Design and Implementation of MPI Collective Operations for Large Message Communication on AMD GPUs

Chen-Chun Chen, Lang Xu, Olga Pearce, David Boehme, Hari Subramoni and Dhabaleswar K. (DK) Panda
The Ohio State University, Lawrence Livermore National Laboratory
{chen.10252, xu.3304}@osu.edu, {pearce8, boehme3}@llnl.gov, {subramon, panda}@cse.ohio-state.edu

Research Motivation

- The high demand for MPI Allreduce runtimes lies in providing high-speed computation and high-throughput communication in intra- and inter-node environments.
- The bandwidth gap between inter-node and intra-node communications creates bottlenecks in HPC systems. GPU-based compression can be leveraged to maximize effective bandwidth utilization.
- While compression-aware collectives work well on NVIDIA systems, the HPC landscape with AMD GPUs and HPE Slingshot demands further optimization studies.

Research Challenges

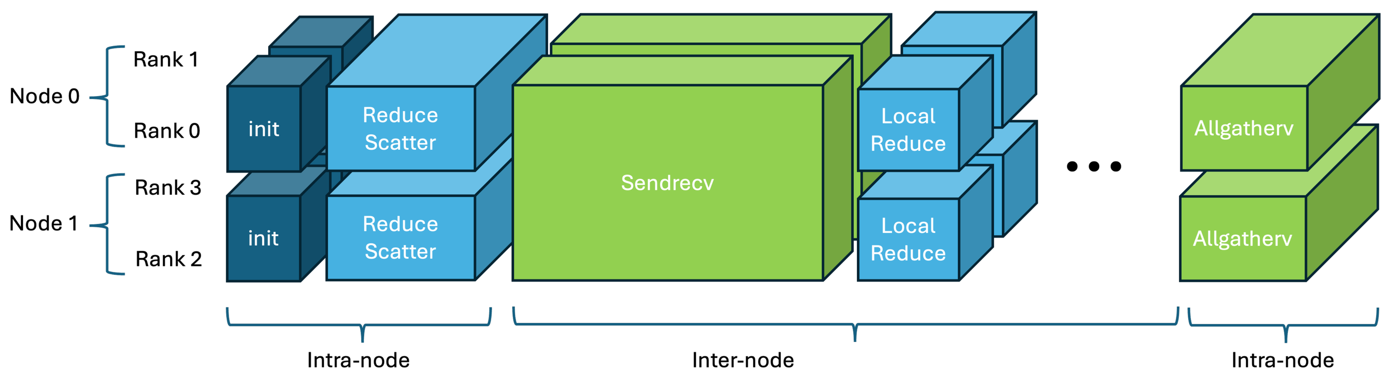
- What strategies and techniques are needed to design and implement a high-performance GPU-aware MPI Allreduce inter-node operation?
- How can we design compression-aware collectives that deliver net performance gains despite compression overhead while maintaining communication efficiency?

Overview of the Designs

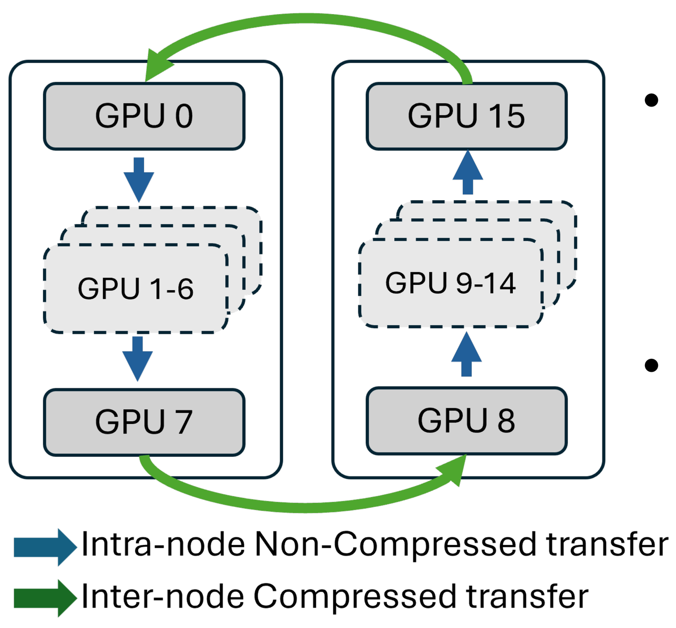
- **Computation-intensive collectives** using kernel-based approach:
  - Optimized HIP kernels for AMD GPUs.
  - Multi-leader two-level designs for inter-node Allreduce runtime.
  - A persistent GPU buffer to optimize the reduction operations in the second-level.
  - Early-triggered pipelined Allreduce algorithm to overlap intra-node and inter-node phases.
- **Heavy data-movement collectives** using compression:
  - Optimized HIP-aware lossy ZFP support.
  - Bandwidth-aware compression design.
  - Efficient computation-communication overlap.
  - Multiple Collectives support (Allgather, Alltoall).
- **Available in MVAPICH-Plus 4.1**

