MVAPICH

MPI, PGAS and Hybrid MPI+PGAS Library

FFT in the Exascale: Performance Bottlenecks and Solutions

FFT in the Exascale: Opportunities and Challenges

by

Hari Subramoni

The Ohio State University

E-mail: subramoni.1@osu.edu

http://www.cse.ohio-state.edu/~subramon

Introduction and Motivation, and Challenges

- FFT's are critical components of several HPC applications and kernels
  - PSDNS, P3DFFT
- FFT's are typically communication intensive operations
  - Use All-to-all communication (MPI_Alltoall or MPI_Alltoallv)
- Performance of FFT's determine performance of applications relying on it
- Several factors impact the performance
  - What communication primitive is being used – MPI_Alltoall or MPI_Alltoallv?
    - What is the message size being used for communication – small or large?
  - Is there an attempt to overlap of computation and communication?
    - Who progresses the communication?

Broad Challenge

Can communication runtimes/middleware and applications/kernels be co-designed with these factors in mind to deliver scalable performance on emerging Exascale systems?

MPI_Alltoall or MPI_Alltoallv?

- Choice of primitives has a significant impact on communication performance
- Semantics of primitive limits the ability of communication runtime to perform optimized communication

MPI_Alltoall (2 nodes, 28ppn)

# Size	Avg Latency(us)
1	49.71
2	45.37
4	45.43
8	46.71
16	47.22
32	48.60
64	50.24
128	56.01
256	126.19
512	175.72
1024	265.68
2048	344.03
4096	444.77
8192	751.16
16384	1625.26
32768	2676.25
65536	5755.36
131072	11420.39
262144	22418.51
524288	44999.75
1048576	89322.51

Small messages performs worse with MPI_Alltoallv

MPI_Alltoallv (2 nodes, 28ppn)

# Size	Avg Latency(us)
1	113.93
2	96.56
4	95.92
8	95.72
16	97.39
32	98.19
64	134.40
128	108.89
256	109.58
512	116.13
1024	146.80
2048	226.79
4096	407.12
8192	809.81
16384	1601.48
32768	2705.95
65536	5685.02
131072	11114.28
262144	22104.01
524288	44589.75
1048576	88250.39

- Use “padding” and make use of MPI_Alltoall instead of MPI_Alltoallv

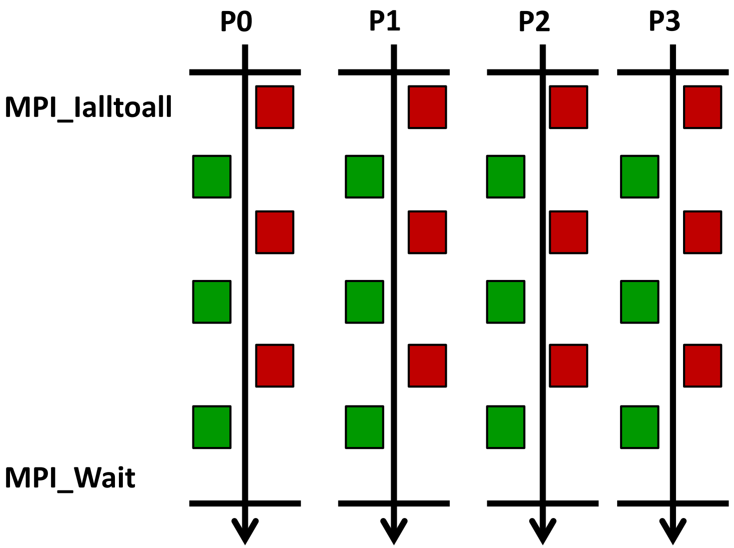
Overlap of Computation and Communication

- Concept simple and benefits obvious
  - Devil is in the details!
- Who progresses communication ???
- Different methods available for progress
  - Application progresses
    - MPI_Test / MPI_Probe / MPI_Iprobe
      - LibNBC (Hoefler et al.)
  - Separate software-based progress entity
    - One thread per process
      - LibNBC (Hoefler et al.)
      - MPICH (MPICH Team @ ANL)
    - One process per node (Functional Partitioning)
      - Hoefler et al., Nomura et al., Schneider et al., Kandalla et al.
  - Dedicated hardware progress engines
    - eg: CORE-Direct & SHARP from Mellanox
    - Venkata et al., Kandalla et al.

