Maintaining trust in reduction: preserving the accuracy of quantities of interest for lossy compression

Qian Gong[1], Xin Liang[2], Ben Whitney[1], Jong Youl Choi[1], Jieyang Chen[1], Lipeng Wan[1], Stéphane Ethier[3], Seung-Hoe Ku[3], R. Michael Churchill[3], C.-S. Chang[3], Mark Ainsworth[4], Ozan Tugluk[4], Todd Munson[5], David Pugmire[1], Richard Archibald[1], and Scott Klasky[1]

[1] Oak Ridge National Laboratory, Oak Ridge TN 37830, USA
[2] Missouri University of Science and Technology, Rolla MO 65409, USA
[3] Princeton Plasma Physics Laboratory, Princeton NJ 08540, USA
[4] Brown University, Providence RI 02912, USA
[5] Argonne National Laboratory Lemont IL 60439, USA

Abstract. As the growth of data sizes continues to outpace computational resources, there is a pressing need for data reduction techniques that can significantly reduce the amount of data and quantify the error incurred in compression. Compressing scientific data presents many challenges for reduction techniques since it is often on non-uniform or unstructured meshes, is from a high-dimensional space, and has many Quantities of Interests (QoIs) that need to be preserved. To illustrate these challenges, we focus on data from a large scale fusion code, XGC. XGC uses a Particle-In-Cell (PIC) technique which generates hundreds of PetaBytes (PBs) of data a day, from thousands of timesteps. XGC uses an unstructured mesh, and needs to compute many QoIs from the raw data, f.

One critical aspect of the reduction is that we need to ensure that QoIs derived from the data (density, temperature, flux surface averaged momentums, etc.) maintain a relative high accuracy. We show that by compressing XGC data on the high-dimensional, nonuniform grid on which the data is defined, and adaptively quantizing the decomposed coefficients based on the characteristics of the QoIs, the compression ratios at various error tolerances obtained using a multilevel compressor (MGARD) increases more than ten times. We then present how to mathematically guarantee that the accuracy of the QoIs computed from the reduced f is preserved during the compression. We show that the error in the XGC density can be kept under a user-specified tolerance over 1000 timesteps of simulation using the mathematical QoI error control theory of MGARD, whereas traditional error control on the data to be reduced does not guarantee the accuracy of the QoIs.

Keywords: lossy compression · error control · quantities of interest · XGC simulation data

1 Challenges in Lossy Compression for Physics Simulations

Storage and I/O capacities have not increased as rapidly as computational power over the last decade. Storage constraints influence how many files can be output, their frequency, and how long the output files can be kept in short-term storage like parallel file systems. With the exascale computing era approaching, there has been an urgent call for general and reliable reduction techniques that achieve large compression ratios for scientific applications. The compression of scientific data is challenging in several aspects. First, most scientific data are stored in 32- or 64-bit floating-point format. As the low-order bits of floating-point numbers are essentially random, lossless compression algorithms can only achieve limited compression ratios on most scientific data [10]. Second, compression algorithms targeting scientific data must provide guaranteed and quantifiable error bounds so that scientists can use the reduced data in their investigations and trust the results. Recently, several lossy compression algorithms for scientific data have been proposed based on prediction (SZ [11]), block transformation (ZFP [12]), multilevel decomposition (MGARD [13–15]), and machine learning (VAPOR [16]). This paper concerns the problem of compression under constraints on the errors incurred in quantities of interest (QoI), so we limit our study to error-bounded lossy compressors (i.e., MGARD, SZ, and ZFP).

Ideally, lossy compressors should be flexible with regard to the structure of the data, generalize to arbitrarily high dimensions, and allow control of errors both in the original degrees of freedom and in downstream QoIs. Scientific data usually resides on high-dimensional, underlying uniform, nonuniform, or unstructured grids [1, 4, 9, 22]. Compressing data in the same high-dimensional space where it is defined can make more of the data's spatial correlations visible to the compression algorithm, resulting in higher compression ratios. Similarly, compression algorithms should make use of as much of the data's spatial structure as possible. Compressin nonuniform or unstructured data as though it were defined on a uniform grid risks obscuring redundancies and patterns in the data, resulting in lower compression ratios. A third design goal is the control of errors incurred by compression algorithms. A natural starting point is to bound the error in the 'raw' data—i.e., the difference between the original dataset and the reduced dataset output by the compression algorithm. Often, though, scientists are less concerned with the pointwise error in the raw data than with the change to the QoIs computed from the data. The mathematics required to relate errors in the raw data to errors in QoIs is nontrivial, especially for QoIs that are nonlinear and/or obtained by complex post-processing. Empirical approaches can provide estimates for, but not guaranteed bounds on, QoI errors by extrapolating from previously encountered datasets and QoIs.