Multi-leader Two-level Designs for Allreduce

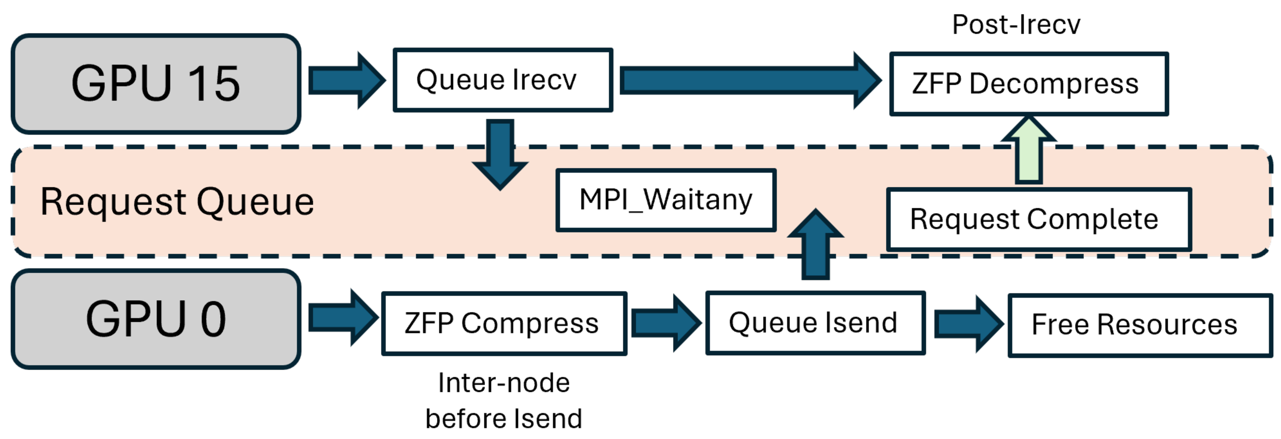
- Two-level Allreduce:
  - 1st-level: intra-node Reduce-Scatter and Allgatherv kernel.
  - 2nd-level: inter-node leader-based Allreduce.
- Multi-leader Designs:
  - Processes with the same local rank form a leader group to perform the second-level Allreduce.
- Optimization:
  - Persistent GPU buffer:
    - The *tmp* buffer for the local reduction step.
  - Early-triggered pipelined Allreduce algorithm:
    - Overlapping of the intra-node and inter-node phases.
    - Allowing inter-node communication step to occur earlier.