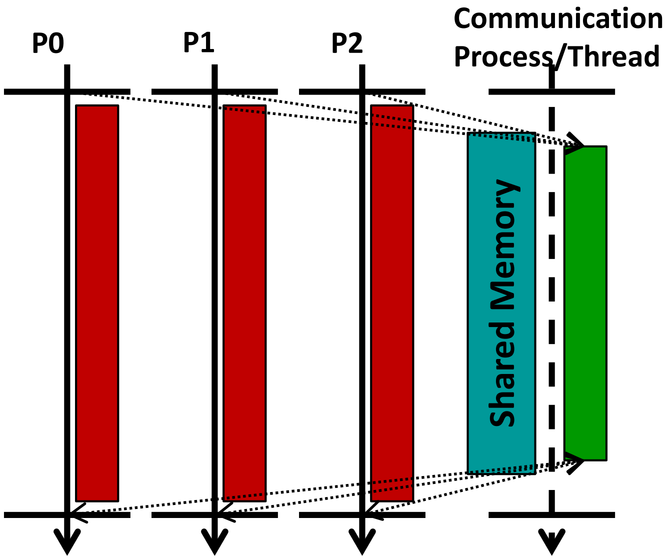
Problem Space for Designing NBCs

Communication

Computation



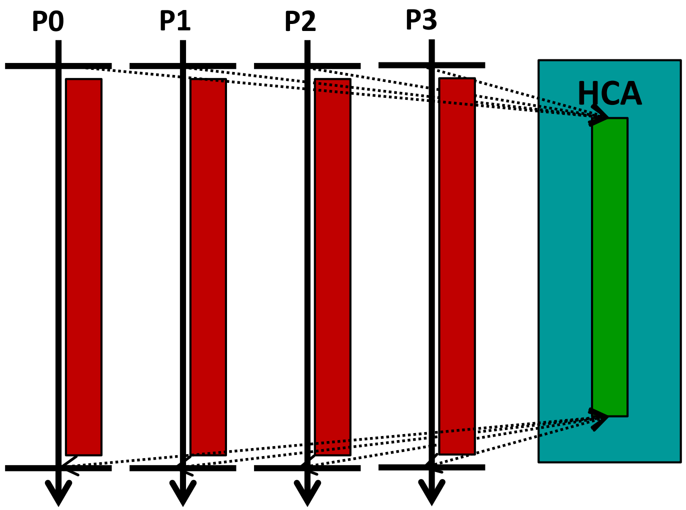
Application; LibNBC/MPICH



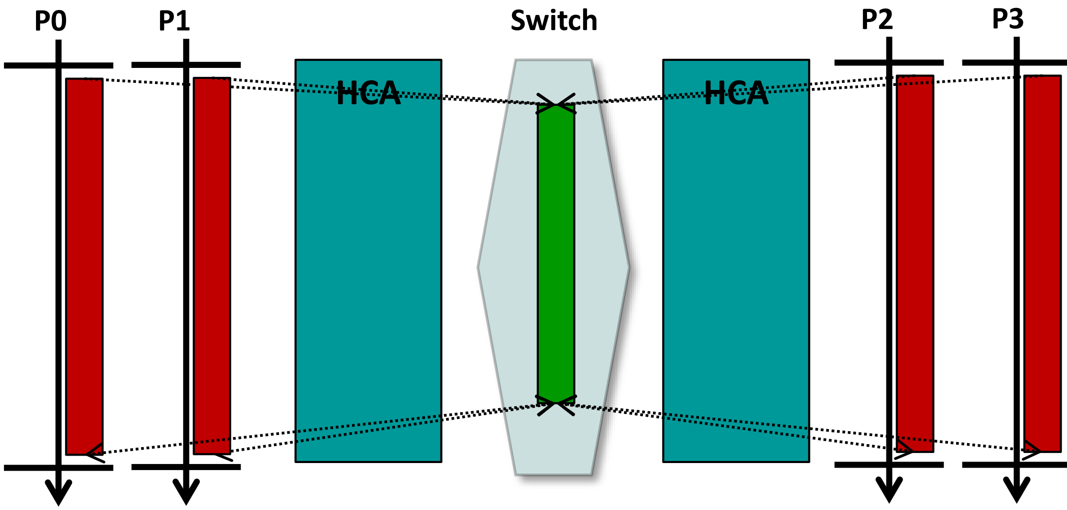
Functional Partitioning (FP), Auxiliary Progress Threads