In this paper we focus our attention on XGC [1, 2], a leadership-class application in the fusion community which simulates high-dimensional data (five phase space dimensions + time) on an unstructured mesh and whose output is used to compute many simple and complicated, linear and nonlinear QoIs. XGC is a full-f gyrokinetic particle-in-cell (PIC) code which specializes in simulating kinetic

transport in edge tokamak plasmas, where strong particle and energy sources and sinks drive the plasma away from thermal equilibrium. The code represents particles as samples with specific positions, velocities and weights [21] and solves the gyrokinetic equations for a 5-dimensional particle distribution function f. XGC can run in parallel on supercomputers such as the Oak Ridge Leadership Computing Facility's (OLCF) Summit [20], fully utilizing all of the CPUs and GPUs. Although the parallel code enables fusion scientists to model more complicated tokamak experiments at finer resolution and for longer timescales, the data generated is too large for permanent storage on the parallel file system at OLCF. To give an idea of the scale, a simulation modelling ITER-scale [6] problems will typically contain trillions of particles and run for thousands of timesteps and can each day produce over 200 petabytes of data [5], which would, if stored in its entirety, fill 80% of the storage capacity of Summit's parallel file system.

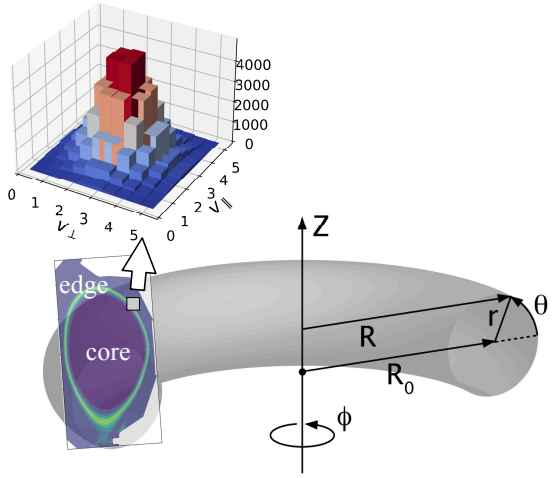


Fig. 1: Coordinates of the XGC 5D distribution function $f(\phi, \theta, r, v_{\parallel}, v_{\perp})$ in a tokamak device. To reduce the particle data, the output routine in XGC discretizes the 3D space $\{\phi, \theta, r\}$ into cross-sections uniformly spaced across the toroidal direction ϕ and groups the particles on each cross-section into histograms in 2D velocity space $v = \{v_{\parallel}, v_{\perp}\}$.

Direct compression of the XGC simulation data at such enormous scale is unrealistic due to the extreme compression ratio required. Instead, compression is preceded by a discretization and binning step in the XGC output routine. As shown in Figure 1, XGC discretizes the 3D space $x = \{r, z, \phi\}$ into radial-poloidal cross-sections (RZ planes) [7]. An unstructured mesh is used to model the common geometry of the RZ planes, and XGC groups the particles on each plane into volumes around the nodes of the mesh. The particles at each mesh node are further sorted into bins in the 2D Cartesian velocity space $v = \{v_{\parallel}, v_{\perp}\}$ [8].

The code then outputs a particle distribution histogram defined on this five-dimensional, nonuniform, unstructured grid, reducing the output size from hundreds of petabytes to hundreds of terabytes per run. This histogram data must be further reduced by compression techniques before being written to storage.

To shed light on the relationship between compression errors and downstream QoIs, we investigate in this paper the impacts of errors in the histogram data on five physical quantities specified by XGC scientists. The QoIs we consider are density (n), parallel temperature ($T_\parallel$), perpendicular temperature ($T_\perp$), flux surface averaged density ($n0$_avg), and flux surface averaged temperature ($T0$_avg). The density, n, is computed by integrating f over the mesh nodes,

$$n(x,t) = \int \frac{f(x,v,t)}{\mathrm{vox}(x)}\,\mathrm{d}v_\parallel\,\mathrm{d}v_\perp, \tag{1}$$

where t represents the timestep and $\mathrm{vox}(x)$ is the volume of the mesh node at x. $T_\parallel$ and $T_\perp$ are integrals of parallel and perpendicular kinetic energy, mv^2,

$$T_\perp(x,t) = \frac{1}{2}\int \frac{mv_\perp^2 f(x,v,t)}{n(x,t)}\,\mathrm{d}v_\parallel\,\mathrm{d}v_\perp,$$
$$T_\parallel(x,t) = \frac{1}{2}\int \frac{m(v_\parallel - u_\parallel)^2 f(x,v,t)}{n(x,t)}\,\mathrm{d}v_\parallel\,\mathrm{d}v_\perp,$$

where m is the mass of atomic particles corresponding to f and $u_\parallel$ is the parallel flow, computed using $u_\parallel(x,t) = \int v_\parallel f(x,v,t)/n(x,t)\,\mathrm{d}v_\perp\,\mathrm{d}v_\parallel$. Finally, the flux surface averaged density and temperature are computed by averaging the density and temperature over a thin volume between toroidal magnetic flux surface contours. For brevity, we omit the equations for the flux surface averaged quantities [3]. These five physical quantities represent different types of QoIs. Among these, density is a linear function of f, $T_\perp$ and $T_\parallel$ are nonlinear quantities, and $n0$_avg and $T0$_avg represent QoIs that are smoothed over the space. Throughout the paper, we measure the error using the Normalized Root Mean Square Error (NRMSE), defined as follows:

$$\mathrm{NRMSE}(u,\tilde{u}) = \frac{\sqrt{\sum_{i=0}^{N}(u_i - \tilde{u}_i)^2/N}}{\max(u) - \min(u)},$$

where u is the original data, $\tilde{u}$ is the reconstructed data, and N is the number of degrees of freedom in u. To ensure the integrity of scientific discovery, the downstream analysis of XGC requires that f and the above five physical quantities have errors no larger than 10^{-3}.

