A Tutorial on HPC and ML Communication Benchmarking

Moshe Voloshin and Hemal V. Shah

Data Center Solutions Group (DCSG), Broadcom Inc.

Agenda

- **Why Benchmark HPC and ML Communications?**
- **Micro-Benchmarks**
  - Perftest, OMB, IMB, NCCL, RCCL…
- **HPC Benchmarks**
  - GROMACS, HPCG, LAMPPS, OpenFOAM, WRF…
- **ML Benchmarks…**
  - PARAM, Horovod, MLPerf
- **Congestion Control: GPCNeT**
- **Performance Monitoring Tools**
- **Automation and Analysis**
- **Gaps in Benchmarking**
- **Summary**

Communication Benchmarking in HPC/ML Applications

- **HPC/ML applications are distributed/parallelized for scale & performance**
  - A process group on an HPC/ML node represents a collection of processes
  - The number of processes can be 100s per node (typically 1 per CPU core)
  - The number of nodes can scale to 1000s in a cluster
- **HPC/ML apps use send, receive, put, get, collectives, etc. comm operations**
- **Communication pattern of processes is represented by a logical topology**
  - Ring, Binary cube, Tree, etc.
- **Selection of logical topologies depends number of ranks and message size**
- **Comm operations, patterns, and topologies impact app performance**

Benchmarking communications in HPC/ML Applications is important

Classes of benchmarks

- **Micro benchmarks**
  - Single communication or collective operation, e.g. send, Reduce, Alltoall
  - Metric is completion time (latency) and/or BW
  - Main functional components under evaluation:
    - MPI layer including collective algorithms (algorithm selection)
    - API (verbs, OFI, UCX) and Provider SW
    - Provider HW (NIC) and congestion Control (CC) algorithm
    - Network elements
- **Application benchmarks**
  - Computation and communication – various patterns and interactions
  - Metric – overall performance and scaling with #nodes/PPN
  - Functional components under evaluation:
    - CPU/GPU, Memory, IO and other hosts components
    - MPI layer
    - API and Provider SW
    - Provider HW and CC
    - Network elements

Other classes (between micro and application benchmarks)

- **Generic HPC (HPL, HPCC, HPCG..)**
  - Combines several types of computation and communication that are common in HPC
- **Network effectiveness (e.g. GPCNet)**
  - Aim at evaluating effectiveness of network & Congestion Control (CC)
  - Measure a common set of communication operation with and without load
  - Targeted at large networks (multi switch)
- **Is there a need for other midlevel benchmarks?**
  - More generic than specific application BM
  - Yet richer than micro benchmarks
  - Possible scope
    - Combine (mix) collective ops as relevant to HPC, storage, or ML
    - Create certain patterns without computation itself (e.g. mimic Training, or Stencil simulation)
    - Combine collective and background load (as GPCNet) but targeted at a smaller network (single switch)

Micro-Benchmarks

Micro-Benchmark	Overview
Linux-rdma Perftest	A collection of latency and bandwidth tests for RDMA operations (Send, RDMA Read, RDMA Write, and RDMA Atomic). Uses verbs. Intended for use as a performance micro-benchmark for RDMA ops. The tests are useful for low level HW or SW tuning as well as for functional testing.
OSU Micro Benchmarks (OMB)	A collection of tests for point-2-point and collective communication operations. Uses Message Passing Interface (MPI) for communication operations. The tests are useful for tuning MPI libraries on a cluster system.
Intel MPI Benchmarks (IMB)	A set of MPI performance measurements for point-to-point and global communication operations for a range of message sizes. Useful for characterizing performance of a cluster system, including node performance, network latency, and throughput for a given MPI implementation
NCCL/RCCL Tests	NVIDIA/RoCm Collective Communication Library (NCCL/RCCL) tests check performance and correctness of NCCL/RCCL operations. Useful for characterizing specific GPUs and associated libraries

Micro-Benchmarks can't cover cluster level application communication patterns and network congestion

