FZ-GPU

A **Fast** and **High-Ratio Lossy** Compressor for **Scientific Computing** Applications on **GPUs**

Boyuan Zhang *	*Indiana University*
Jiannan Tian *	*Indiana University*
Sheng Di	*Argonne National Laboratory*
Xiaodong Yu	*Argonne National Laboratory*
Yunhe Feng	*University of North Texas Denton*
Xin Liang	*University of Kentucky*
Dingwen Tao	*Indiana University*
Franck Cappello	*Argonne National Laboratory*

INDIANA UNIVERSITY BLOOMINGTON

Argonne NATIONAL LABORATORY

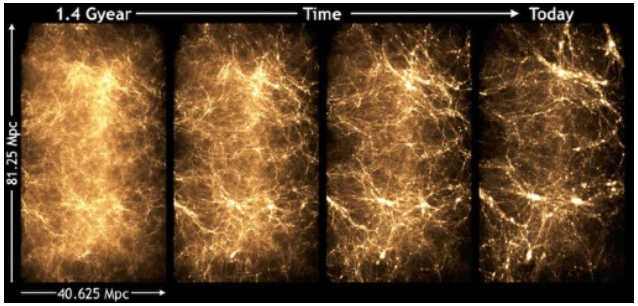
The 32nd International Symposium on High-Performance Parallel and Distributed Computing
Orlando, Florida, United States, June 21, 2023

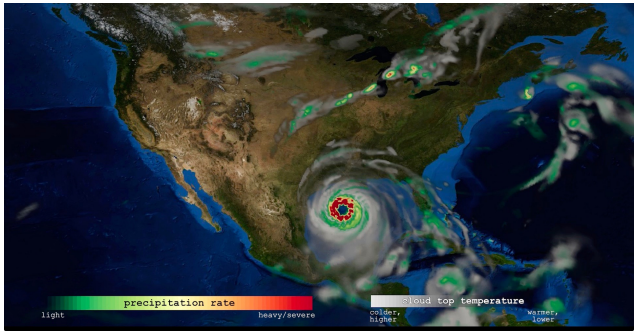
Big Data Problem for Scientific Applications

INDIANA UNIVERSITY BLOOMINGTON

Argonne NATIONAL LABORATORY

application	data scale	bottleneck	reduce by
HACC cosmology simulation	**20 PB** one-trillion-particle	**use up filesystem (26 PB in total)** Mira@ANL	**10×** in need
CESM climate simulation	**50%** vs 20% storage in hardware budget, **2017** vs 2013	**5h30m to store** NSF Blue Waters 1-TBps I/O	**10×** in need





Lossy Compressors for HPC

2:1 (FP-type)	**10:1 or higher**
lossless on scientific datasets	reduction ratio in need
industry lossy compressor (JPEG)	high in reduction rate, but **not** suitable for **HPC**
need **diverse** compression modes	1) absolute error bound (infinity-norm) 2) pointwise relative error bound 3) RMSE error bound (2-norm) 4) fixed bitrate 5) satisfying post-analysis requirements

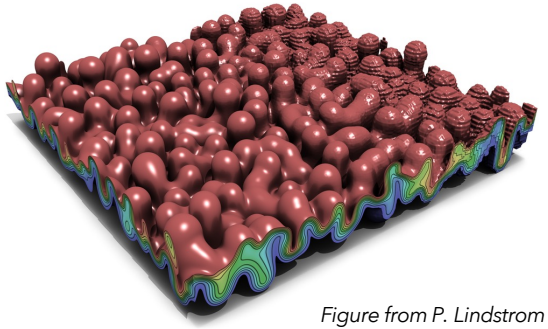


Figure from P. Lindstrom (LLNL)

Lossy compression for scientific data at varying reduction ratio (10:1 to 250:1, left to right)

Issues of the SOTA Compressors

SOTA GPU error-bounded lossy compressors suffer from:

Low quality rooted in fixed-rate method

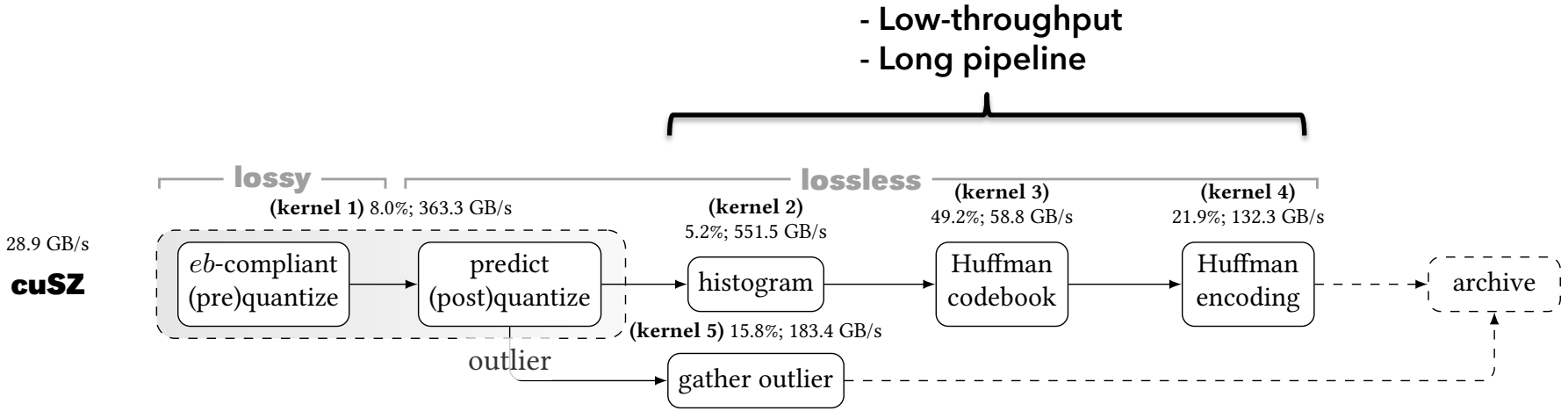
- Seen in cuZFP with limited bit budget
- Slightly higher in throughput compared with cuSZ and MGARD-GPU
- At a high cost of much lower compression quality

Issues of the SOTA Compressors

SOTA GPU error-bounded lossy compressors suffer from:

