 -genv MALLOC_MMAP_MAX_ 0 -gen
v MALLOC_TRIM_THRESHOLD_ -1 /home/bsm/BSMBench-1.0/bin-impi-2017/bsmbench_compu
te -i sets/compute-1024.bsmbench -o output/log-i2017-compute-impi-32-32-1024-ED
R-OFA-13263-20170131-1823.txt
```