OOKAMI PROJECT APPLICATION

Date: 05/01/2021–04/30/2022

Project Title: Improving Performance of Lossy Compression for Extreme-scale Scientific Applications on A64FX

Usage:

- Testbed

Principal Investigator: Dingwen Tao

- University/Company/Institute: Washington State University
- Mailing address including country: School of EECS, 355 NE Spokane St, Pullman, WA 99163, USA
- Phone number: +1 (509) 335-6602
- Email: dingwen.tao@wsu.edu
- Webpage: https://www.dingwentao.com/

Names & Email of initial project users:

Jiannan Tian, jiannan.tian@wsu.edu
Chengming Zhang, chengming.zhang@wsu.edu
Daoce Wang, daoce.wang@wsu.edu
Sheng Di, sdi1@anl.gov

Usage Description:

SZ is a widely used modular parametrizable lossy compressor for scientific data (https://szcompressor.org/). It has been ported to multiple architectures with massive parallelism, including GPU (https://github.com/szcompressor/cuSZ) and FPGA (https://github.com/szcompressor/SZ_HLS). Among the actively developing variants, CUDA-implemented SZ (called cuSZ) is a state-of-the-art lossy compressor on GPU with good compression speed and quality with a central hub of parallelization technique tryouts. Our preliminary work has demonstrated that both high memory bandwidth and the high degree of parallelism are keys to the attainable throughput. Our long-term optimization goal is to improve the end-to-end compression performance in order to help with I/O

and in-memory compression for extreme-scale scientific applications. cuSZ's plain CUDA-C programming enables portability to other existing architectures in HPC. There are two key aspects that makes A64FX CPU distinct and an ideal testbed for our next-generation SZ compressor.

1. The high-latency high-throughput GPU architecture can be naturally unfriendly to hard-to-parallelize kernels such as Huffman encoding and decoding. Based on our existing study (Tian et al., IPDPS'21), CPU has much lower latency than GPU on those kernels.

2. The data movements between host and device are undesired in general because of the relatively high latency of CPU-GPU interconnect. Also, our $\mathcal{O}(n)$-complexity compressing algorithm is essentially memory-bounded. Thus, the high-bandwidth unified memory of A64FX CPU (which has four core memory groups and each is connected to the respective HBM2) would significantly alleviate this issue.

These two aspects are entangled rather than mutually exclusive. Moreover, practice GPU programming mandates using kernel as a boundary of cache validity and context switch. At a larger scale in a system view, heterogeneous paradigm is much emphasized for the best possible performance, which further complexifies the performance analysis and system design.

In this project, we plan to test our cuSZ lossy compressor on A64FX CPU and attempt to optimize its performance if needed. We will compare the performance on different CPU and GPU architectures, especially A64FX CPU and Nvidia A100 GPU. This will guide us to design a high-performance scientific lossy compressor for different incoming exascale supercomputers.

Computational Resources:

- Total node hours per year: 1,000 node hours per year in approximate.
- Size (nodes) and duration (hours) for a typical batch job: 1 node for one hour per batch job in typical.
- Disk space (home, project, scratch): 10 GBs for home space, 500 GBs for project space, no requirement on scratch space.

Personnel Resources:

Required software:

If your research is supported by US federal agencies:

- Agency: National Science Foundation
- Grant number(s): 2042084