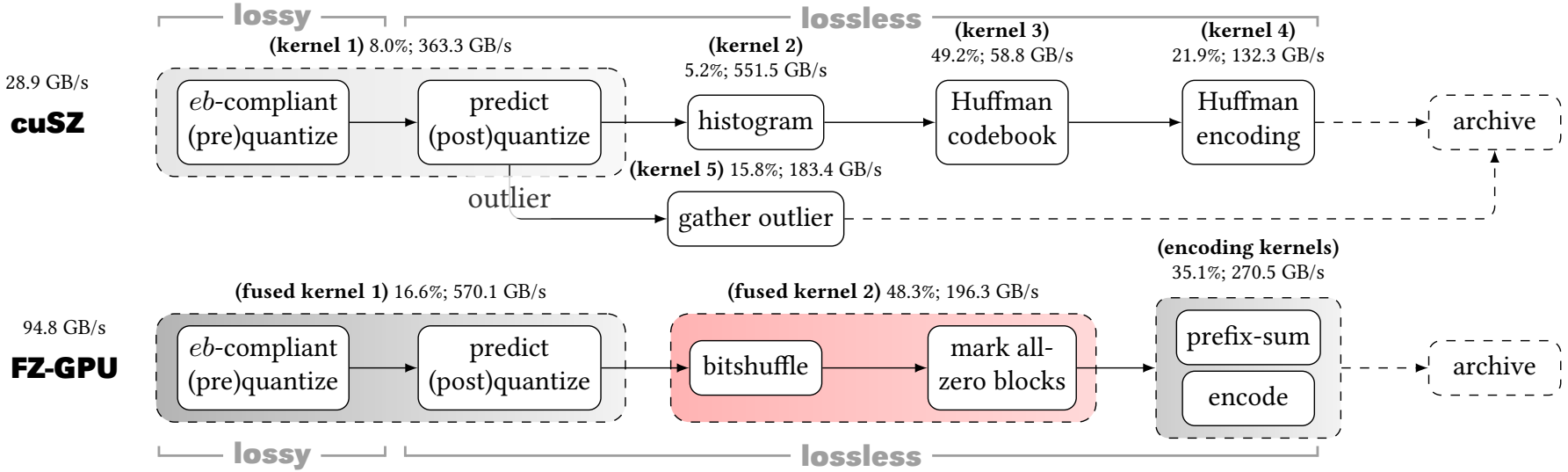
Low throughput rooted in Huffman encoding

- Seen In cuSZ and MGARD-GPU.
- Huffman encoding being a long pipeline: multiple kernels, resulting in low throughputs.
- Challenging to parallelize in a fine-grained manner

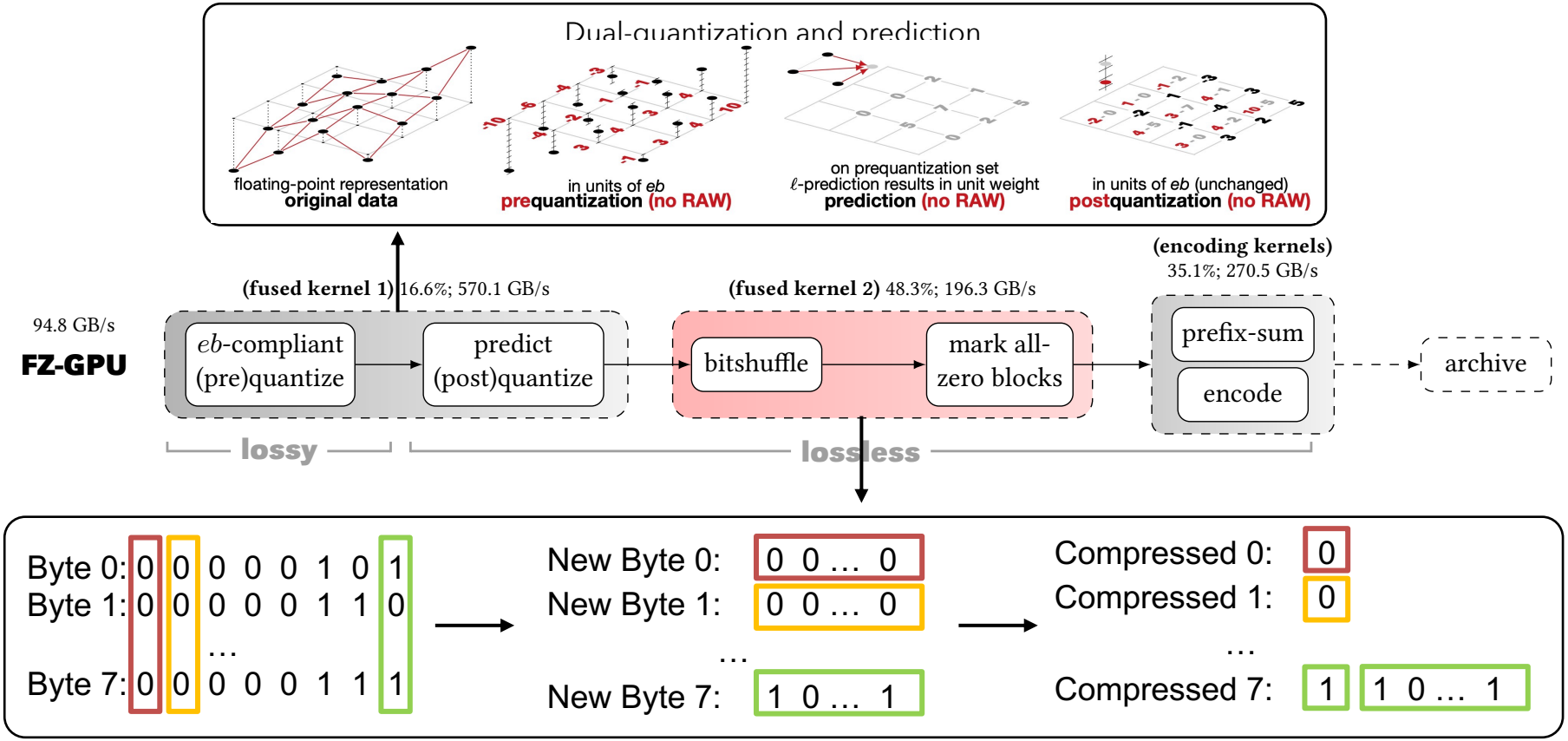


Our Solution: FZ-GPU

1. Inspired by cuSZ, we apply the **dual-quantization** in the first stage
2. Instead of using Huffman encoding as cuSZ, we utilize **bitshuffle**
3. We propose **a fast lossless encoding** kernel specifically for our pipeline
4. To save the data movement time between CPU and GPU, we use **kernel fusions**



