Productive, Performant, and Parallel Generic Lossy Data Compression with LibPressio

Victoriana Malvoso
Jon Calhoun (Advisor)
Robert Underwood

Abstract

- Ensuring productivity and good performance in HPC is important to gather results quickly and accurately.
- Data compression reduces the size of data in order to reduce the amount of memory required to store and process it. Many supercomputers simply do not have the storage to contain the data of large-scale datasets [1].
- Determining the best compressor to use to compress data can be time-consuming. Not all compression techniques are appropriate for all data sets
- Our work demonstrates that the library LibPressio combines the ability to use a single interface for compressors without compromising performance.

What is LibPressio?

Switching between compressors can be time-consuming and requires trial and error. LibPressio is a library that allows scientists to easily switch between compressors to determine which is best for their data. It also contains a generic parallel implementation for thread-safe compressors to increase speed of compression.

The Why

Is LibPressio's compression time and ratio meaningfully different from the base compressors? Is the performance difference worth the ease of use of LibPressio?

Methodology

- Implement LibPressio's thread-safe version of SZ.
- LibPressio's performance is evaluated based on time overhead serially and scalability and compression ratio in parallel.

Conclusions

- LibPressio decreases code duplication and engineering effort.
- LibPressio's performance matches compressor performance.
- LibPressio's generic thread-safe implementation improves productivity and runtime.

Results

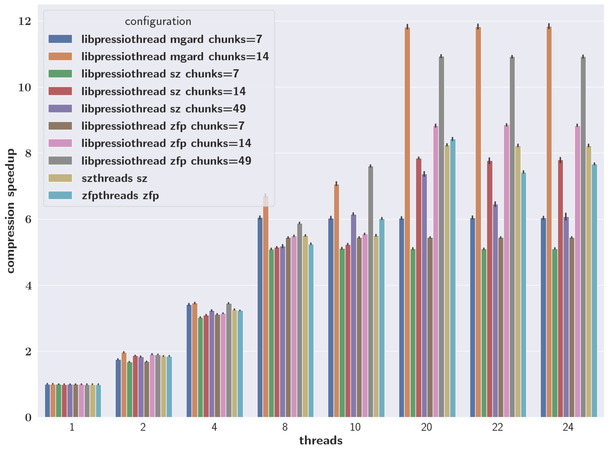
Task	Compressors	Lines Native	Lines LibPressio	Improvement	Relative Improvement
ADIOS2 [20]	3	744	367	377	50.67%
BindingJulia [22]	1	299	25	274	91.64%
BindingPython [9], [23] †	2	768	363	405	52.73%
BindingR	-	-	793	-	-
BindingRust [24]	1	112	34	78	69.64%
CLI [9], [16], [17] †	3	1649	756	893	54.15%
Configuration Optimizer [25]	1	4683	1869	2814	60.09%
DistributedExperiment	-	-	613	-	-
Fuzzer	-	-	24	-	-
HDF5 filter [9], [16] †	2	1469	438	1031	70.18%
Z-Checker [5]	7	3052	405	2647	86.73%

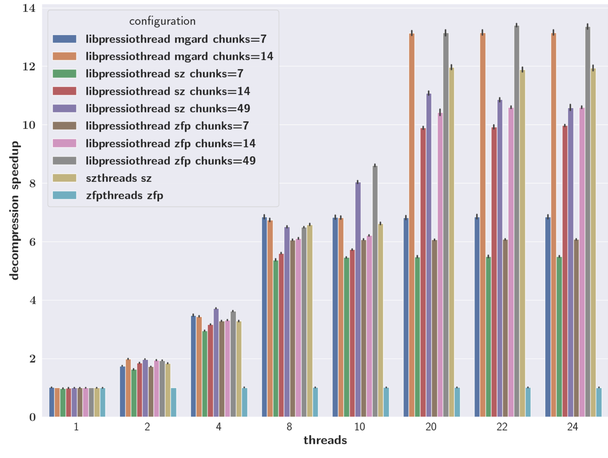
This data shows that LibPressio decreases engineering effort by reducing the lines of code required for the implementation of different tasks.