Problem Space for Designing NBCs (Cont.)

Communication   Computation



CORE-Direct/Generic HCA-based Offload

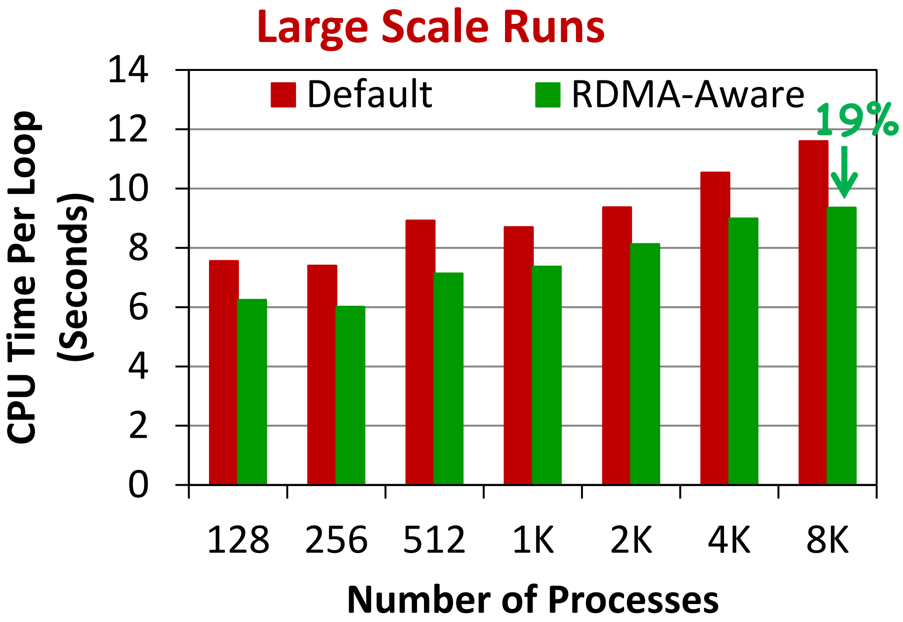


Switch-based Offload

Comparison of Communication Progress Schemes

Metric	Application; LibNBC/MPICH	FP/ Threads	HCA Offload	Switch Offload
Communication Latency	Good	Good	Good	Good
Computation/Communication Overlap	Poor	Good	Good	Good
Network Scalability	Good	Good	Fair	Good
Availability of Cores for Compute	Poor	Fair	Good	Good

Performance of P3DFFT Kernel



- Weak scaling experiments; problem size increases with job size
- HCA-Offloaded scheme delivers 19% improvement over Default @ 8,192 procs
  - Stampede@TACC (Sandybridge + IB FDR)
  - 512 nodes, 16 processes per node

Designing Non-Blocking Personalized Collectives with Near Perfect Overlap for RDMA-Enabled Clusters, H. Subramoni, A. Awan, K. Hamidouche, D. Pekurovsky, A. Venkatesh, S. Chakraborty, K. Tomko, and D. K. Panda, ISC '15, Jul 2015

Thank You!

subramon@cse.ohio-state.edu

Network-Based Computing Laboratory
http://nowlab.cse.ohio-state.edu/

The High-Performance MPI/PGAS Project
http://mvapich.cse.ohio-state.edu/

The High-Performance Big Data Project
http://hibd.cse.ohio-state.edu/

The High-Performance Deep Learning Project
http://hidl.cse.ohio-state.edu/