Zaoui, Fabrice

A Lossy Compression Experiment of Telemac Data

A Lossy Compression Experiment of Telemac Data

Fabrice Zaoui
EDF R&D – National Laboratory for Hydraulics and Environment (LNHE)
6 quai Wattier, 78401 Chatou, France
E-mail: fabrice.zaoui@edf.fr

***Abstract*— The Telemac-Mascaret system can require or produce large amount of data to address complex problems in geoscience or industry. This statement is verified with the availability of new databases and the increase of problem sizes solved by numerical simulation nowadays. But how to efficiently deal with such data when file sizes can range from several gigabytes to terabytes, or more? Dealing with a vast volume of data is not always possible. Copy, transfer or storage of large files can quickly become an issue to carry out a study in a short period of time. The high memory usage is mainly due to the double precision floating point representation of billions of real numbers, which can hardly be avoided except with doing a compromise between necessity and feasibility. For example, the latter can be the frequency of saving results in file which may differ from the calculation time step. In this paper, another compromise is presented. It is the possibility of reducing the insignificant part of the information by compressing the data with a lossy algorithm. The acceptance of a loss of accuracy may have meaning in view of the uncertainty present in data especially if it is linked to the definition of error values that cannot be exceeded. To this end, the lossy compressor SZ is tested on hydrodynamic data coming from very large Selafin files. This tool can highly decrease the size of the data with compression ratio values directly depending on the user's choice for the error bounds. Some limitations and perspectives of this experiment are also discussed.**

I. INTRODUCTION

Many sources of uncertainty lie in real-world problems. The underlying data needed to describe and analyse a case study always has a high level of numerical accuracy when stored on disk leading to large file sizes. However, this level of accuracy does not necessarily reflect the reliability of the information whether it comes from observations or calculation results. Telemac simulations can require or produce large amount of disk data, in the range of several gigabytes to terabytes, which can cause copy, transfer or storage issues.

Such a vast volume of data is not still possible or tractable. Even if the floating-point representation of real numbers in 64 bit double-precision format helps keep rounding errors within an acceptable range for the convergence of many numerical algorithms, it is not always necessary or relevant when storing a database with millions of samples for example. The well-known lossless compressors like gzip or 7z are possible solutions to decrease file sizes but with a low compression ratio for binary data. With the help of HPC, this paper proposes to investigate better compression rates by accepting certain but controlled losses on hydraulic data.

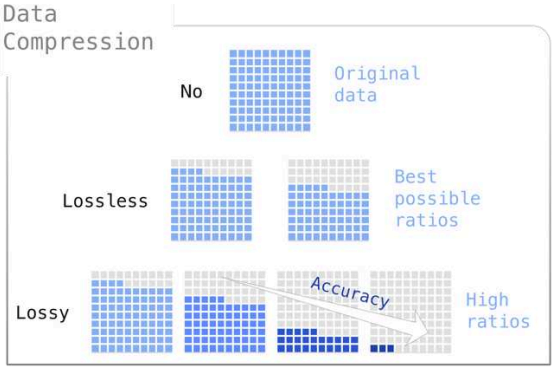

Figure 1. Lossless and lossy data compression

Accuracy is one of the criteria to evaluate or measure data quality, see Fig. 2. It tells how far the data is from reality, considering this last one is known with certainty. Other criteria exist like: *Completeness* (is the data diversity sufficient to fulfill the requirements of phenomena studies?); *Consistency* (is there any kind of contradictory information in the database?); *Timeliness* (how old is the data and does it still reflect a possible new reality?); *Validity* (is the available data in the correct structure and format?); *Uniqueness* (does a recording with specific characteristics only appear once?); *Integrity* (can the relevant information always be found in the database whatever the request?); and *Auditability* (are data changes traceable?).

Figure 2. Data quality criteria

Accuracy should not be confused with precision. Accuracy is important to depict reality well but is not directly related to precision where the concepts of repeatability and reproducibility dominate. Accuracy refers to how close data is to a known value while precision refers to data dispersion. Consequently, accuracy and precision are not correlated, see Fig. 3.