In this paper, we demonstrate the use of compression accounting for high dimension, various grid structures, and error control for different types of QoIs. Our application driver is data generated by XGC. We show that higher compression quality can be achieved when compressing in high dimension, allowing high-dimensional correlations to be utilized. We next show that the compression quality can be further improved when the decorrelation and quantization steps

of the compression algorithm are adapted to the nonuniform grid structure and to characteristics of QoIs to be preserved. Lastly, we show the importance of mathematically guaranteed error control for derived QoIs and how to achieve this control using MGARD.

The rest of the paper is organized as follows. In Section 2, we introduce the three error-controlled compressors that are evaluated in this work. In Section 3, we demonstrate the impact of utilizing high-dimensional decorrelation. In Section 4, we show how the grid can be leveraged to improve compression ratios using MGARD. In Section 5, we demonstrate how to adaptively compress the data for desired QoIs and how to guarantee the accuracy of the QoIs computed from the reduced data. In Section 6, we discuss future work. We conclude the paper in Section 7.

2 Background of Error-Controlled Lossy Compression

Compression techniques have been extensively studied as a direct way to reduce data size. Traditional lossless compressors [23–25] can recover full precision data but usually suffer from low compression ratios on floating-point scientific data [10]. Although lossy compressors widely used in the image processing community [32, 33] are able to trade off precision for higher compression ratios, they are not preferred in the scientific computing community as the errors are difficult to quantify. Hence, error-controlled lossy compression [13–15, 11, 12] was proposed as an alternative to significantly reduce the size of scientific data with guaranteed error control on the reconstructed data.

There are two popular categories of error-controlled lossy compressors, namely prediction- and transform-based models. The models differ in the method used to decorrelate the original data. Prediction-based models decorrelate the data using prediction or approximation techniques such as curve fitting, while transform-based models use domain transforms such as discrete cosine and wavelet transforms. After the decorrelation stage, data is quantized and then encoded for the actual byte reduction. In this paper, we focus on three error-controlled lossy compressors – SZ, ZFP, and MGARD – as they are recognized as the state of the art according to recent studies [27, 28, 26].

SZ leverages a general prediction-based lossy compression pipeline. Specifically, it first performs data decorrelation based on a multi-algorithm predictor, followed by error-controlled linear-scaling quantization and lossless encoding. ZFP is a transform-based lossy compression method. It splits d-dimensional data into blocks of size 4^d and compresses each independently. Data in a block is aligned to the same exponent, converted to a fixed-point format, and then decorrelated using a customized invertible transform. The transform coefficients are ordered by energy and encoded using embedded encoding. MGARD provides another elegant method for reducing scientific data. Due to the space limit, here we only sketch the key steps of the algorithm. MGARD decomposes the original data into a sequence of multilevel components using L^2 projection and multilinear interpolation. The multilevel coefficients encoding the multilevel components

are then quantized adaptively based on the target error metric. A more detailed introduction is available in [28] and the full mathematical treatment can be found in [14].

The unique features of MGARD include compression over nonuniform coordinate space, compression optimized for QoI preservation, and compression with mathematically guaranteed error control on certain linear QoIs. SZ and ZFP make predictions and transformations based on the assumption that the data are defined on a uniformly spaced d-dimensional Cartesian grid. MGARD does not rely on this assumption in its decorrelation step. Rather, it carries out the requisite projections and interpolations using the actual distances between mesh nodes and quantizes the multilevel coefficients using their actual volumes in the multilevel subgrid space. Moreover, MGARD can adaptively improve the compression quality of certain QoIs by tuning a smoothness parameter s. This parameter controls the bin widths used to quantize the multilevel coefficients, allowing more aggressive quantization of coefficients that will have little impact on the accuracy of the QoI. The smoothness parameter s can be further combined with the operator norm of the QoI to provide guaranteed error control [14] as follows:

$$|\mathcal{Q}(u) - \mathcal{Q}(\tilde{u})| \leq \Upsilon_s(\mathcal{Q}) \Big(\sum_{l=0}^{L} 2^{2sl} \sum_{x \in \mathcal{N}_l^*} \mathrm{vol}(x) |\mathtt{u\_mc[}x\mathtt{]} - \tilde{\mathtt{u}}\mathtt{\_mc[}x\mathtt{]}|^2 \Big)^{1/2} \quad (2)$$

where $\mathcal{Q}$ is the target bounded linear operator, $\Upsilon_s(\mathcal{Q})$ is its operator norm, $\mathrm{vol}(x)$ is the volume of the level l element centered at x, $\mathcal{N}_l^*$ is the collection of nodes in level l but not level $l-1$, and $\mathtt{u\_mc[}x\mathtt{]}$ and $\tilde{\mathtt{u}}\mathtt{\_mc[}x\mathtt{]}$ are the original and quantized multilevel coefficients at node x, respectively.

