Benefits of On-the-Fly Compression on GPU-to-GPU Communication for HPC and Data Science Applications

Presentation at MUG ‘21

Qinghua Zhou

Network Based Computing Laboratory (NBCL)
Dept. of Computer Science and Engineering, The Ohio State University

Follow us on

zhou.2595@osu.edu

https://twitter.com/mvapich
Outline

• Motivation
• Focus of the Work
• On-the-fly Compression Designs
  – Compression algorithms
  – Framework for GPU-based on-the-fly compression
• Performance Evaluation
• Conclusions and Future Work
Motivation

- For HPC and data science applications on modern GPU clusters
  - With larger problem sizes, applications exchange orders of magnitude more data on the network
  - Leads to significant increase in communication times for these applications on larger scale (AWP-ODC)
  - On modern HPC systems, there is disparity between intra-node and inter-node GPU communication bandwidths that prevents efficient scaling of applications on larger GPU systems

- CUDA-Aware MPI libraries saturate the bandwidth of IB network
- Compression can reduce the data size and lower the pressure on network with limited bandwidth

Motivation

(a) Disparity between intra-node and inter-node GPU communication on Sierra OpenPOWER supercomputer [1]

(b) Saturated bandwidth at large message size

Focus of the Work

- Designing **on-the-fly** message compression schemes in an MPI library:
  - The **first of its kind** GPU-based compression design is implemented using MVAPICH2-GDR
- Optimizing the existing GPU based compression algorithms
- Accelerating **point-to-point** communication performance of transferring large GPU-to-GPU data
- Demonstrating performance benefits for two categories of applications:
  - AWP-ODC (HPC) [2]
  - Dask (Data science) [3]

---


Compression Algorithms for Floating-point Data

- **Lossless**
  - Fpzip: CPU, supports double (64 bit) & single (32 bit) precision FP, low throughput
  - FPC: CPU, supports double & single precision FP, low throughput
  - ISOBAR: CPU, supports double & single precision FP, low throughput
  - GFC: GPU, supports double precision FP, high throughput
  - **MPC** [4]: GPU, supports double & single precision FP, *high throughput*

- **Lossy**
  - ZFP [5]: GPU, supports double & single precision FP, *high throughput*
  - SZ: GPU, supports double & single precision FP, high throughput

---


Framework for GPU-based on-the-fly compression

- Compression algorithms **MPC** and **ZFP** are integrated into **MVAPICH2-GDR** with further optimization.
- Rendezvous protocol is used to send the header data and compressed data.

---

# Experimental Environment

<table>
<thead>
<tr>
<th>Cluster Specs</th>
<th>Frontera Longhorn</th>
<th>Frontera Liquid</th>
<th>Lassen</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU Processor</td>
<td>Dual-socket IBM POWER9 AC922 2.3GHz, 20 Cores/socket</td>
<td>Dual-socket Intel Xeon E5-2620 2.10GHz, 8 Cores/socket</td>
<td>Dual-socket IBM POWER9 AC922 3.14GHz, 44 Cores/Socket</td>
</tr>
<tr>
<td>System Memory</td>
<td>256 GB</td>
<td>384 GB</td>
<td>256 GB</td>
</tr>
<tr>
<td>GPU Processor</td>
<td>4 NVIDIA Tesla V100</td>
<td>4 NVIDIA Quadro RTX 5000</td>
<td>4 NVIDIA Tesla V100</td>
</tr>
<tr>
<td>GPU Memory</td>
<td>4 x 16 GB</td>
<td>4 x 16 GB</td>
<td>4 x 16 GB</td>
</tr>
<tr>
<td>Interconnects between CPU and GPU</td>
<td>NVLink-2 (one-way 75 GB/s)</td>
<td>PCIe Gen3 x16 and x64 switches (one-way 16 GB/s)</td>
<td>NVLink-2 (one-way 75 GB/s)</td>
</tr>
<tr>
<td>Interconnects between GPUs</td>
<td>NVLink-2 (one-way 75 GB/s)</td>
<td>PCIe Gen3 x16 and x64 switches (one-way 16 GB/s)</td>
<td>NVLink-2 (one-way 75 GB/s)</td>
</tr>
<tr>
<td>Interconnects between nodes</td>
<td>Mellanox InfiniBand EDR (one-way 12.5 GB/s)</td>
<td>Mellanox InfiniBand FDR (one-way 7 GB/s)</td>
<td>Dual-rail Mellanox InfiniBand EDR (one-way 25 GB/s)</td>
</tr>
<tr>
<td>Operating System</td>
<td>RHEL 7.6 (4.14.0-115.10.1.1)</td>
<td>CentOS 7.6.1810 (3.10.0-957.27.2.el7)</td>
<td>RHEL 7.3 (4.14.0-115.10.1.1)</td>
</tr>
<tr>
<td>NVIDIA Driver Version</td>
<td>440.33.01</td>
<td>430.40</td>
<td>418.87.00</td>
</tr>
</tbody>
</table>
Inter-node GPU-GPU latency