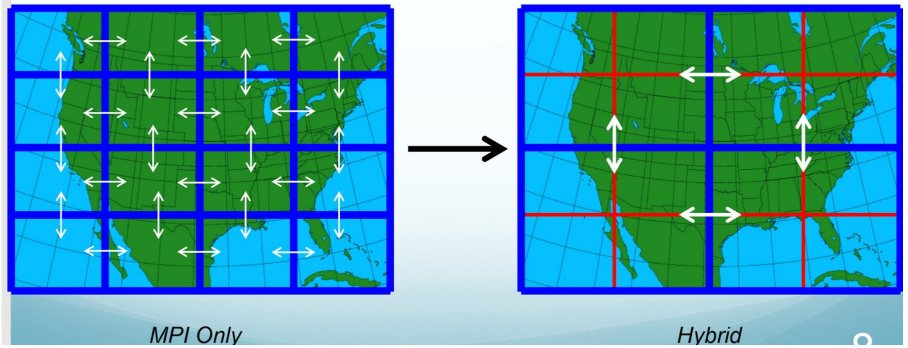
Generic HPC Benchmarks

- **HPL – High Performance Linpack**
  - Factoring and solving large dense system of linear equations
  - Dominant calculation is matrix-matrix multiplication (mostly done by GPU today)
- **HPCG - High Performance Conjugate Gradient**
  - Complement HPL and target a broader set of HPC applications governed by differential equations, which tend to have much stronger needs for high bandwidth and low latency
  - Tend to access data using irregular patterns
  - Iterative and heavily use neighborhood collectives
- **HPCC**
  - Consist of 7 test (HPL is one of them)
  - Each test focuses on a different aspect, e.g. floating point, memory access, communication, etc.

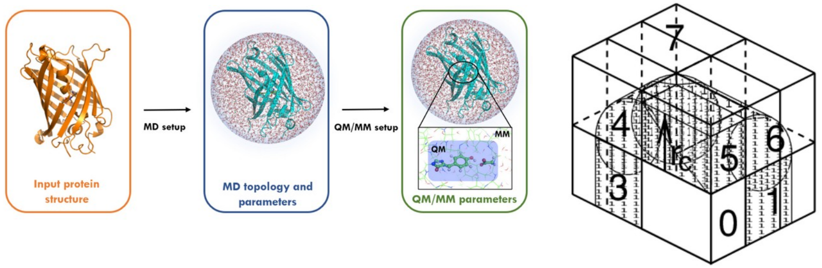
HPL - https://netlib.org/benchmark/hpl/
HPCG - https://hpcg-benchmark.org/
HPCC - https://hpcchallenge.org/hpcc/

HPC Application level benchmarks

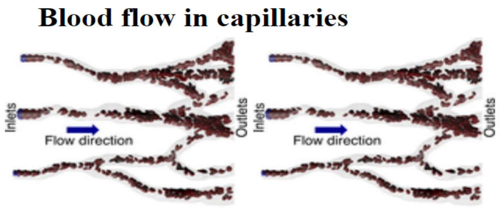
- **WRF - Weather Research and Forecasting**
  - Numerical weather prediction system
  - Uses OpenMP
  - Iteration loop time

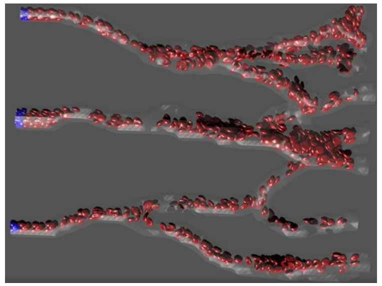


- **GROMACS**
  - Molecular dynamics
  - Primarily designed for biochemical molecules like proteins, lipids and nucleic acids
  - Differential equations, linear algebra, 3D stencil, 3D FFT
  - Uses OpenMP
  - ns/day plus detailed time breakdown of various steps



- **LAMMPS** - Large-scale Atomic/Molecular Massively Parallel Simulator
  - Molecular dynamics
  - Focus on materials modeling, solid state and soft matter
  - Conjugate gradient, DFT
  - Multiple benchmarks (Lenard-Jones, polymer chain, eam, etc.)
  - Metric – ns/day, % scaling with #processors