We conclude this section with a concrete description of the dataset used in our experiments. The particle distribution function f output by XGC comprises a velocity histogram at each node of each unstructured RZ plane mesh. SZ, ZFP, and MGARD all require data to be given on Cartesian grids or very particular triangulation/tetrahedration mesh structure [15], so none of the three can compress the XGC output in its original format. To enable the application of the compressors, we unroll the meshes in 2D unstructured RZ planes by radius, as the magnetic field in a tokamak tends to expand the plasmas outward along the major diameter. The spacing of the 1D grid is determined by the edge lengths of each unstructured mesh. This conversion changes the distribution function f from 5D to 4D: $\{\phi, \text{mesh nodes}, v_\perp, v_\parallel\}$. We conduct our experiments using a coarse resolution XGC dataset with 1000 simulation timesteps, each containing $\{8, 16395, 39, 39\}$ double-precision floating-point values. With the exception of the resolution, the dataset was generated using the same parameters and settings as high-resolution production runs. Throughout the paper, we always perform compression on f rather than on any QoI computed from f. We reduce f with a prescribed NRMSE tolerance, record the compression ratio, and compute the achieved NRMSE in either f or a QoI. The reported NRMSE is this observed error rather than the initial prescribed tolerance. If a figure calls for the compression ratio at an achieved NRMSE that is not exactly observed in our

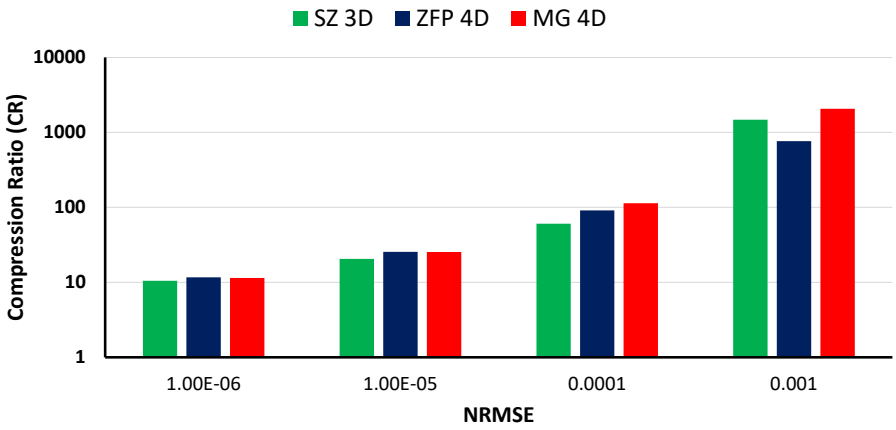
experiments, we compute an estimate by linearly interpolating the compression ratios at the neighboring measurements.

3 Error-Controlled Lossy Compression in High-Dimensional Space

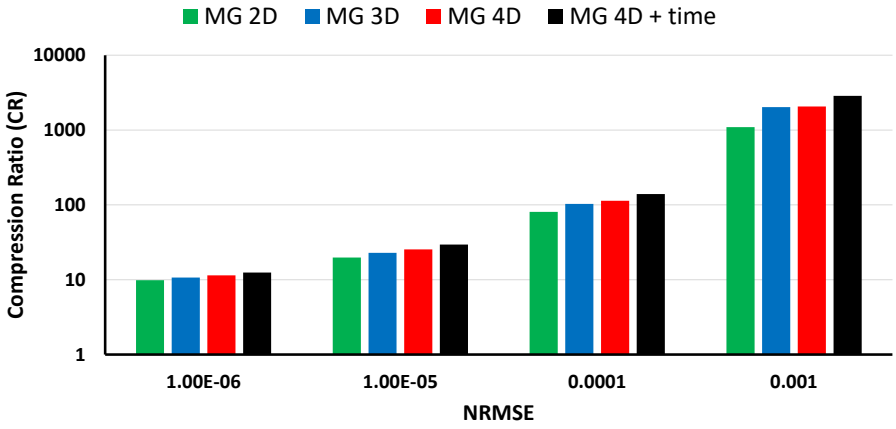
This section demonstrates how the compression of f can be improved by utilizing data correlations in high-dimensional space. In our first experiment, we apply SZ, ZFP, and MGARD to the 700^{th} timestep of f, an array with dimensions $\{8, 16395, 39, 39\}$. ZFP currently supports 1D, 2D, 3D, and 4D input, and MGARD supports input of arbitrary dimension, so these compressors can be applied to the dataset without modification. SZ currently supports 1D, 2D, and 3D input, so we interpret the dataset as an array with dimensions $\{8 \times 16395, 39, 39\}$ when applying SZ. Whether compressing in 3D or 4D, we must also decide how to handle the spacing between the data values. SZ and ZFP always interpret their input as an array defined on a uniform Cartesian grid. Because this experiment is focused on the impact of dimensionality, not spacing, we also use uniform node spacing in MGARD. Figure 2a shows the compression ratios achieved by each compressor. When the NMRSE level is low, e.g. 10^{-6}, the compression ratios of the three methods are similar, as the tight error bound does not provide enough flexibility for the algorithms to reduce the data. When the NMRSE is at the range of $[10^{-5}, 10^{-4}]$, ZFP 4D achieves better compression ratios than SZ 3D and worsen compression ratios than MGARD 4D. When the measured NRMSE keeps increasing, SZ 3D obtains better compression ratios than ZFP 4D but still worsen compression ratios than MGARD 4D. In our second experiment, we investigate the effect of dimension on the compression ratio achieved by MGARD. The results are shown in Figure 2b. When compressing in 4D plus time, we use timesteps 700 to 770. Across all tested NRMSE levels, the compression ratio increases with the dimension used for compression.