Design Details

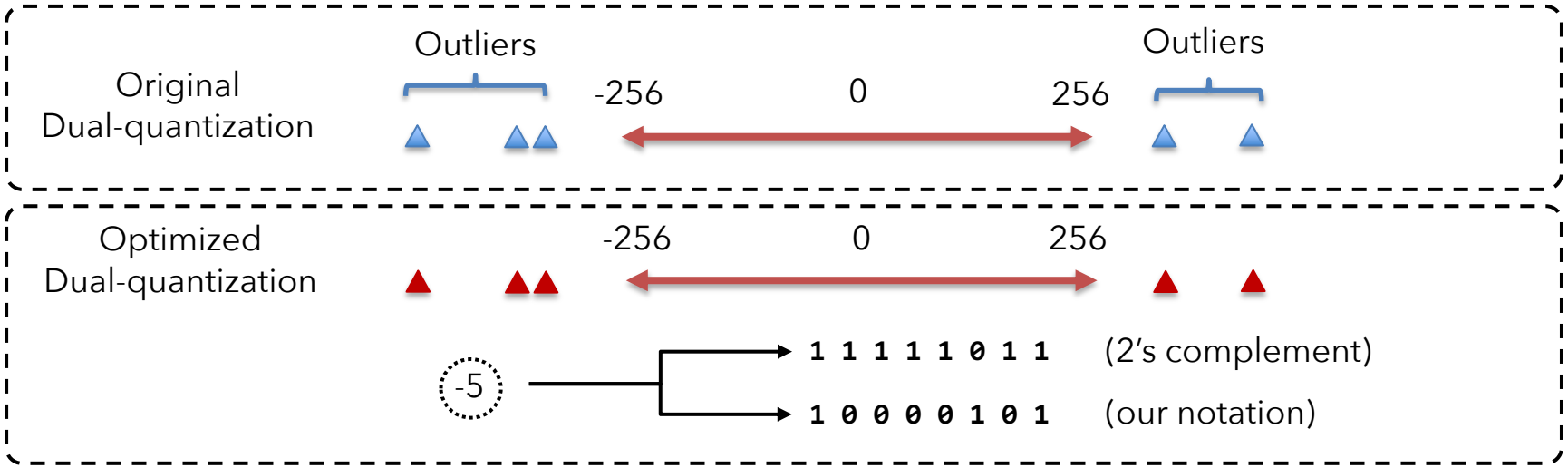


Algorithm Level Optimizations

1. **Optimizing dual-quantization**
2. **Optimizing bitshuffle on GPUs**
3. **Fast lossless encoders and kernel fusions**

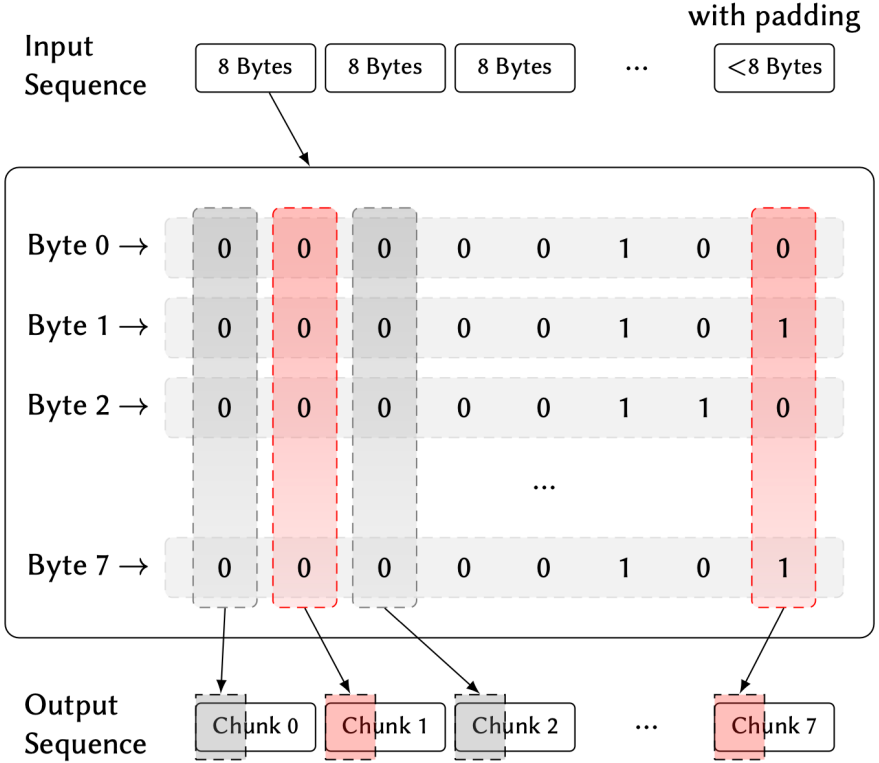
Optimizing Dual-Quantization

1. **Avoid separately handling outliers** for high performance
   - One less branched data path
2. **Use 1 bit to denote the sign** of each quantization code instead of using 2's complement
   - Much fewer set bits (1's)



Optimizing Bitshuffle on GPUs

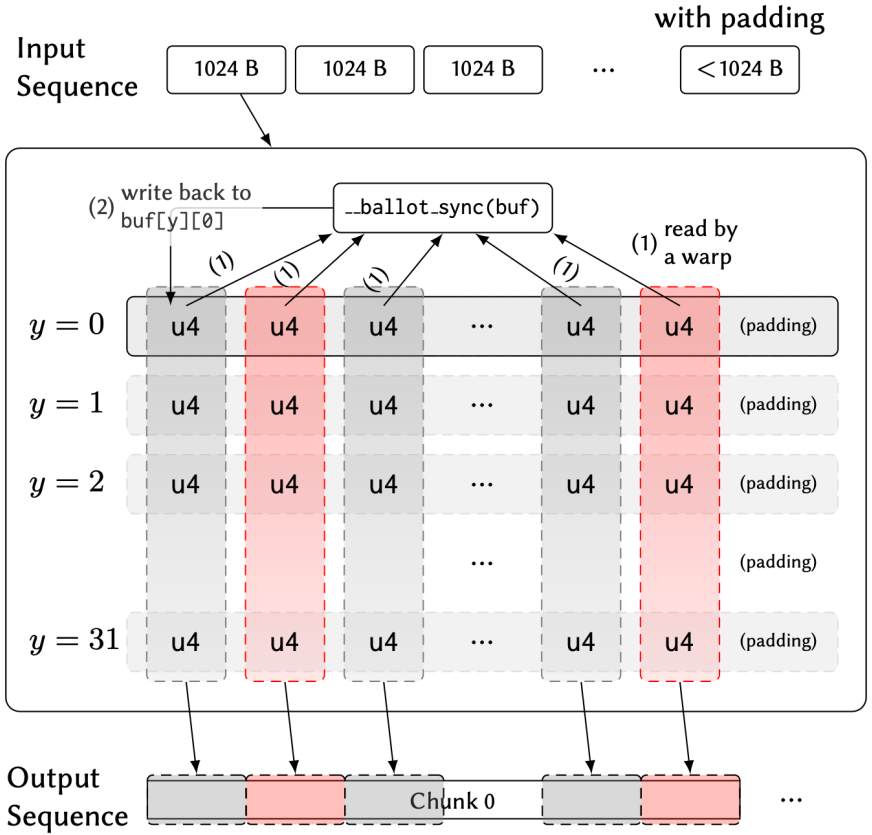
- Use a warp-level vote function to shuffle bits to **resolve data access conflicts**
- Store the result locally to **enable coalesced memory access**
- **Fully leverage** shared memory in each thread block



A simplistic fine-grained parallel bitshuffle

Optimizing Bitshuffle on GPUs

- Use a warp-level vote function to shuffle bits to **resolve data access conflicts**
- Store the result locally to **enable coalesced memory access**
- **Fully leverage** shared memory in each thread block



Our **proposed** scalable GPU bitshuffle method. **u4** stands for 4-byte unsigned integer type.