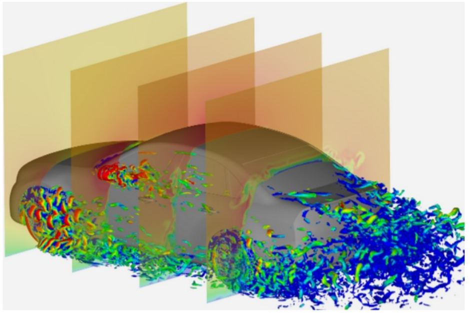
WRF - https://www.mmm.ucar.edu/weather-research-and-forecasting-model

GROMACS - https://www.gromacs.org/

LAMMPS - https://www.lammps.org/

HPC Application-level Benchmarks…

- **OpenFOAM**
  - Computational Fluid Dynamic
    - Includes chemical reactions, turbulence/heat transfer, acoustics, solid mechanics/electromagnetics, aerodynamics



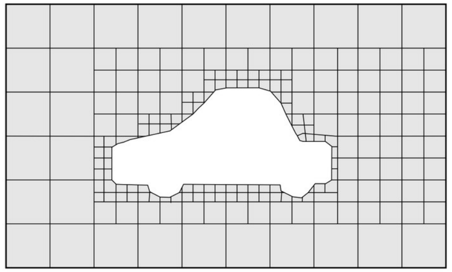
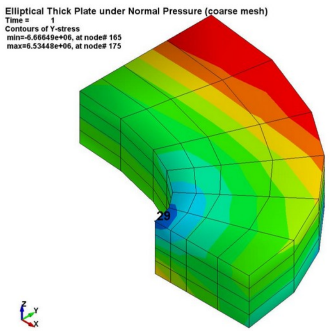
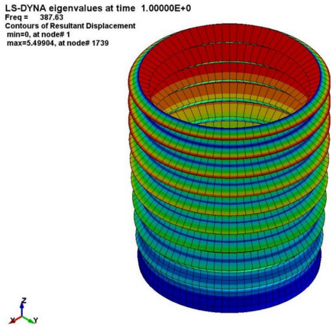


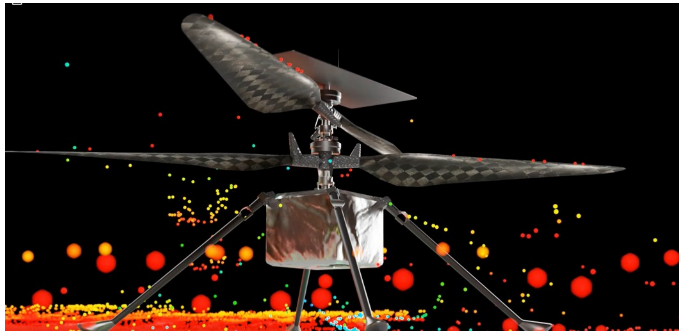
Figure 4.13: Surface snapping in snappyHexMesh meshing process

- **NAMD**
  - Molecular dynamic – large bio-molecular systems
  - Based on Charm++
- **LS-Dyna**
  - Structural analysis
  - Car crash, explosions, deformation, jet engine blade containment, bird strike
  - Stencils, system of PDEs





- **Fluent**
  - Fluids, acoustic, optics, avionics, etc.