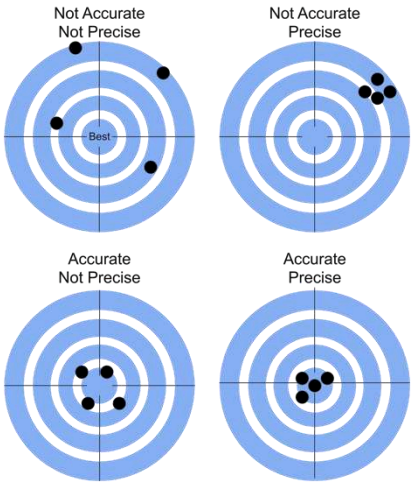

Figure 3. Accuracy vs Precision

A higher numerical accuracy in data does not necessarily reflect, paradoxically, a better reality. This is due to the uncertainty that occurs at all stages of data acquisition and processing. Indeed, data acquisition is prone to errors because it can be intrusive and then change reality, or it uses measurement or communication devices with limited specifications or reduced performance over time (perhaps in addition of one or more human operations) or it can be simply partial. The simulation of physics is also error prone. Nowadays it mainly relies on numerical modeling with incomplete or approximative consideration of complex phenomena. Moreover, all the real numbers can not exactly be represented by computers and rounding operations are performed. This is due to the binary representation in memory with a limited number of bit. With this drawback in mind, the best possible accuracy in the floating-point representation of real numbers still remains primordial for the numerical computations as it mainly governs the convergence of solving algorithms and confidence in output results.

Thus, the level of accuracy for data storage must be fixed according to the knowledge of the errors involved in the processing chain (uncertainty, see Fig. 4).

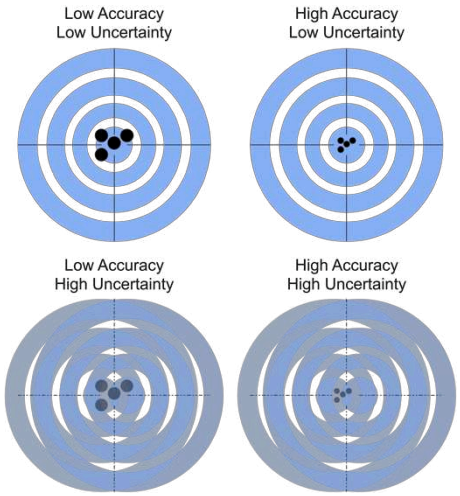

Figure 4. Accuracy vs Uncertainty

II. Data Compression

When considering the study of a geoscience problem with the help of numerical modeling, one will sooner or later face the huge size of data in the input sets or output results. Even with a supercomputing infrastructure, the data file sizes without any compression can be a restriction on what it is really possible to study, save and share because of the limited amount of memory and network bandwidth.

Data compression for reducing the logical file size is usually done without any loss of information but with a limited power (ratio between the uncompressed and compressed sizes). Lossy compressors (SZ, ZFP, ISABELA, NUMARCK…) have better ratio performances but with a loss of accuracy when compressing information.

SZ is an open framework designed for scientific data and has the advantage of providing several criteria to control loss of accuracy error [1]. It can be used for many purposes involving data processing and has implementations on CPU, GPU and FPGA. It supports different languages like C, Fortran, Java and Python. It is also included in I/O libraries like HDF5.

The version of June 2020 (v2.1.8.3) has been installed on EDF GAIA HPC cluster with the default options (no OpenMP, Intel Xeon Gold 6140 2.3 GHz, 384 GB per node). It is used on some Mascaret and Telemac hydraulic results but not directly on Selafin files as it is designed for the (de)compression of floating-point arrays in byte stream format. Thus, before using SZ, the Selafin files are first converted into raw binary two-dimensional files: a number of mesh nodes and a number of time steps. If this file is named 'telemac.dat' with results on a mesh of 10,000 nodes for 2,000 time steps, its compression based on SZ is simple:

```
> sz -z -d -c sz.config -i telemac.dat -2 10000 2000
```

where:

-z is for compression;

-d for double precision;

-c for the configuration file (user criteria on the errors produced);

-i for the input data file;

-1, **-2**, **-3**, **-4** for the array sizes.