Fast GPU Lossless Encoder & Kernel Fusion

- **Partition data** into chunks and iterate all data blocks
- Record **whether all values in one block are zeros** (use 1 bit to denote) and copy data if not all zeros
- **Fuse bitshuffle kernel** and the first phase of our encoding to save one time of global memory access

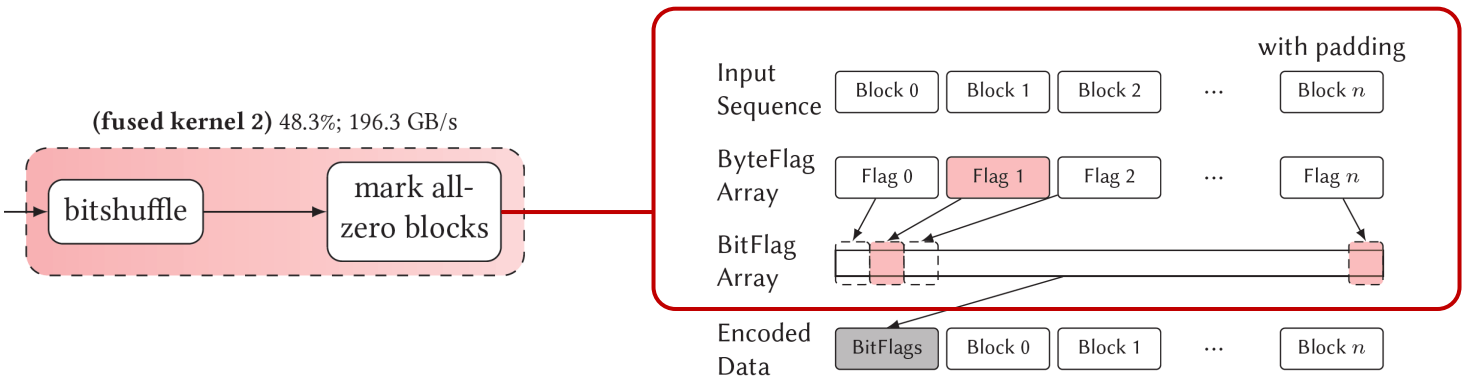


Figure 6: Our proposed fast GPU encoding method.

Evaluation

INDIANA UNIVERSITY BLOOMINGTON

Argonne NATIONAL LABORATORY

IU BigRed 200 HPC Cluster node

- 2x 64-core AMD EPYC 7742 CPUs at 2.25GHz .
- 4 NVIDIA Ampere A100 GPUs (108 SMs, 40GB), CUDA 11.4.120.

Workstation

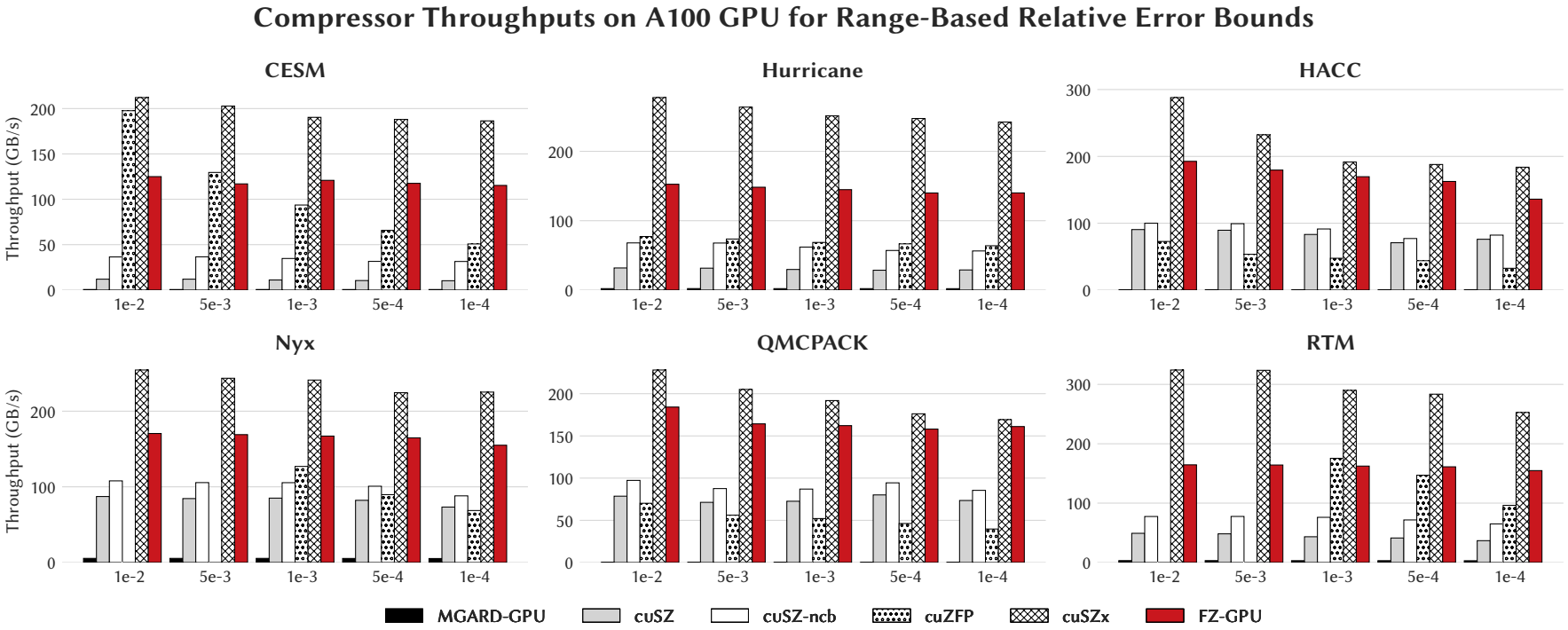
- 2x 28-core Intel Xeon Gold 6238R CPUs at 2.20GHz.
- 2x NVIDIA GTX A4000 GPUs (40 SMs, 16 GB), CUDA 11.7.99.