4 Error-Controlled Lossy Compression on Nonuniform Grids

In this section, we demonstrate that compressing using nonuniform grid can improve the accuracy of QoIs computed from the reduced data. We begin with a description of the nonuniform grid used. XGC uses an unstructured mesh, shown in Figure 3a, to model the complex RZ plane geometry and a nonuniform Cartesian grid, shown in Figure 3b, to discretize the velocity space. As described in Section 2, we unroll the unstructured mesh into a nonuniform 1D grid to allow compression of the data by existing methods. The result is a Cartesian grid with coordinates $\{\phi, \text{mesh nodes}, v_\perp, v_\parallel\}$ which is nonuniform in all coordinates except ϕ. SZ and ZFP do not support nonuniform grids, so we only use MGARD for the experiments in this section. Within MGARD, the nonuniform grid spacing impacts the decorrelation, where the central operations of L^2 projection and



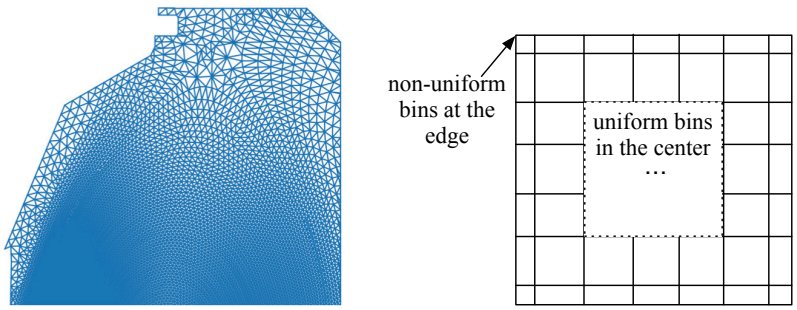
a Compression ratios at which f meets various NRMSE levels when f is compressed on a uniform Cartesian grid with SZ 3D, ZFP 4D, and MGARD 4D. The color of each bar indicates the compressor used. The horizontal position indicates the achieved NRMSE in f. The height indicates the achieved compression ratio.



b Compression ratios achieved when f is compressed on a uniform Cartesian grid with MGARD 2D, MGARD 3D, MGARD 4D, and MGARD 4D plus time.

Fig. 2: Illustration of the improvement in compression ratios achieved by compressing f in high-dimensional space. a shows the compression ratios achieved when compressing the 700^{th} timestep of f with SZ, ZFP, and MGARD using the highest dimension supported by each compressor. b shows the compression ratios achieved using MGARD in 2D, 3D, 4D, and 4D plus time.

multilinear interpolation depend on the grid, and the quantization, where errors are prorated according to the spacing at each node.



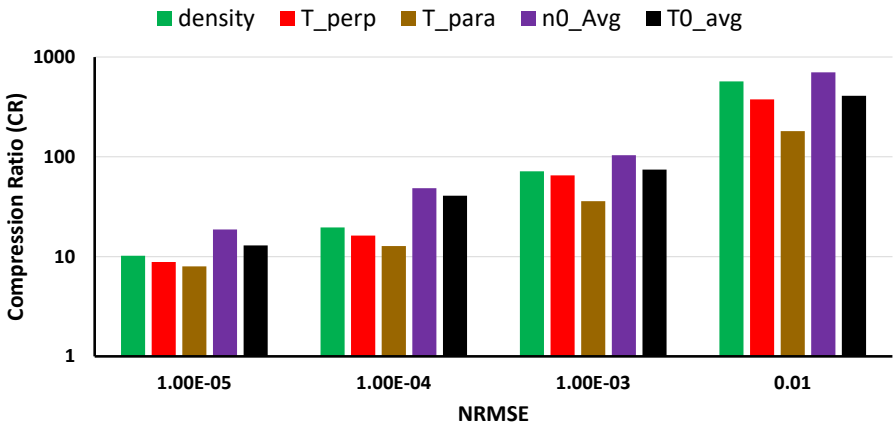
a Unstructured RZ plane mesh. b Nonuniform velocity space grid.

Fig. 3: Illustration of the unstructured mesh and nonuniform grid on which the XGC data are defined. XGC uses an unstructured mesh to model the complex geometry of radial-poloidal planes and Cartesian bins to model the discretized velocity space. The nodes of the 2D unstructured RZ plane mesh are linearized to a 1D nonuniform grid and combined with the velocity space for compression.