Compression Designs for Alltoall



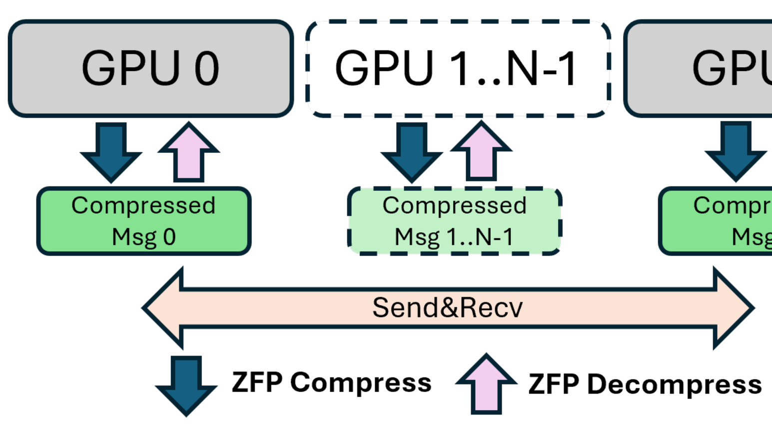
- **Selective Compression**
  Uncompressed data transfer for intra-node pairs while compressed for inter-node pairs.
- **Ring-based Alltoall**
  Receive source iterates clockwise, send destination iterates counter-clockwise. Prevents deadlock.



- **GPU-aware MPI**
  No need to stage to CPU buffers. We directly pass in GPU buffers for point-to-point operations
- **Non-Blocking Communication**
  Pairs are communicated using non-blocking MPI_Isend/Irecv.

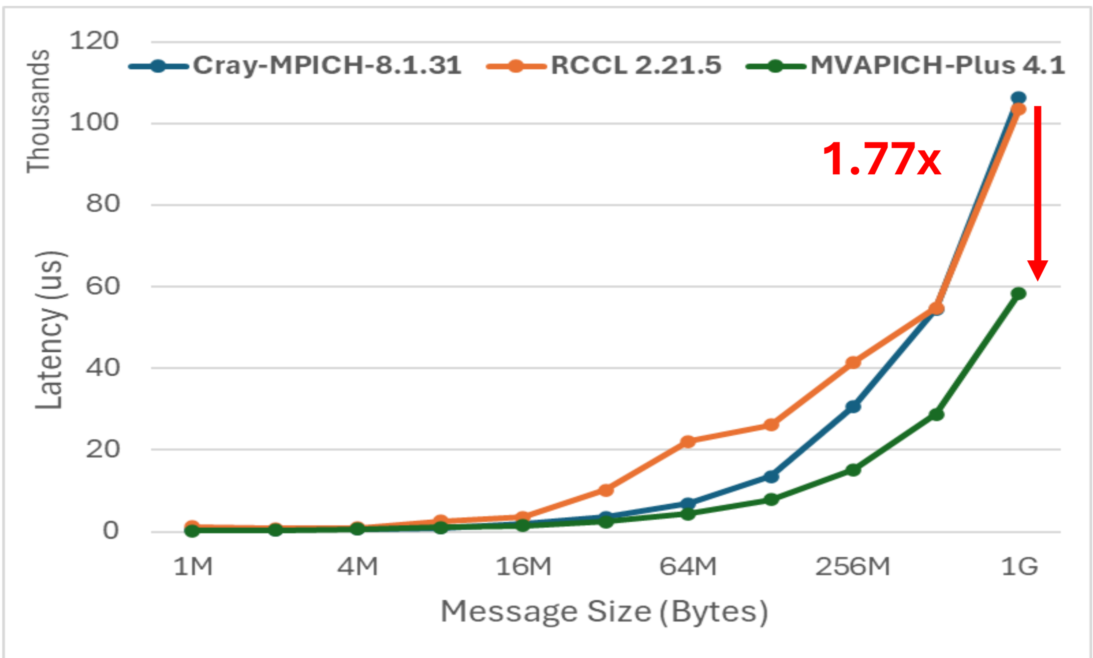
- **Optimized ZFP on ROCm 6**
  90GB/s decode and encode throughput. Less overhead