Metrics

- Compression ratio
- Compression throughput
- Overall throughput/data transfer rate
- Quality of reconstructed data

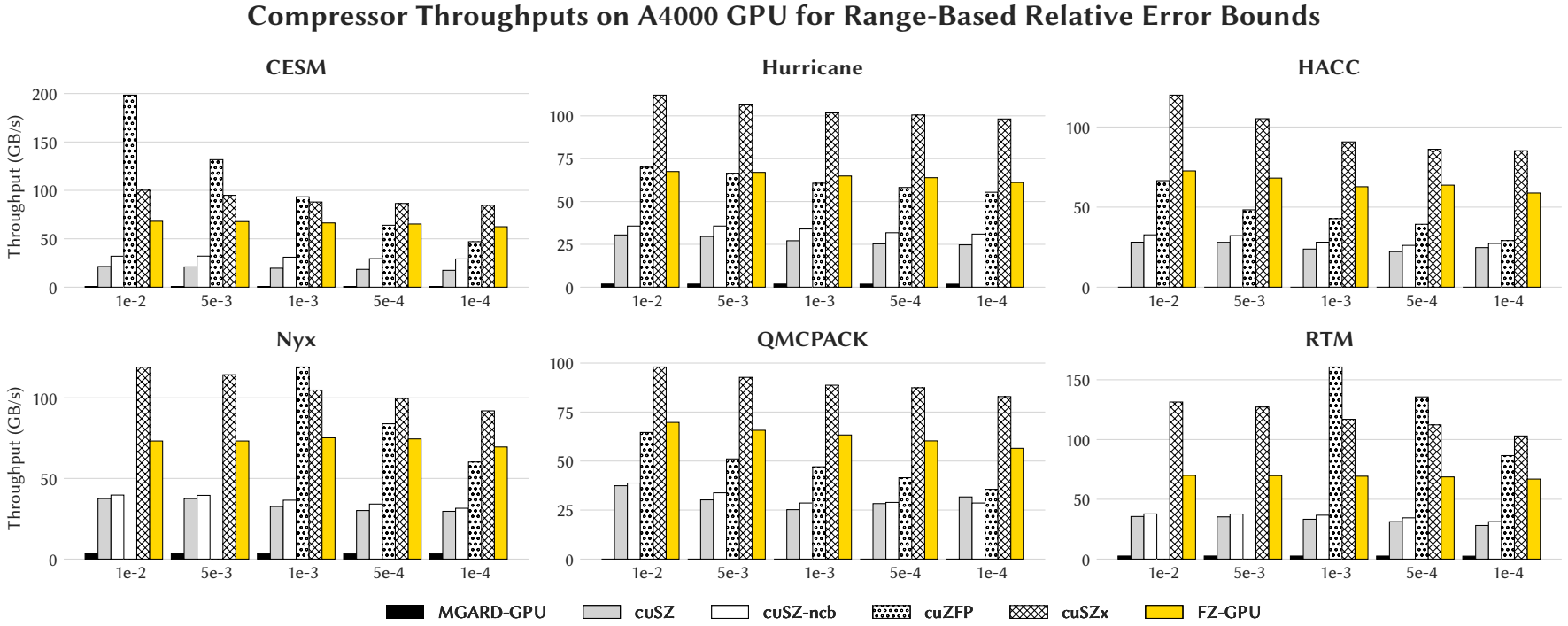
datasets	FIELD DATA SIZE dimensions	#FIELDS examples(s)
COSMOLOGY HACC	1,123.81 MB 280,953,867	6 in total xx, vx
CLIMATE CESM	25.92 MB 1,800×3,600	70 in total CLDICE, RELHUM
COSMOLOGY NYX	536.87 MB 512×512×512	6 in total baryon_density
CLIMATE HURRICANE	100 MB 100×500×500	13 in total CLDICE, QRAIN
QUANTUM CIRCUITS QMCPACK	630.74 MB 7,935×69×288	1 in total einspline
PETROLEUM EXPLORATION RTM	189.50 MB 449×449×235	16 in total snapshot_1200

Evaluation: Compression Throughput



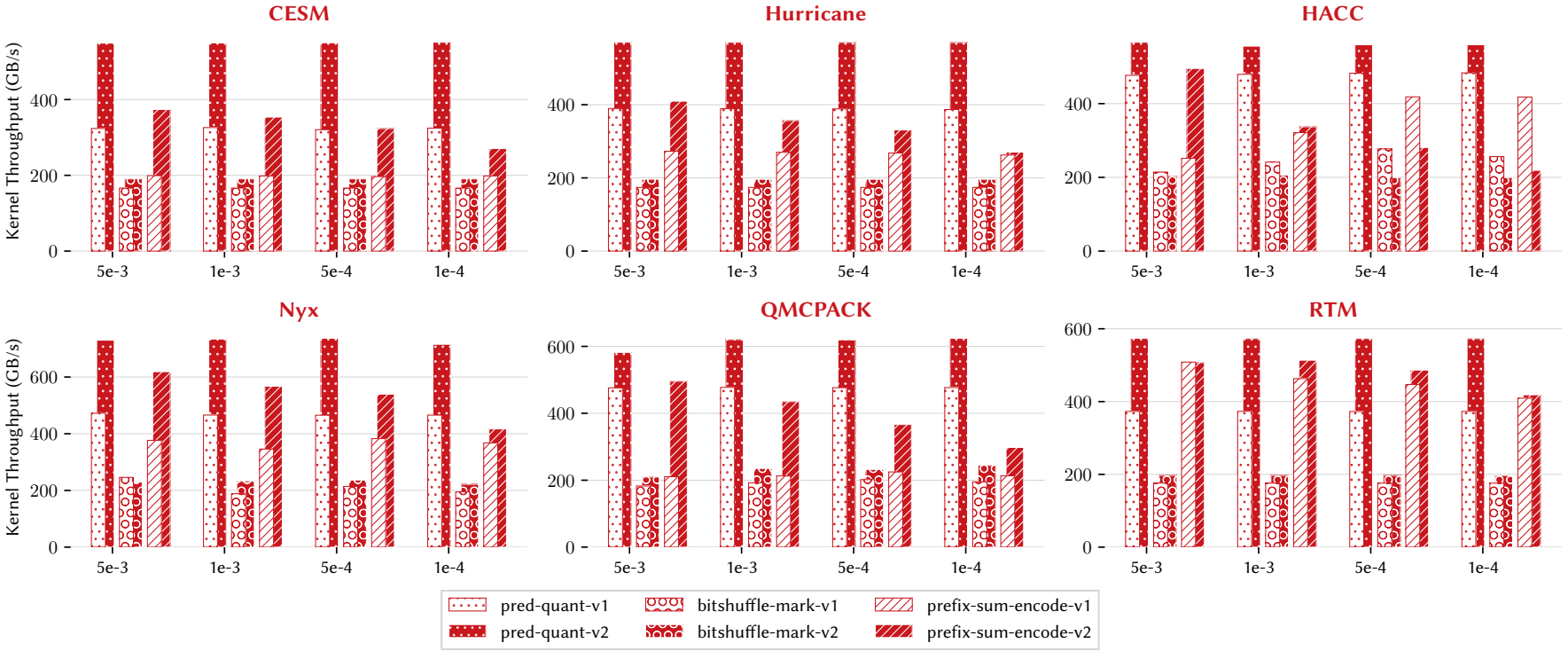
FZ-GPU achieves a speedup of up to **11.2x** over **cuSZ**, and a speedup of up to **4.2x** over **cuZFP** on A100