The result of this command will be the creation of a new file 'telemac.dat.sz' whose size will depend on the configuration file 'sz.config'. For the decompression, the command is similar:

```
> sz -x -d -s telemac.dat.sz -2 10000 2000
```

where:

-x is for decompression;

-s for the input file.

The result of this last command will be the creation of the file 'telemac.dat.sz.out' whose size is the same as 'telemac.dat' but with a loss of accuracy.

More options and combinations are possible, see the in-line help information (> ***sz -h***) if needed.

Eight options are available in the configuration file to control different types of error bounds. Only three of them are presented and tested int this work:

- The absolute error bound (ABS) is to limit the errors to be within an absolute error. For instance, if this value is 10^{-3} then all the (de)compressed values will be in $[V - 0.001, V + 0.001]$ where V are original values;
- The relative bound ratio (REL) is to limit the errors by considering the global data value range size. For instance, if this value is 10^{-3} and the dataset is $\{0, 1, 2, 3, \dots, 100\}$ then the error bound will be $0.1 = (100 - 0) \times 10^{-3}$;
- And the point-wise relative bound ratio (PW_REL) is to limit the errors by considering each value. For instance, if this value is 10^{-3} and the dataset is $\{0,1,2,3, \dots ,100\}$ then (de)compression errors will be limited to $\{0, 0.001, 0.002, 0.003, \dots, 0.1\}$.

III. CASE STUDY

A. Mascaret

The case is a water quality study on the Rhine river where the discharge is saved on file at every time step. This allows to compute the hydraulics only once for the tracer part (convection and diffusion of constituents) and thus speeding-up the overall simulation as the tracer part is generally less computational time-consuming compared to the hydraulic part of Mascaret.

The resulting file is named 'Q.masc' whose description is given in Table 1.

TABLE 1: MASCARET ORIGINAL FILE DESCRIPTION

Number of nodes	1,170
Number of time steps	1,848,961
Number of values (8 Bytes)	2,163,284,370
Original file size (GB)	16
Original file size (Bytes)	17,306,274,960
Min. value ($m^3.s^{-1}$)	15.3
Max. ($m^3.s^{-1}$)	866.51
Mean ($m^3.s^{-1}$)	114.18
Median ($m^3.s^{-1}$)	77.11

The compression of the file 'Q.masc' with SZ is:

> sz -z -d -c sz.config -i Q.masc -2 1170 1848961

This file of 17 GB has been zipped with two lossless compressors (gzip and 7z with default options) to get some comparison items shown in Table 2.

TABLE 2: PERFORMANCES OF LOSSLESS COMPRESSORS

Tool	Output file size	Ratio	Compression time	Decompression time
gzip	14 GB	1.19	20'28"	3'58"
7z	11 GB	1.54	12'05"	17'23"

B. Telemac

For testing the compression of Telemac results, the 2D example case of Malpasset is considered, see Fig. 5 (fine grid mesh version with 104,000 triangular elements). The water depth is saved on the file 'H.tel' at each time step.

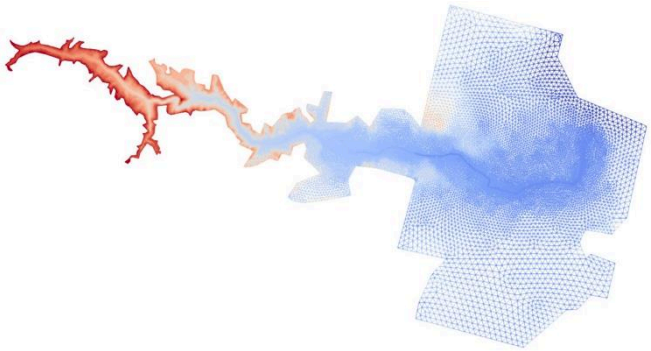

Figure 5. Malpasset triangular mesh

The compression of the file 'H.tel' with SZ is:

> sz -z -d -c sz.config -i H.tel -2 53081 400000

TABLE 3: TELEMAC ORIGINAL FILE DESCRIPTION

Number of nodes	53,081
Number of time steps	400,000
Number of values (8 Bytes)	21,232,400,000
Original file size (GB)	158
Original file size (Bytes)	169,859,200,000
Min. value (m)	0.
Max. (m)	193.92
Mean (m)	4.46

The data file 'H.tel' of 158 GB is too large to conduct all the tests of compression for the different criteria (ABS, REL and PW_REL) due to a lack of RAM on the computational node. It is also the reason why the median value of the water depths is not in Table 3.