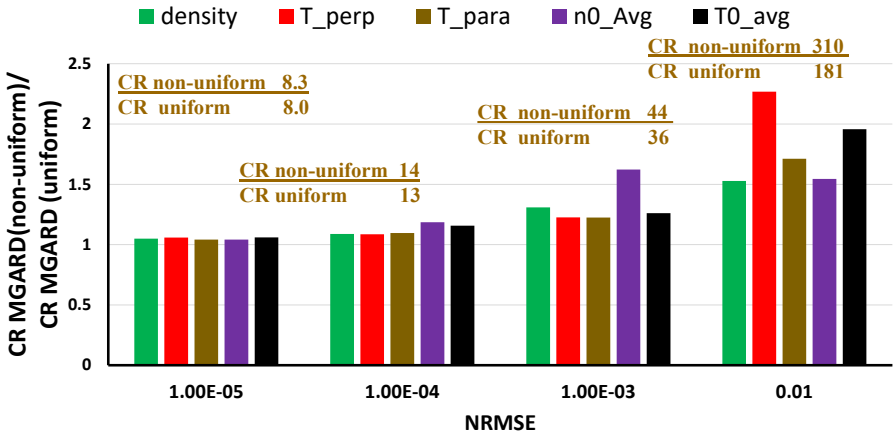
In our first experiment, we establish a baseline by measuring the QoI errors when compressing using uniform grid spacing. We apply MGARD 4D to the 700^{th} timestep of the XGC data with a variety of error tolerances, yielding a collection of reduced datasets with different achieved compression ratios. For each reduced dataset, we compute the five QoIs described in Section 1 and calculate the achieved NRMSEs. These achieved NRMSEs are generally different, so that the compression ratios at which the QoIs meet a given NRMSE level are likewise generally different. The results of the experiment are shown in Figure 4a. At all NRMSE levels tested, the achieved compression ratio is lowest for $T_{\|}$. This is the consequence of $T_{\|}$ generally exhibiting the highest NRMSE for a given reduced dataset, as a result of its non-linear computation and the complicated structure of the large parallel velocity components. To ensure the fidelity of downstream analysis, the physicists using XGC require that all QoIs simultaneously satisfy appropriate error bounds. If the prescribed NRMSE tolerance is the same for all QoIs, the results of this experiment suggest that the achievable compression ratio will be determined by the error in $T_{\|}$.

Our second experiment is identical to the first except that we compress using nonuniform grid spacing. Providing the spacing information to MGARD allows the quantizer to adjust the error bound used for a node according to the volume of the corresponding element. Larger elements result in tighter error bounds; see Equation (2). Accordingly, the coefficients that represent variations over small regions will be compressed more heavily than the coefficients that represent variations over large regions. We expect that this adaptive, nonuniform error quantization will benefit the errors observed in the QoIs. The results of the experiment are shown in Figure 4b. Using nonuniform grid spacing in the

compression lifts the compression ratios at which all five QoIs meet the NRMSE levels, and the benefit increases as the NRMSE grows larger.



a Compression ratios at which the QoIs meet various NRMSE levels when f is compressed on a uniform Cartesian grid with MGARD 4D. The color of each bar indicates the QoI. The horizontal position indicates the achieved NRMSE in the QoI. The height indicates the achieved compression ratio.



b Ratios of compression ratios at which NRMSE levels are met when using nonuniform spacing to those at which the levels are met when using uniform spacing. The compression ratios given above each group of bars are those at which all of the QoIs meet the NMRSE level.

Fig. 4: Comparison between using uniform and nonuniform spacing when compressing the 700$^{\text{th}}$ timestep of f with MGARD 4D. a shows the results obtained using uniform spacing; b compares those results with those obtained using nonuniform spacing.

5 Error-Controlled Lossy Compression for QoIs

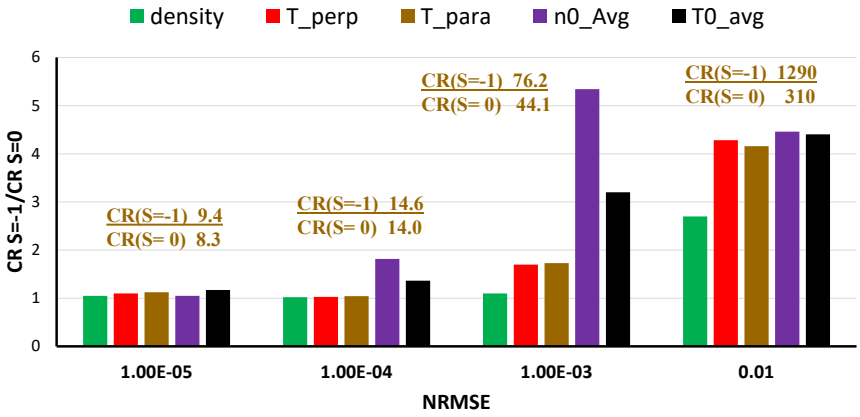
In this section we show how adapting MGARD's compression algorithm (specifically, the quantization stage) to a QoI can reduce the error incurred in that QoI at a given compression ratio. This adaptive quantization is a unique feature of MGARD. The multilinear decomposition of MGARD decorrelates the data using a mesh hierarchy. The low and high frequency components of the data are represented by the multilevel coefficients on the coarse and fine levels of the hierarchy, respectively. A smoothness parameter s can be used to change the bin widths used to quantize the multilevel coefficients on the different levels. $s = 0$ is the baseline. When $s < 0$, MGARD imposes relatively tighter error bounds on the coarse level coefficients (low frequency components) and relatively looser error bounds on the fine level coefficients (high frequency components); see Equation (2). As a result, taking $s < 0$ tends to yield larger compression ratios for QoIs that are determined by low frequency components. Conversely, taking $s > 0$ generally benefits QoIs that are sensitive to high frequency components.

Our first experiment focuses on the effect of the smoothness parameter s on the errors. We apply MGARD with $s = -1, 0, 1$ to f and measure the compression ratios at which the QoIs meet various NRMSE levels. The results are shown in Figure 5. For all five QoIs, taking $s = -1$ leads to the best results, with the average quantities $n0$_avg and $T0$_avg benefiting most from that choice of smoothness parameter. This outcome may be explained by the relative insensitivity of the QoIs considered to the high frequency components of the data; as seen in Section 1, all can be written as convolutions with functions that are not highly oscillatory. As was the case with nonuniform compression in the experiment in Section 4, the improvement from adaptive quantization is more significant at higher NRMSE levels.