Evaluation: Compression Throughput



FZ-GPU achieves a speedup of up to **2.8x** over **cuSZ**, and a speedup of up to **2.1x** over **cuZFP** on A4000

Evaluation: Optimizing Kernels



Dual-quant
1.7x speedup

Kernel fusion
1.1x speedup

Prefix-sum
1.9x speedup

Evaluation: Overall Throughput

FZ-GPU achieves the **best** overall GPU-CPU throughput on almost all datasets and evaluated relative error bounds

$$T_{overall} = ((BW \times CR)^{-1} + T_{comp}^{-1})^{-1}$$

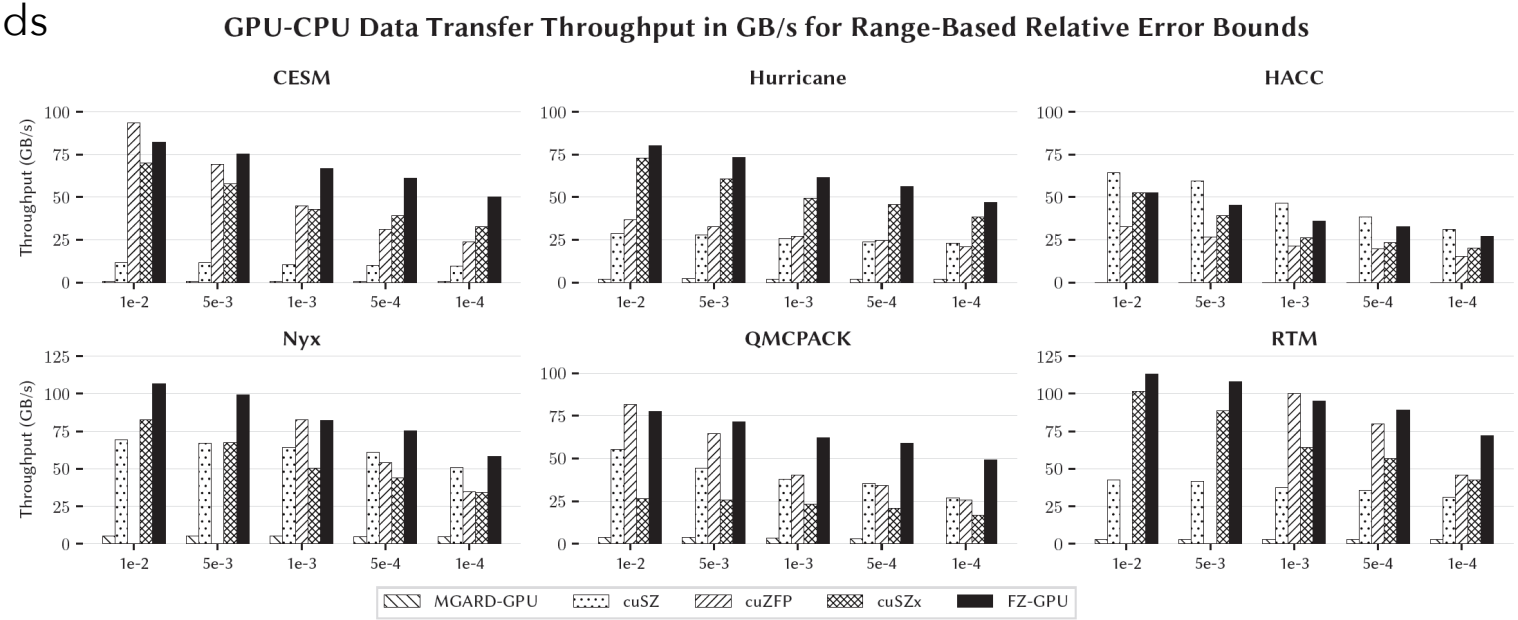
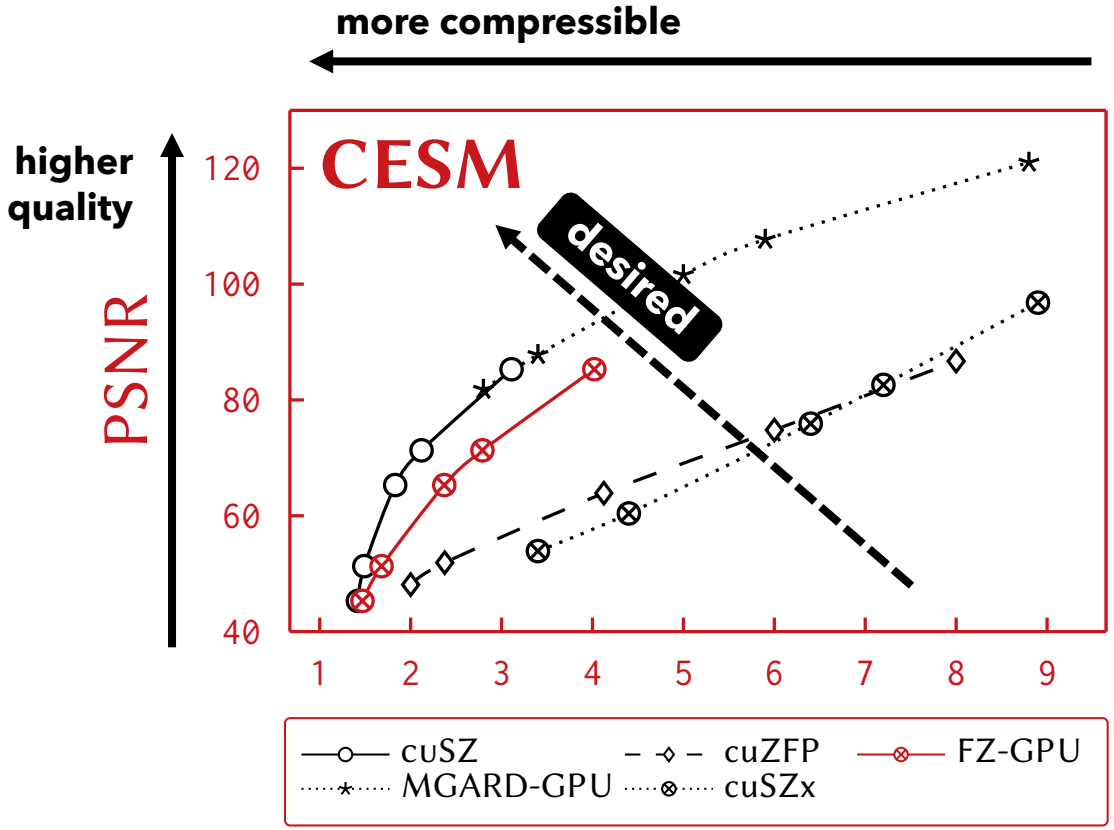