OpenFOAM - https://www.openfoam.com/

NAMD - http://www.ks.uiuc.edu/Research/namd/

LS-DYNA - https://www.lstc.com/products/ls-dyna

Fluent - https://www.ansys.com/products/fluids/ansys-fluent

BROADCOM

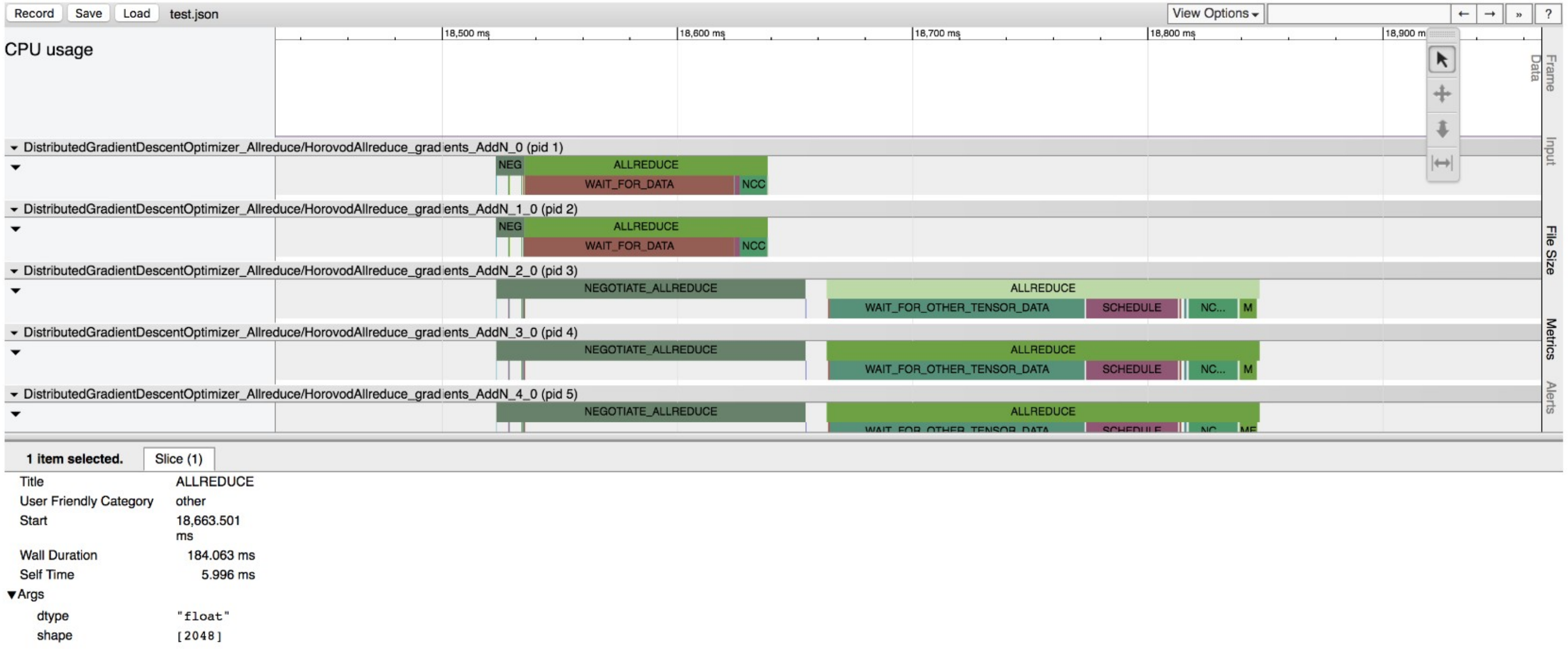
HPC application take away

- **Decompose problem domain to sub-volumes/areas**
- **Exchange simulation parameters between neighbors on each iteration**
- **Interesting communication aspects**
  - Size of parameter exchange
  - Frequency and overlap of exchange
  - Type of communication used in exchange – reduce, alltoall, etc
  - Alignment of communication start among processes -> affect burstiness and congestion
- **Can the communication aspect be profiled?**
- **Can BM mimic relevant comm aspects without computation behind them?**
  - That is focus on communication part but make sure it is realistic

Machine learning application level

- **Perform training of various popular neural nets**
  - Resnet50, SSD (single shot Detection), DLRM, NLP
  - Using common frameworks – e.g. tensorflow, pytorch, with Horovod for example
- **Could take long**
- **May require GPU to represent realistic use case**

Horovod has the ability to record the timeline of its activity, called Horovod Timeline.