IV. RESULTS

A. Mascaret

The first exercise is to test the compression of the file 'Q.masc' with the ABS criterion. Eight error threshold values have been set in the configuration file and all corresponding compression results are presented in Table 4.

TABLE 4: MASCARET ABS TESTS

ABS	Size	Compression time	Decompression time	Ratio	Check
1	5.7 MB	1'14	31"	2932	YES
0.1	33 MB	1'15	31"	509	YES
0.01	125 MB	1'23	34"	133	YES
0.001	215 MB	1'26	37"	77	YES
10^{-6}	533 MB	1'29	42"	31	YES
10^{-9}	2.8 GB	1'53	2'14"	6	YES
10^{-12}	8.3 GB	2'42	2'23"	1.94	NO
10^{-15}	12 GB	2'46	1'24"	1.45	NO

A systematic check of the ABS error on the decompressed data was performed. For the seventh and eighth test, the error threshold check did not work. Indeed, the value of 1.023×10^{-12} for the maximal error was obtained instead of 10^{-12} as requested, and 1.78×10^{-15} instead of 10^{-15}. This difference remains nevertheless very acceptable in view of the accuracy value to be satisfied (near the machine precision for floating point numbers stored in eight bytes).

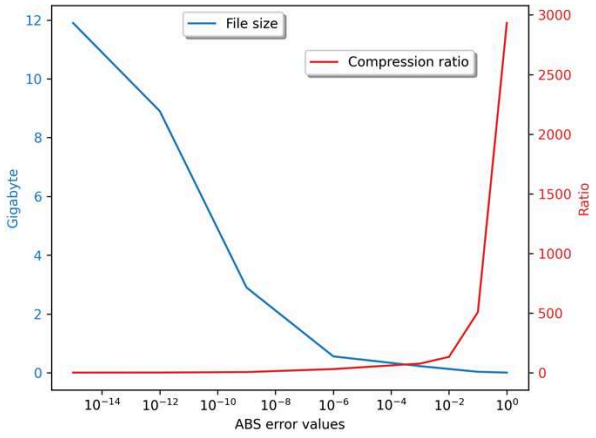

Figure 6. PCA representations of the eight tests

A Principal Component Analysis (PCA) of this dataset is shown in Fig. 7 and 8. PCA is a dimensionality reduction technique that is used here to bring out trends and strong patterns in the dataset. The dataset can be divided into three subsets according to the first two dimensions (with an explained variance of 92%), see Fig. 7. The first point is clearly an extreme value with a very high compression ratio but also with the highest error on the hydraulic discharge of $1\ m^3.s^{-1}$.

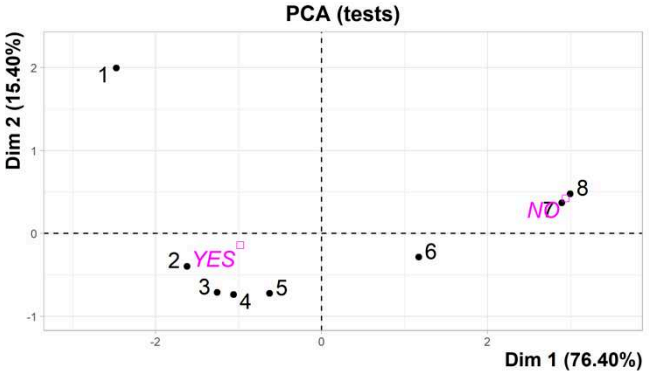

Figure 7. PCA representations of the eight tests

The dataset can be divided into three subsets according to the first two dimensions (with an explained variance of 92%), see Fig. 7. The first point is an extreme value with a very high compression ratio but for errors on hydraulic discharge of $1\ m^3.s^{-1}$. Tests 2 to 5 are more comparable in terms of processing time and gain on file sizes. The last three tests 6 to 8 are more demanding on the loss of information and present the lowest ratios.