Figure 11: Overall CPU-GPU data-transfer throughput of cuZFP, cuSZ, cuSZx, MGARD-GPU, and FZ-GPU on NVIDIA A100.

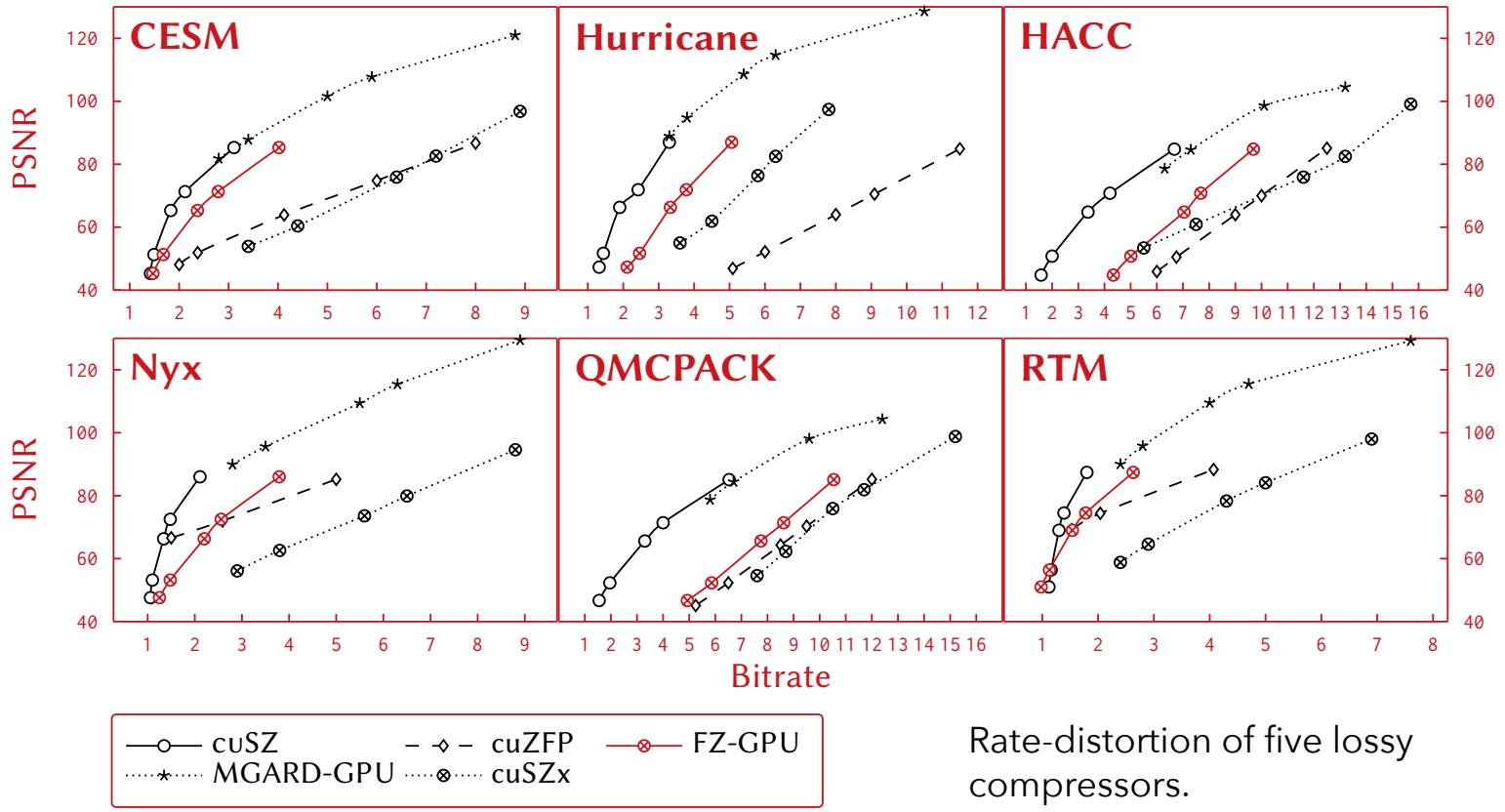
Evaluation: Compression Ratio & Quality



Rate-distortion

- Defined as $\frac{\texttt{bitsof(Type)}}{\texttt{comp. ratio}}$
- **Integrate** evaluations of compression ratio and data quality.
- If fixed-rate mode, the quality is **linear** to the bitrate (ZFP)
- Our compressor is **superlinear** at the highly compressible end.