Compression Designs for Allgather

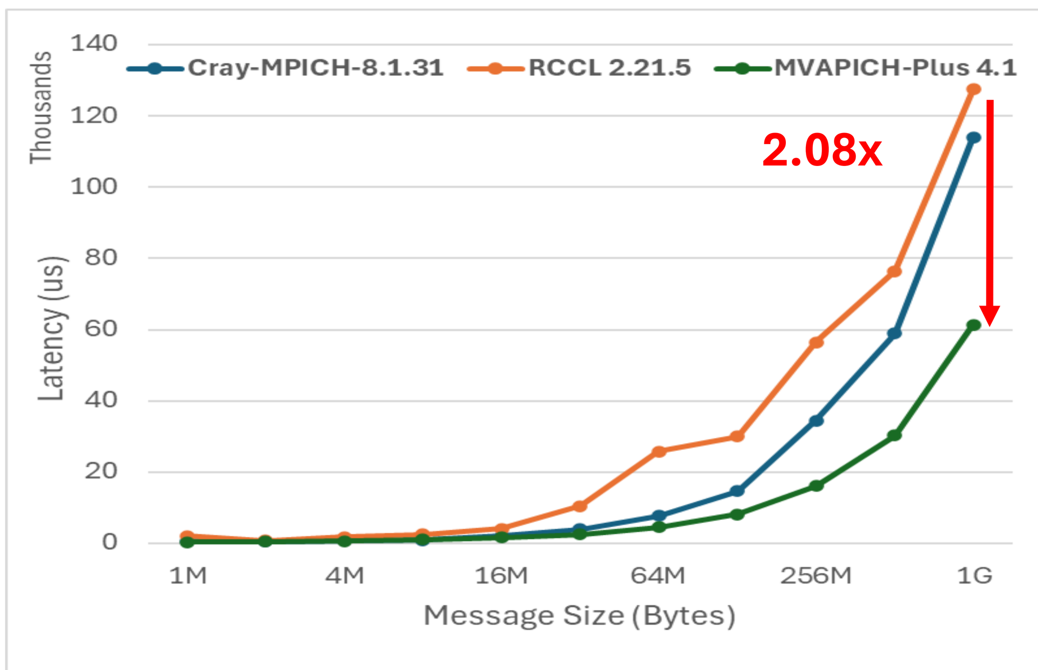


- **Allgather Online Compression**
  Different from Alltoall, we compress data once at the beginning and uses ring exchange to transfer compressed data. We decompress the message upon receiving
- **Selective Compression only across node boundaries (WIP)**

Benchmark-level Performance Evaluations - Allreduce



Allreduce - 8 Node (64 GPUs)