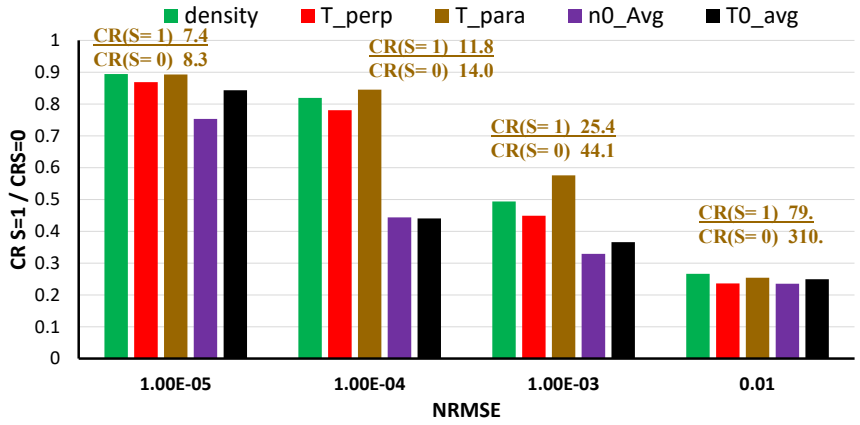
Our next experiment compares MGARD 4D with nonuniform spacing and $s = -1$, ZFP 4D, and SZ 3D. Recall that ZFP and SZ support compression in up to four and three dimensions, respectively. As before, we compress f and measure the errors in the QoIs. The results are presented in Figure 6. As seen in Figure 6a, MGARD 4D achieves higher compression ratios than ZFP 4D at the same NRMSE level. MGARD 4D also outperforms SZ 3D in this example, as seen in Figure 6b, with the discrepancy being larger. In both comparisons, the advantage increases as the NRMSE becomes larger.

In our third experiment, we compare MGARD 4D with nonuniform spacing and $s = -1$ to MGARD 2D with uniform spacing and $s = 0$, the baseline configuration. The results are shown in Figure 7. For each configuration and each of four NRMSE levels, we measure the maximum compression ratio at which all five QoIs meet the prescribed error level. These maximum compression ratios for MGARD 4D with nonuniform spacing and $s = -1$ are improved 1.24×, 1.47×, 4.62×, and 21.6× over the baseline at NRMSE levels 10^{-5}, 10^{-4}, 10^{-3}, and 10^{-2}, respectively. The degree of improvement increases with the NRMSE level and is more significant for the average quantities $n0$_avg and $T0$_avg.

Most state-of-the-art compression algorithms, with the exception of MGARD in certain linear cases, only support error bounds set on the 'raw' data. If users



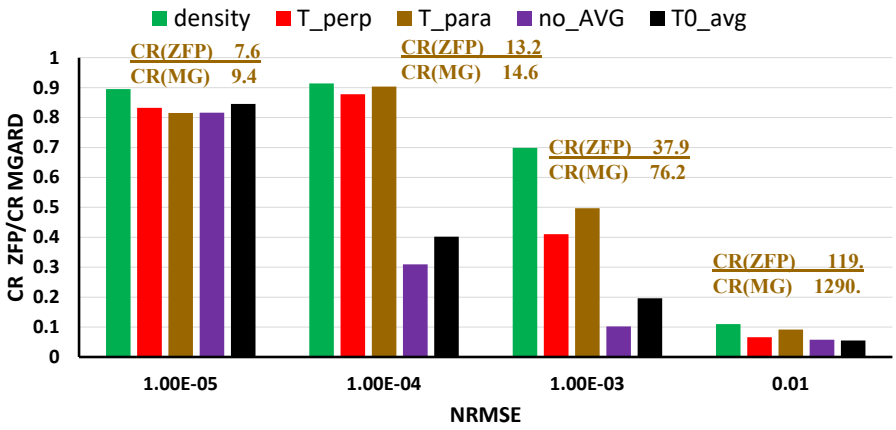
a Ratios of compression ratios at which NRMSE levels are met when using $s = -1$ to those at which the levels are met when using $s = 0$.



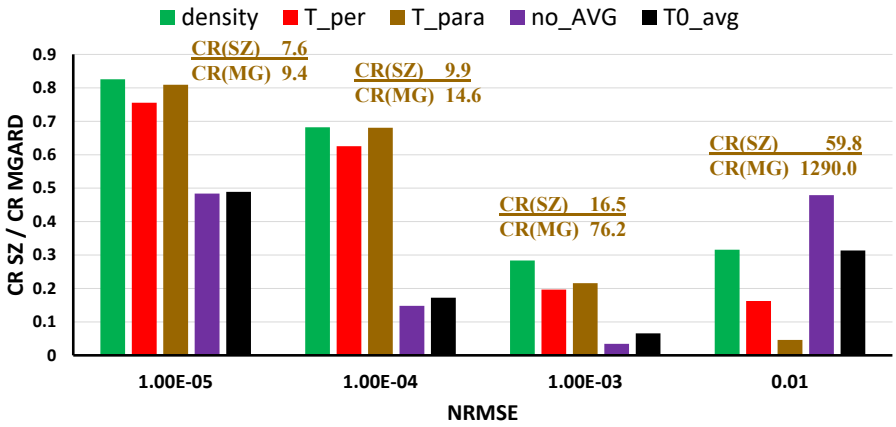
b Ratios of compression ratios at which NRMSE levels are met when using $s = 1$ to those at which the levels are met when using $s = 0$.