- OSU Micro Benchmark for MPC-OPT and ZFP-OPT on RTX5000 and V100 GPU nodes
  - MPC-OPT and ZFP-OPT are optimized and integrated into MVAPICH2-GDR library
- MPC-OPT reduced latency up to 77.1% on RTX5000 and 62.5% on V100 at 32Mbytes
- ZFP-OPT reduced latency up to 83.1% on RTX5000 and 78.3% on V100 at 32Mbytes

![Graph showing latency comparison between baseline, MPC-OPT, and ZFP-OPT for different message sizes on Frontra-RTX5000 and Longhorn-V100.](image)
Intra-node GPU-GPU latency

- MPC-OPT reduced latency up to **60.6%** on RTX5000 at 32Mbytes
  - High-speed NVLink on Longhorn is faster than the MPC compression/decompression
- ZFP-OPT reduced latency up to **79.8%** on RTX5000 and **40.5%** on V100 at 32Mbytes
Collective Operations

- **MPI_Bcast**  
  MPC-OPT: 57% benefits on msg_sppm, ZFP-OPT(rate:4): 85% benefits

- **MPI_Allgather**  
  MPC-OPT: 30% benefits on msg_sppm, ZFP-OPT(rate:4): 73% benefits
Application Results (AWP-ODC)

- Weak-Scaling of HPC application **AWP-ODC** on Lassen cluster (V100 nodes)
- MPC-OPT achieves up to **+18%** GPU computing flops, **-15%** runtime per timestep
- ZFP-OPT achieves up to **+35%** GPU computing flops, **-26%** runtime per timestep (rate=8, compression ratio=4)
Application Results (Dask)

- Data science framework **Dask** on RI2 cluster (V100 nodes)
- Dask benchmark creates cuPy array and distributes its chunks across Dask workers
- ZFP-OPT achieves up to **1.56x** throughput, **-37%** runtime (rate=8, compression ratio=4)

![Graph showing throughput and execution time for baseline and ZFP-OPT with different number of Dask workers.](image)

(cuPy Dims: 10Kx10K, Chunk size: 1K)
Conclusions and Future Work

• Presented on-the-fly compression techniques for optimizing GPU-to-GPU communication in an MPI library

• Enhanced GPU based pt2pt communication in MVAPICH2-GDR with optimized MPC and ZFP

<table>
<thead>
<tr>
<th>Compression</th>
<th>Benchmark Benefits (GPU-GPU Latency)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MPI Bcast</td>
</tr>
<tr>
<td>MPC-OPT</td>
<td>77.1%</td>
</tr>
<tr>
<td>ZFP-OPT</td>
<td>83.1%</td>
</tr>
</tbody>
</table>

• Application-level Benefits
  - AWP-ODC: up to 18% (MPC-OPT on 512 GPUs) and 35% (ZFP-OPT on 128 GPUs) improvement of GPU computing flops
  - Dask: up to 1.56x speedup of throughput and 37% reduced runtime

• Future work
  - Study and incorporate more GPU-based compression algorithms (e.g. cuSZ, NVIDIA nvCOMP, etc.)
Thank You!

Zhou.2595@osu.edu

Network-Based Computing Laboratory

http://nowlab.cse.ohio-state.edu/

Follow us on

https://twitter.com/mvapich

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/