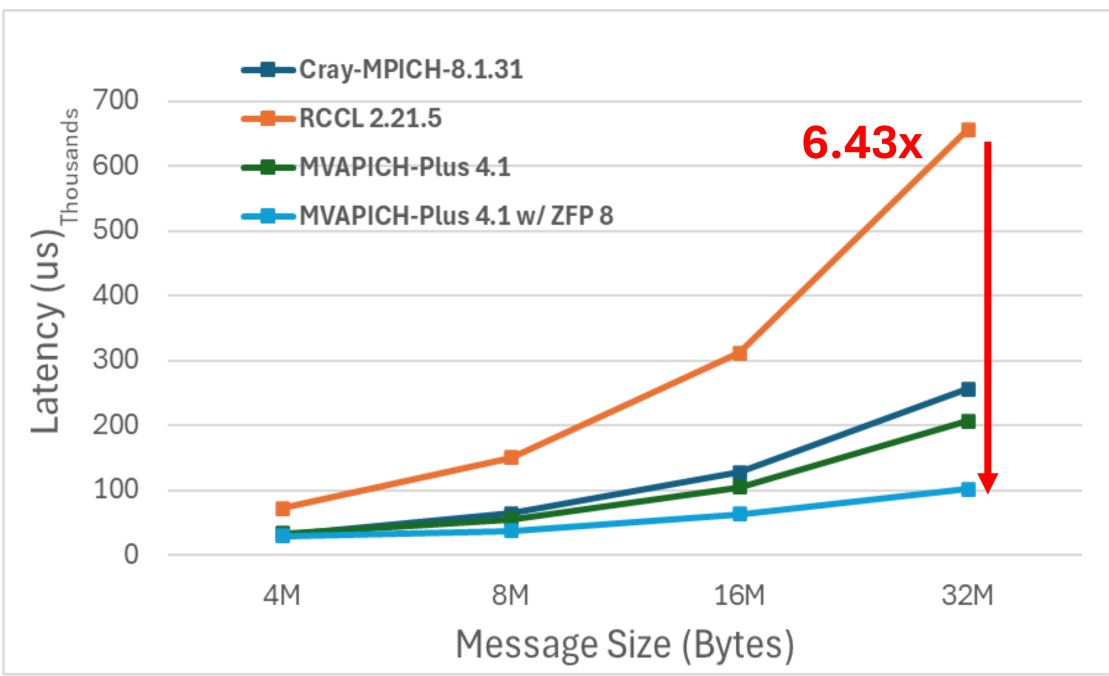


Allreduce - 16 Node (128 GPUs)

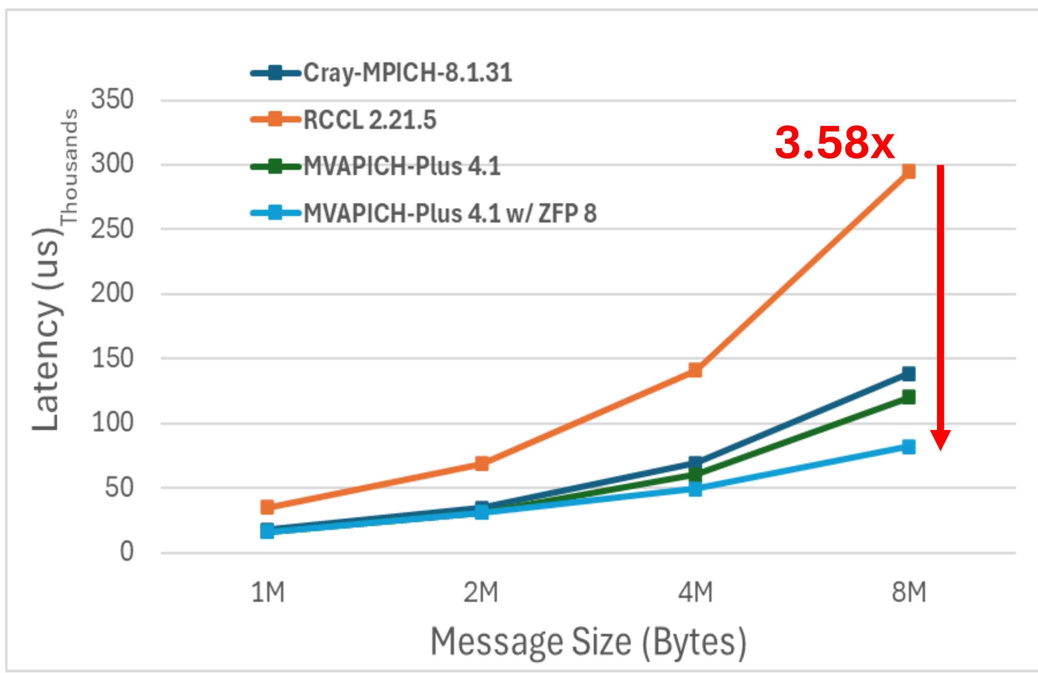
Experimental Setup - Frontier (OLCF)

Component	Configuration
GPU	4 AMD MI250X (8 GPU(GCD)s)
Device Memory per GPU	64 GB HBM2e
CPU	AMD EPYC 7A53
Memory	512 GB DDR4
Sockets	1
Core per Sockets	64
Inter-connection	4 HPE Slingshot 200 Gbps NICs
Libraries	MVAPICH-Plus 4.1 ROCm 6.3.1 Cray MPICH 8.1.31 RCCL 2.21.5 + OFI