Processing times and sizes are clearly anticorrelated to errors, with a correlation coefficient (Pearson) of -0.93 and a strong statistical significance (p-value coefficient of 7.8×10^{-4}) for the link between the file size and the ABS error criterion. The relation with the ratio is positive (ratio increases with the acceptance of a larger error) but less linear.

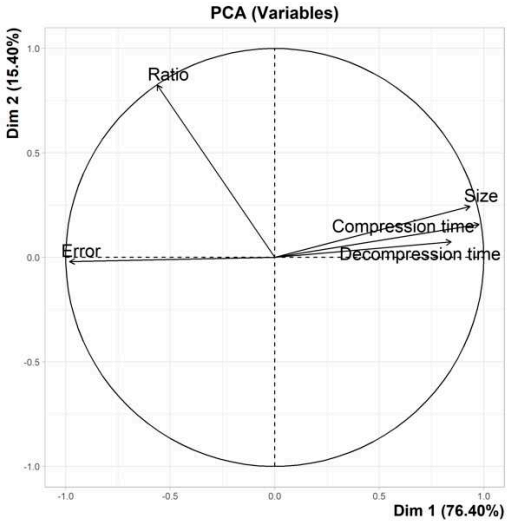

Figure 8. Correlation circle with PCA

The next test of SZ compression performance concerns the relative bound error REL. Only an error value of 10^{-3} is tested here which implies a maximum error of $0.8512\ m^3.s^{-1}$ as $10^{-3} \times (Q_{max} - Q_{min}) \approx 0.8512$. This test is successful as shown in Table 5.

TABLE 5: MASCARET REL TESTS

REL	Size	Compression time	Decompression time	Ratio	Check
0.001	6.3 MB	1'11	26	2623	YES

Lastly PW_REL criterion is tested for six error values as shown in Table 6. Compression and decompression times are not indicated in Table 6 as they are very similar for every test with approximatively $1'30''$ for the compression phase and $50''$ for the decompression one. It can be noticed that the PW_REL value of 10^{-3} can be directly compared to the REL value in Table 5 as they nearly give the same maximal error after decompression ($0.8512\ m^3.s^{-1}$ for REL and $0.84\ m^3.s^{-1}$ for PW_REL) but with a higher compression ratio for the REL algorithm.

PW_REL in Table 6 and ABS in Table 4 can be directly compared but they work differently depending on the compression ratio or the maximum error made.

TABLE 6: MASCARET PW_REL TESTS

PW_REL	Size	Ratio	Max. error ($m^3.s^{-1}$)
10^{-1}	197 KB	86184	83.03
10^{-2}	19 MB	907	8.35
10^{-3}	36 MB	471	0.84
10^{-4}	132 MB	126	0.0864
10^{-5}	256 MB	64	0.00848
10^{-6}	296 MB	56	0.000866

B. *Telemac*

The Telemac test file is about ten times larger than the Mascaret one. For this reason, not all compressions could be done due to the huge amount of RAM required by SZ leading to segmentation faults in some cases. These cases are indicated with a NA value in the following tables.

TABLE 7: TELEMAC ABS TESTS

ABS	Size	Compression time	Decompression time	Ratio	Check
1	4.7 MB	13'10"	4'54	34997	YES
0.1	31 MB	13'12"	4'52"	5299	YES
0.01	211 MB	13'57"	5'44"	770	YES
0.001	1.1 GB	14'9"	5'55"	149	YES
10^{-6}	15 GB	19'11"	14'37"	11	YES
10^{-9}	41 GB	26'58"	33'30"	4	YES
10^{-12}	NA	NA	NA	NA	NA
10^{-15}	NA	NA	NA	NA	NA

With the acceptance of 1 meter error on the water depths, the file size reduction is tremendous. Unfortunately, this maximal error value concerns a large part of the hydraulic domain, as indicated in the distribution graph in Fig. 9. This histogram presents the distribution of the error absolute value on the water depths for a randomly chosen time step. Many of the water depths have an error above $0.8\ m$.

