Bardia Mojra – 1000766739

02/08/2022 – Assignment 01 – Essay 02

Supercomputer #3 – Sierra Supercomputer

High-Performance MPI Libraries with On-the-fly Compression for Modern GPU Clusters

Sierra is the world’s third-fastest supercomputer and it is located at the Lawrence Livermore National Laboratory (LLNL). It became operational in 2018 in order to serve as the main high-performance supercomputer (HPC) system for the Advanced Simulation and Computing (ASC) program, under the National Nuclear Security Administration. Moreover, Sierra serves as one of the two new flagship exascale architecture supercomputers which are designed with the latest NVIDIA GPU-accelerated systems based on the IBM OpenPOWER platform.

As a result of The Collaboration of Oak Ridge, Argonne, and Livermore (CORAL) national laboratories, Sierra and Summit were designed with exascale computing as their prime goal. Sierra features a heterogeneous compute model that couples one of the world's fastest serial processors with one of the world’s fastest parallel processors. Each node has an IBM Power9 CPU, 4 NVIDIA V100 GPUs, and Mellanox Infiniband interconnect between the nodes. Although GPUs on Sierra are interconnected via NVIDIA NVLink connections (up to 75 GB/s), the designers used Infiniband (IB) with lower bandwidth of 25 GB/s for inter-node communication lines. In this essay, I review recent developments in high-performance message passing interface (MPI) libraries with on-the-fly compression for modern GPU clusters such as the Sierra supercomputer. Sierra is designed with 4320 compute nodes and in total, they deliver 125 PFLOPs of theoretical peak performance and actual performance of 96 PFLOPs, per LINPACK benchmark [2]. Figure 1 shows the intra-node and inter-node GPU communication lines on the Sierra supercomputer [1].

In [1], the authors introduce their redesigned MVAPICH2 MPI library which supports GPU-aware on-the-fly compression and decompression and delivers up to 37% improvement in GPU compute flops. They optimize existing lossless and lossy compression algorithms, MPC and ZFP, to deliver improved on-the-fly message compression at both microbench and application levels. The authors acknowledge and cite numerous contributions from the research community and claim to introduce the first MPI library that uses GPU-based compression schemes to significantly improve GPU communication performance for various MPI primitives and HPC applications.

Message passing or MPI-based parallel computing is well-known and widely used by the scientific and research community. In recent years, demand for HPC has increased significantly which has led to the development of highly efficient MPI libraries. Although such libraries are highly optimized and one can develop an efficient and low latency parallel computing system given modern hardware, compute data transfer remains one of the main bottlenecks.

In recent years, there has been a push to improve communication performance for data resident on the GPU; but in most cases, the communication bandwidth is already maxed out to reduce latency. The authors also argue even though GPU-GPU communication data rate as high as 75 GB/s has been achieved by NVIDIA NVLink, most supercomputers such as Sierra use Mellanox Infiniband with only 25 GB/s bandwidth for intra-node communication. For the time being, we should explore other avenues for advancing better performance [1].

As described in [1], they pose the question, what other techniques besides increasing communication bandwidth (which is already optimized and limited by hardware) could be used to reduce the communication time of compute data for HPC application on dense GPU clusters? To answer this question, they examined existing CPU-based and GPU-based compression algorithms and identified areas that could be improved for execution time. They examined a series of latest data compression designs such as MPC and ZFP against various data compression benchmarks. Moreover, the authors introduce MPC-OPT and ZFP-OPT compression algorithms with %19 and %37 improvements in performance on GPU clusters. These algorithms are integrated into their MVAPICH2 MPI library featuring *on-the-fly* compression, which means the algorithm avoids unnecessary memory allocations and function calls.

In order to understand their proposed compression scheme; first, we must understand how existing compression algorithms work with existing MPI communication middleware. Current compression schemes for GPU data transfer consist of typical communication handshake calls such as Request-To-Send (RTS) and Clear-To-Send (CTS) packets. Depending on which compression scheme to be used, there is normally an acceptable range for the packet size in order to yield the desired compression ratio, given a constant header size. Often multiple packets are sent to transmit compressed data, compression parameters, transmission, and compute information which all add up to saturate communication bandwidth. A reduction in the number of packets sent for each data transmission process could induce an incremental improvement in the GPU-GPU data transmission rate.

In their proposed framework [1], first, they reduce unnecessarily message exchanges by attaching compression parameters to RTS packet. This is a clever idea since compression parameters are often very small in size and will be transmitted with ease. Moreover, they take advantage of the latest capabilities of supercomputers such as Sierra which allows the GPU to compress and transmit data by itself while the CPU proceeds with executing communication processes. Although, as mentioned in the paper this requires additional buffer allocation on the GPU. For this reason, the compressed data or application data size to be transmitted is also attached to RTS packet for the receiver to adjust its dynamic buffer pool, which is discussed in more detail in the next section.

The authors [1] propose new techniques that take advantage of NVIDIA and IBM’s latest advances on the hardware side. First, they used a feature to allow GPU and CPU to access each other’s memory which is similar to what NVIDIA describes are Unified Memory architecture. This allows them to reduce overall read and write operations which contributes to the reduction in latency in both compression and transmission operations. This is the main contributing factor for their proposed novel *on-the-fly* compression feature for GPU clusters. Additionally, they introduce a dynamic buffer pool that adjusts to application demands and on larger scale results in optimized use of memory on GPU and reduction in overall compute time. Moreover, they eliminate memory expensive function calls and use low-latency routines to copy data from GPU memory instead of loading it to CPU memory first. This reduces this specific function call by a factor of 5-20 times. Figures 3, 4, and 7 show the proposed framework.

The proposed advancements in [1] are made possible by hardware features deployed on supercomputers such as Sierra. Advancements in both hardware and software play a crucial role in advancing the general state of science, as it depends on our total compute power. This is an example where Sierra plays its role in advancing science by providing state-of-the-art HPC to the research community.

Corresponding Authors:

1. {zhou.2595, chu.368, senthilkumar.16, kousha.2, ghazimirsaeed.3, subramoni.1, panda.2}@osu.edu (Q. Zhou, C. Chu, N. S. Kumar, P. Kousha, S. M. Ghazimirsaeed, H. Subramoni, and D. K. Panda)
2. zwm-dcs@tsinghua.edu.cn (W. Zheng).
3. {chu.368,kousha.2,awan.10,shafiekhorassani.1}@osu.edu, subramon,[panda}@cse.ohio-state.edu](mailto:panda@cse.ohio-state.edu), (C. Chu, P. Kousha, A. A. Awan, K. S. Khorassani, H. Subramoni, and D. K. Panda)

References:

1. Zhou, Q and Chu, C and Kumar, NS and Kousha, P and Ghazimirsaeed, SM and Subramoni, H and Panda, DK, “Designing High-Performance MPI Libraries with On-the-fly Compression for Modern GPU Clusters,” IEEE International Parallel and Distributed Processing Symposium (IPDPS), 2021, PP. 444-453, IEEE.
2. Zheng, “Research trend of large-scale supercomputers and applications from the TOP500 and Gordon Bell Prize,” Science China Information Sciences, 2020, PP. 1-14, Springer.
3. Chu, Ching-Hsiang and Kousha, Pouya and Awan, Ammar Ahmad and Khorassani, Kawthar Shafie and Subramoni, Hari and Panda, Dhabaleswar K, “NV-group: Link-efficient reduction for distributed deep learning on modern dense GPU systems,” Proceedings of the 34th ACM International Conference on Supercomputing, 2020, PP. 1-12, ACM.