Benchmark-level Performance Evaluations - Alltoall

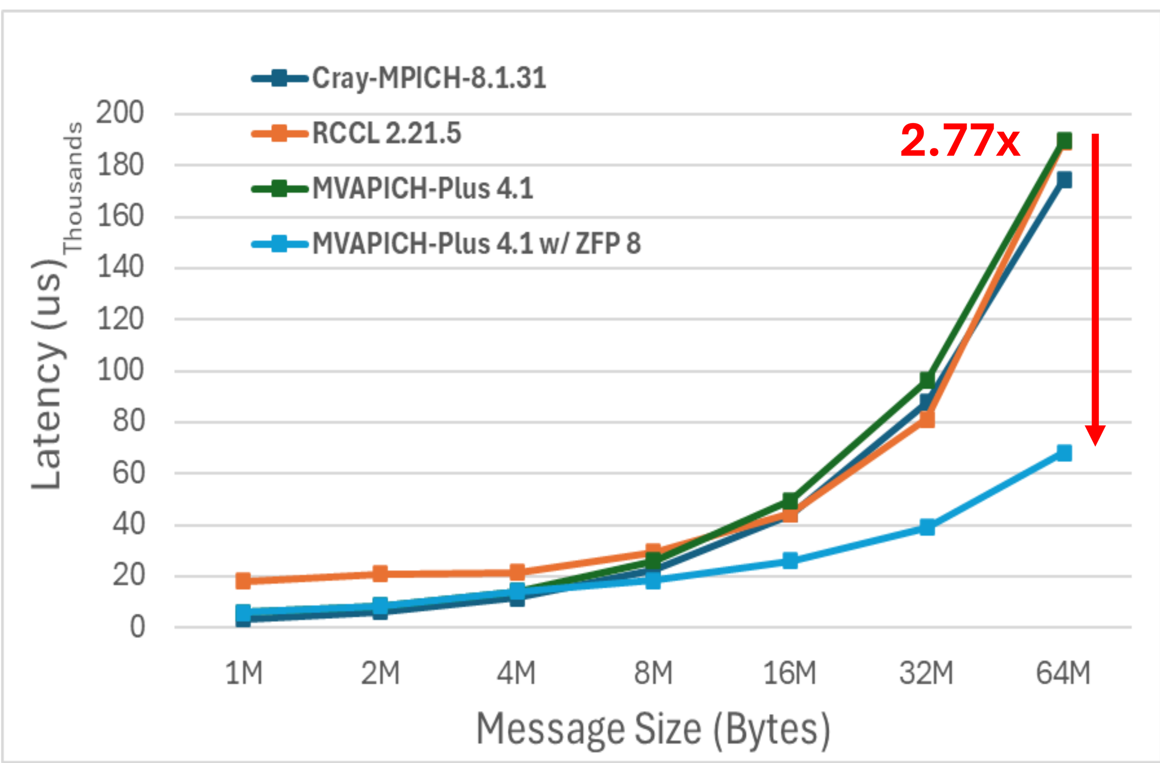


Alltoall - 8 Node (64 GPUs)