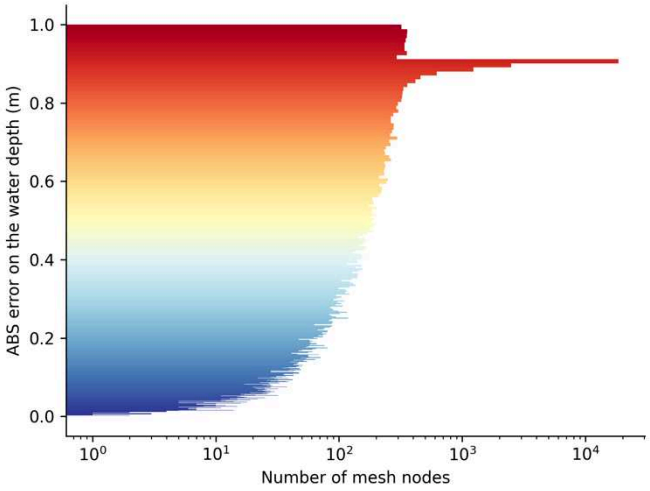

Figure 9. Absolute error distribution for ABS equal to 1 m

SZ does not produce a symmetric error centered on a zero-mean value in this case. Most of the decompressed values are greater than those in the original dataset, see Fig. 10.

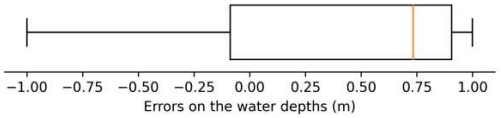

Figure 10. Non symmetric distribution of the errors

REL and PW_REL results are presented in Table 8 and 9 respectively. Once again it was not possible to go to the end of the compression test in both cases for the most restrictive criteria values. However, when the compressions are performed successfully, the compression ratios with a maximum water depth error of the order of a centimeter or less remain very important.

TABLE 8: TELEMAC REL TESTS

REL	Size	Compression time	Decompression time	Ratio	Check
0.001	19 MB	12'54"	5'41"	8726	YES
10^{-6}	2.5 GB	13'55"	5'41"	110	YES
10^{-9}	19 GB	19'51"	19'44"	8	YES
10^{-12}	NA	NA	NA	NA	NA
10^{-15}	NA	NA	NA	NA	NA

TABLE 9: TELEMAC PW_REL TESTS

PW_REL	Size	Ratio	Max. error (m)
10^{-1}	425 MB	381	17.883
10^{-2}	1.5 GB	110	1.8572
10^{-3}	2.8 GB	57	0.1842
10^{-4}	6.2 GB	26	0.0186
10^{-5}	11 GB	14	0.0018
10^{-6}	NA	NA	NA

V. CONCLUSIONS

The purpose of this article is to test a lossy compression tool for shrinking Telemac data. The tool used makes it possible to control the error made by the loss of accuracy. Different error control criteria were tested on two hydraulic data cases. It is shown that files can be compressed heavily compared to standard tools with an acceptable loss of information in many cases. Nevertheless, the consequence of this loss of information is not investigated here whereas it could influence the results of a study, for instance with a poor state initialization of the Telemac computations.

Not all the tests could be completed due to the memory usage that may be too high for the computer used for the tests. One possible solution without changing computers is to split the data files into several pieces so that they are processed separately. This approach has been tested with success with a parallel distribution of compressions on files larger than 1 terabyte.

Finally, not all SZ options have been tested. In particular, it is possible to consider the dependence of the data between each time step in order to achieve better compression performance.

REFERENCES

[1] S. Di and F. Cappello, "Fast Error-Bounded Lossy HPC Data Compression with SZ," 2016 IEEE International Parallel and Distributed Processing Symposium (IPDPS), 2016, pp. 730-739, doi: 10.1109/IPDPS.2016.1