Generic ML benchmarks

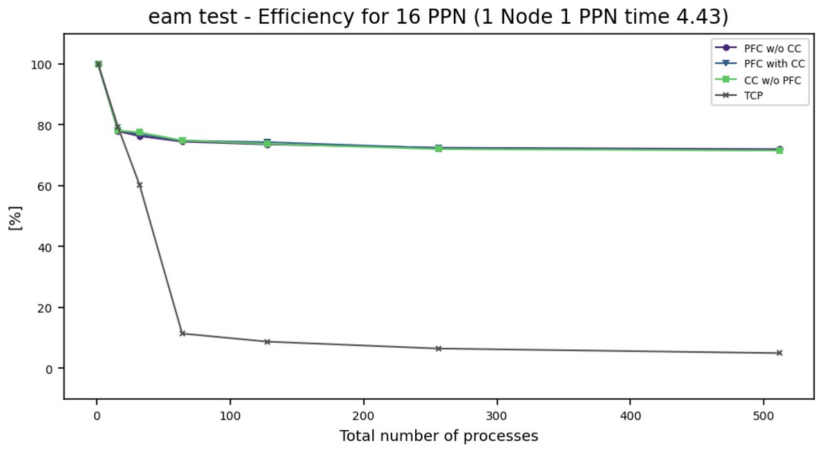
- **Deep Bench from Baidu** (https://github.com/baidu-research/DeepBench)
  - "uses the neural network libraries to benchmark the performance of basic operations "
    - basic operations - dense matrix multiplies, convolutions and communication
- **PARAM from Meta** (https://github.com/facebookresearch/param)
  - " repository of communication and compute micro-benchmarks as well as full workloads"
    - stand-alone compute and communication benchmarks using cuDNN, MKL, NCCL, MPI libraries
    - Application benchmarks – DLRM at this point
    - ML Framework – pytorch
- **ML benchmarks take away**
  - Focused on training accuracy, and/or performance
  - Communication specific is under micro benchmarks for specific collective operations
  - Desired: communication aspects that are relevant to ML/AI training
    - Similar questions as for HPC apply:
      - Can the communication aspect be profiled?
      - Can BM mimick COMM aspects without the computation behind them?

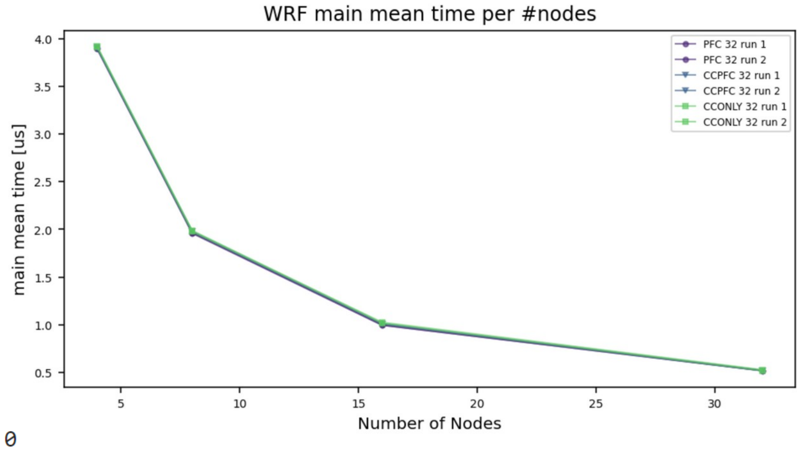
Congestion Control effectiveness

- **Application progress can be effected by congestion in different ways**
  - Unfairness between competing flows
  - Over-reaction of CC to the point switch link BW is underutilized some periods of time
  - Not capturing available BW fast enough (after congestion episode)
  - Interference between applications running concurrently – e.g. ML and storage
  - Interference between flows – if and when multiple communication ops running concurrently
  - Congestion at the receiver
    - Example: limited access to host memory over IO bus due to heavy memory access by CPU cores
  - Peak switch buffer usage could lead to drops in lossy network or PFC congestion spreading
- **These effects are heavily dependent on CC algorithm**
- **GPCNet is a relevant benchmark**
  - It is mostly focus on interference between nodes running different applications/tasks
    - As such it is effective with multi-switch cluster where flows from different nodes can cross-path
    - But not for interference between flows from same node and single switch cluster
  - Does not explicitly test the other effects
- Is there room for a CC focused benchmark?