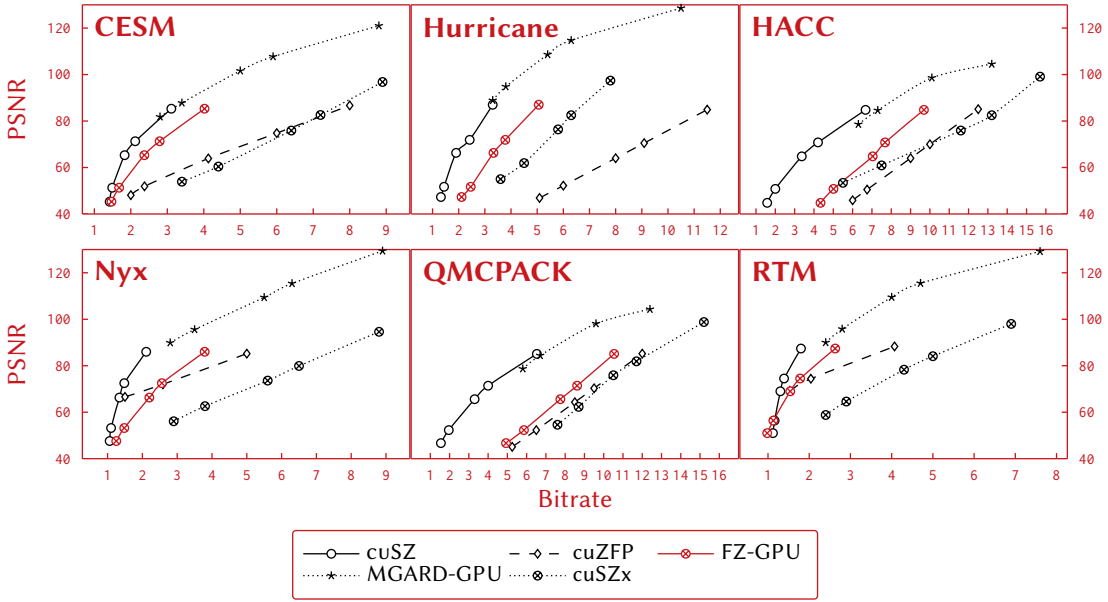
Evaluation: Compression Ratio & Quality



Rate-distortion of five lossy compressors.

Evaluation: Compression Ratio & Quality

- FZ-GPU is up to **1.1x** higher than cuSZ and **1.7x** higher than cuZFP on average.
- FZ-GPU has an average compression ratio improvement of **2.4x** and **4.3x** higher compression ratio at most than cuSZx.
- MGARD-GPU, similar curve in RTM dataset. But much lower compression throughput.



Rate-distortion of five lossy compressors.

Evaluation on Distortion

INDIANA UNIVERSITY BLOOMINGTON

Argonne NATIONAL LABORATORY

SSIM:

FZ-GPU has the **highest** SSIM among all compressors

PSNR advantage of FZ-GPU

- VS cuZFP/cuSZx: **1.3X/1.1X** higher
- VS MGARD-GPU: multi-grid-based MGARD-GPU has **slightly** better quality at a **high** cost of **13.34x** FZ-GPU kernel time

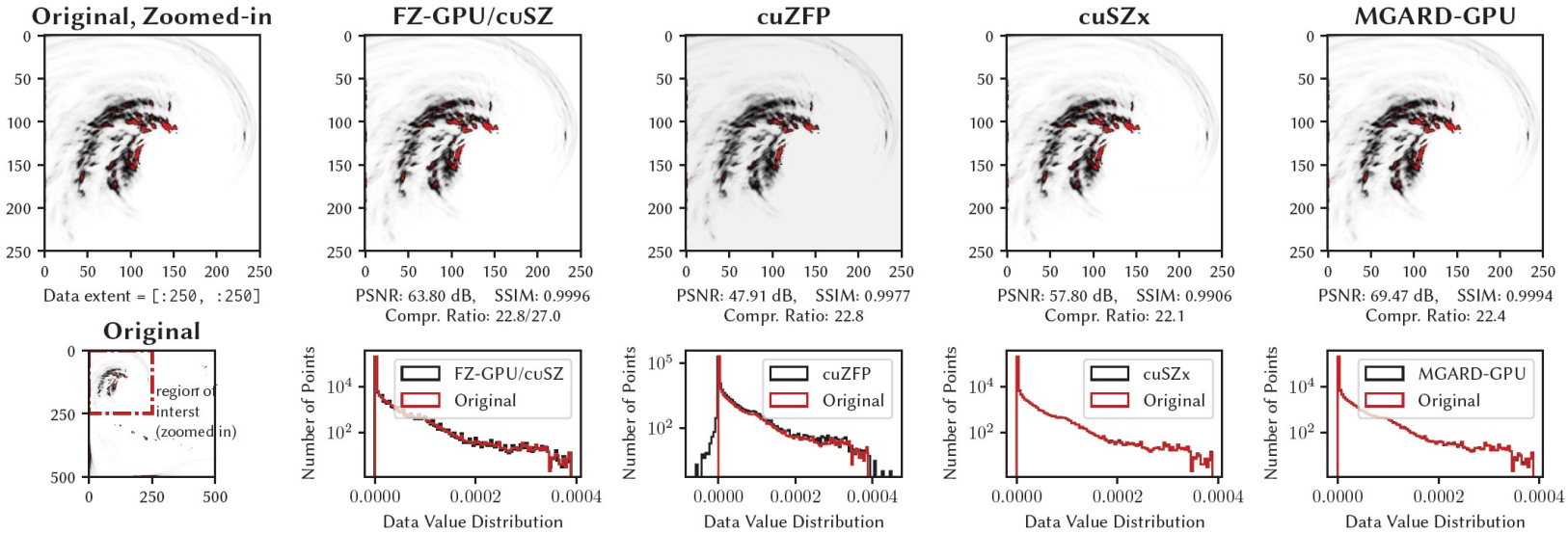


Figure 12: Reconstructed data quality using various GPU-based lossy compressors on field `QSNOWf48` (slice 50) in the Hurricane dataset, under a similar compression ratio. The first row shows the visualization of the region of interest, while the second row shows the data distribution comparison between the decompressed and the original data for each compressor.

Conclusion & Future Work

In this paper, we design a new compression pipeline that consists of

- dual-quantization
- bit-shuffle
- fast lossless encoding.

We also propose a series of architectural optimizations, including

- warp-level optimization for bitwise operations,
- maximization of shared memory utilization,
- and multi-kernel fusion.

In the future, we plan to

1. exploit fusing all GPU kernels into one to improve the performance further,
2. adapt FZ-GPU to other GPU platforms by using code translation tools such as HIPFY for AMD GPUs and SYCLomatic for Intel GPUs
3. Evaluate FZ-GPU with real-world applications requiring fast compression, such as memory compression.

Acknowledgment

This R&D was supported by the Exascale Computing Project (ECP), Project Number: 17-SC-20-SC, a collaborative effort of two DOE organizations—the Office of Science and the National Nuclear Security Administration, responsible for the planning and preparation of a capable exascale ecosystem. This repository was based upon work supported by the U.S. Department of Energy, Office of Science, under contract DE-AC02-06CH11357, and also supported by the National Science Foundation under Grants SHF-1617488, SHF-1619253, OAC-2003709, OAC-1948447/2034169, and OAC-2003624.

Thank you.

Questions?

github.com/szcompressor/FZ-GPU

contact us

Boyuan Zhang
bozhan@iu.edu

Dr. Dingwen Tao
ditao@iu.edu