library	lossless compression	lossy compression	n-d data aware	datatype-aware	embeddable design	arbitrary configuration	option introspection	third party extensions
ffmpeg [10]	✓	✓	❑	✓	✓	✓	✓	✓
Foresight/CBench [2]	✓	✓	✓	✗	❑	✓	✗	✗
HDF5 [11]	✓	✓	✓	✓	✓	✗	✓	✓
imagemagick [12]	✓	✓	❑	✓	✓	✓	✗	✓
libarchive [13]	✓	✗	✗	✗	✓	✗	✓	✗
NumCodecs [14]	✓	✓	✓	✓	✗	✗	✓	✓
SCIL [3]	✓	✓	✓	✓	✓	✗	✓	✗
Z-checker (0.7) [5]	✓	✓	❑	✓	❑	✗	✗	✓
LibPressio	✓	✓	✓	✓	✓	✓	✓	✓

This data shows that LibPressio is inclusive of many different metrics as compared to other compression libraries.

Time Speedup for Parallel Implementation Compression and Decompression





Compression Ratios for Parallel Implementation

config	chunksize	size:compression_ratio
libpressiothread sz	7	2.245248
	14	2.197810
	49	1.887099
libpressiothread zfp	7	1.427047
	14	1.377799
	49	0.830169
szthreads sz	1	2.256079
zfpthreads zfp	1	1.589013

- As number of threads increases, LibPressio's speedup is larger than ZFP, and smaller than SZ with the highest chunk size.
- LibPressio with SZ begins to decline in speedup at 22 threads.
- ZFP's compression ratio is 13% higher than LibPressio with ZFP at chunk size 49.
- SZ's compression ratio is 0.5% greater than LibPressio with SZ at chunk size 7.

Details about the computing environment used to gather results can be found in the abstract.

What is Thread-safe?

- Threading is the ability to split one process into multiple processes that are run simultaneously in order to increase speed of performance.
- Thread-safe processes can split up tasks across multiple threads without causing any errors.
- Many supercomputers today can support multi-threading processes, but not all compressors take advantage of it.

Related Works

- Other abstractions interfaces such as Foresight and SCIL are similar, however:
  - LibPressio respects dimensionality, which gives more accuracy in reporting of compression ratios [2].
  - LibPressio contains a generic parallel implementation [3].

Acknowledgements

Clemson University is acknowledged for generous allotment of compute time on the Palmetto cluster. This material is based upon work supported by the National Science Foundation under Grant No. SHF-1910197.

References

[1]Franck Cappello, Sheng Di, Sihuan Li, Xin Liang, Ali Murat Gok, Dingwen Tao, Chun Hong Yoon, Xin-Chuan Wu, Yuri Alexeev, and Frederic TChong. 2019. Use cases of lossy compression for floating-point data in scientific data sets. The International Journal of High Performance ComputingApplications33, 6 (2019), 1201–1220.
[2]Pascal Grosset, Christopher M Biwer, Jesus Pulido, Arvind T Mohan, Ayan Biswas, John Patchett, Terece L Turton, David H Rogers, Daniel Livescu, and James Ahrens. 2020. Foresight: analysis that matters for data reduction. InSC20: International Conference for High Performance Computing, Networking, Storage and Analysis. IEEE, 1–15.
[3]Julian Martin Kunkel, Anastasiia Novikova, and Eugen Betke. 2017. Towards decoupling the selection of compression algorithms from quality constraints–an investigation of lossy compression efficiency. Supercomputing Frontiers and Innovations4, 4 (2017), 17–33.

Get LibPressio

https://github.com/robertu94/libpressio

Connect with me on LinkedIn!

https://www.linkedin.com/in/victoriana-malvoso-7615471aa/