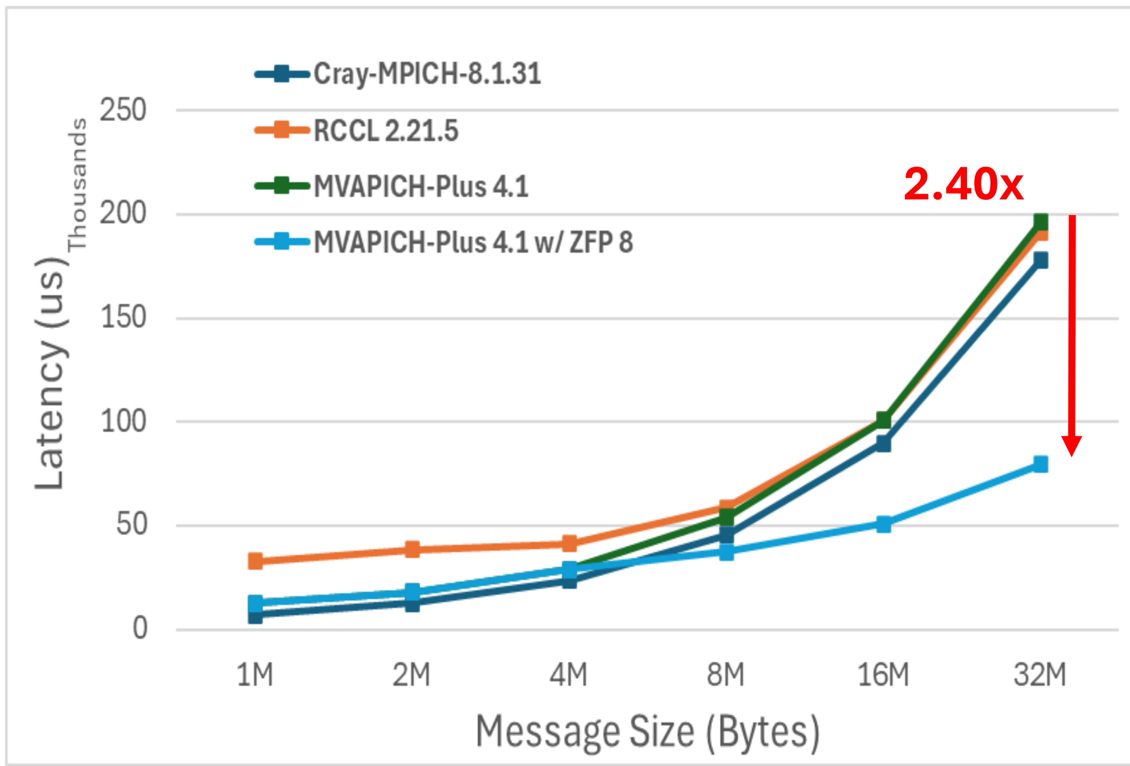


Alltoall - 16 Node (128 GPUs)

Benchmark-level Performance Evaluations - Allgather



Allgather - 8 Node (64 GPUs)



Allgather - 16 Node (128 GPUs)

Conclusion

- Implementation of multi-leader two-level Allreduce designs uses a kernel-based approach, optimized with persistent GPU buffers and an early-triggered pipelined method for AMD GPU systems.
- Implementation of efficient non-blocking compression-aware collectives (Alltoall and Allgather). The design supports asynchronous communication and ZFP lossy encoding and decoding.
- **Benchmark results (16 Nodes)**:
  - **Allreduce: 2.08x** over RCCL
  - **Alltoall: 3.58x** over RCCL
  - **Allgather: 2.40x** over RCCL

MVAPICH
THE OHIO STATE UNIVERSITY

References
1. C. Chen, J. Yao, L. Xu, H. Subramoni, D. Panda, “Unified Designs of Multi-rail-aware MPI Allreduce and Alltoall Operations Across Diverse GPU and Interconnect Systems”, IPDPS 25
2. Q. Zhou, Q. Anthony, L. Xu, A. Shafi, M. Abduljabbar, H. Subramoni, D. Panda, “Accelerating Distributed Deep Learning Training with Compression Assisted Allgather and Reduce-Scatter Communication”, IPDPS 23
3. Q. Zhou, P. Kousha, Q. Anthony, K. Khorassani, A. Shafi, H. Subramoni, D. Panda, “Accelerating MPI All-to-All Communication with Online Compression on Modern GPU Clusters”, ISC 22
4. Q. Zhou, C. Chu, N. Senthil Kumar, P. Kousha, M. Ghazimirsaeed, H. Subramoni, D. Panda, “Designing High-Performance MPI Libraries with On-the-fly Compression for Modern GPU Clusters”, IPDPS 21

Acknowledgements
- NSF grants #2311830, #2312927, #2323116, #2415201, LLNL contract #B668423 and XRAC grant #NCR-130002
- This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344
- Reviewed and released under LLNL-POST-2006313