Automation and Analysis

- **Automation for evaluating multiple BM results**
  - Setting up the system (hosts, NICs, Switches) in different modes
  - Launching Benchmark with various MPI libraries, APIs, options
  - Statistics collection from NIC and Switches
  - BM result parsing/filtering
- **Automation generates large amount of information**
- **Post processing and presentation is crucial**
  - Ability to compare results across different settings & run options
  - Measure scalability






```
Switch stats run 0 (printing only queue level above 64kB):
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

            Total Tx PFCs 26742, max PFCs on port ethernet1/1/12:5 - 1009
            Max Ingress Q Data level 2930.92kB on port 1/1/10:5_pg0
            Max Egress Q Data level 10688.34kB on port,tile 1/1/10:1_q3
            Total Egress ECN markings 831395703
NIC stats run 0
--------------
    run 0 *CONG* Total Rx PFCs to M Pkts Ratio 4.402

    run 0 *CONG* Total Tx cnps to K Pkt Ratio 68.19532
            Total CNPs 828620565
                    Total number of RNR NAKs on all nodes 28
```

```
Stats for node 10.13.231.138 run 0
        Total tx_read_req 178249
        Total tx_read_resp 10433980
        Total rx_read_req 178252
        Total rx_read_resp 10433725
        Total tx_send_req 643351
        Total rx_send_req 643352
         to_retransmits [[17.751182], [1]]
        Total tx_write_req 214245
        Total rx_write_req 214243
        Active QP: 2265
        Active CQ: 2177
        Active MR: 1641
        processes: 33
```

BROADCOM

Goals to possible additions to benchmark classes

- **Test patterns that are common to certain category of applications**
  - Examples: AI/ML, HPC, and Storage applications
  - Mix of collective operations with relevant message sizes
    - Allow tuning per application
    - Allow improved algorithm selection
- **Focus on communication part**
  - Without spending time in computation
  - Allow running iterations faster
- **Cover aspects of congestion control explicitly**
  - Can enable focused and effective effort at improvements in this area

New Benchmark Class – Proposal

- **Profiling**
  - Scope: common applications in each category
  - Identify communication patterns
    - Examples: allreduce or reduce+scatter in AI/ML, or boundary exchange in HPC stencil simulations
    - What operations are performed
    - How often
    - Distribution of Gaps/overlaps between communication phases
    - Size of communication group
    - Distribution of frequently used message sizes (vs. everything from 1B to 8MB or more)
    - Alignment/misalignment between ranks/nodes in starting operations
- **Build sets of few benchmarks per category (ML, HPC1, HPC2, Storage)**
  - A benchmark could include several collective ops and other communication ops
    - Chained or concurrent – depend on what typical for category
    - With message size and communication size characteristic to the category/scale
    - Repeated with fixed gap/overlap or with certain distribution of gaps/overlaps
  - Potentially allow for algorithm selection within the BM options

Congestion Control Focused Benchmarks - Examples

- **Loaded latency – N to 1 transmission concurrent with short message test measuring latency**
  - Test could be done at perftest level (e.g. ib_write_bw/lat), or
  - MPI level combining collective (alltoall) and point to point or another collective from same nodes
    - Similar to what’s done in GPCNet but both types threads on same node
    - Allow running with single switch as well as measure interference within the NIC
- **BW capture after congestion episode - N+M to 1**
  - N nodes start, M nodes join, then M nodes stop – measure N nodes utilization during 3’rd phase
  - Possibly measure convergence to fairness during second phase
- **Over-Reaction – utilization after large incast, measure BW ramp curve**
- **Switch memory usage**
  - Concurrent ‘latency measuring’ thread to estimate peak and steady state switch queue build up

Summary

- Communication benchmarking is important for HPC/ML applications
- Micro-Benchmarks cover only low-level operations and APIs
- Application benchmarks focus on specific applications compute/communications
- Communication patterns and CC aspects are not well covered
- Promote an intermediate class of BMs that can assist community in tuning, research and improvements in communications and congestion control

Thank You

BROADCOM®

connecting everything®