Fig. 5: Illustration of the effect of the smoothness parameter s when compressing the 700^{th} timestep of f with MGARD 4D with nonuniform spacing. For $s = -1, 0, 1$, we compress f and measure the compression ratios at which the QoI meet various NRMSE levels. a shows the ratios of the compression ratios obtained when using $s = -1$ to those obtained when using $s = 0$; b shows the ratios of the compression ratios obtained when using $s = 1$ to those obtained when using $s = 0$.

want the errors in QoIs to stay below a certain threshold, they have to estimate the corresponding error bound on the raw data through empirical studies. Empirical relations, though, may not continue to hold as the data distribution changes. Our fourth experiment concerns the relationship between the error bound on the



a Ratios of compression ratios at which NRMSE levels are met when using ZFP 4D to those at which the levels are met when using MGARD 4D with nonuniform spacing and $s = -1$.



b Ratios of compression ratios at which NRMSE levels are met when using SZ 3D to those at which the levels are met when using MGARD 4D with nonuniform spacing and $s = -1$.

Fig. 6: Comparison between MGARD, SZ, and ZFP on the 700^{th} timestep of f. Each compressor is used with its best settings: for MGARD, 4D, nonuniform spacing, and $s = -1$; for ZFP, 4D and uniform spacing; and for SZ, 3D and uniform spacing. a compares ZFP to MGARD; b compares SZ to MGARD.

raw data and the measured error in QoIs. We apply MGARD to each of the 1000 timesteps of f with an NRMSE tolerance of 10^{-4}. We then compute the NRMSE of f and two QoIs, density n and parallel temperature $T_{\|}$. The results are shown in Figure 8. The error in f stays below the prescribed error bound, but the er-

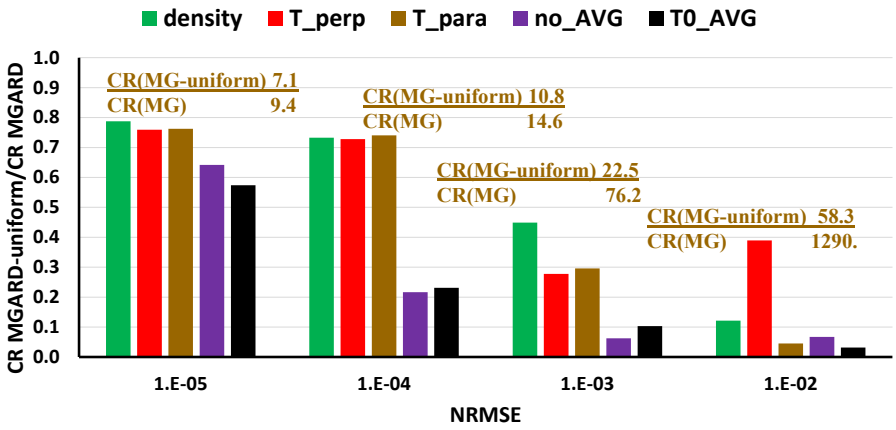


Fig. 7: Illustration of the improvement in compression ratios due to compressing f in high-dimensional, nonuniform space and catering the error quantization to the characteristics of the QoIs. `CR(MG-uniform)` is obtained with MGARD 2D using uniform spacing and $s = 0$. `CR(MG)` is obtained with MGARD 4D using nonuniform spacing and $s = -1$.

rors in density and temperature rise above 10^{-4} and 10^{-3} over time. It is not entirely unexpected to observe these increased errors in the QoIs, since the distribution functions become more complicated as the turbulence builds up, but it does make empirical error control challenging. The data generated by many scientific experiments and simulations, including XGC, would be costly or even impossible to reproduce if the errors in QoIs were later found to be unacceptable, so compressors should mathematically guarantee that the errors in QoIs will respect user-specified tolerances.

MGARD allows error bounds set on scalar, linear QoIs $\mathcal{Q}$. To enable this, the operator norm $\Upsilon_s(\mathcal{Q})$ is first computed using a procedure developed in [14]. $\Upsilon_s(\mathcal{Q})$ represents the maximum growth from the compression error $\|f - \tilde{f}\|_s$ to the QoI error $|\mathcal{Q}(f) - \mathcal{Q}(\tilde{f})|$. By reducing f using an error bound of $\epsilon/\Upsilon_s(\mathcal{Q})$, MGARD can then ensure that the error in $\Upsilon_s(\mathcal{Q})$ is no more than ϵ. Our final experiment is identical to the timestep experiment described above except that we set the error bound on density rather than on f. We compute the operator norm $\Upsilon_s(\mathcal{Q})$ for density, set ϵ=10^{-3}, then use $\epsilon/\Upsilon_s(\mathcal{Q})$ to compress f in the 2D velocity space and compute density and parallel temperature from the reduced data. To measure the error, here we use the relative L^∞ norm, another commonly used error metric, which is given by $\|f-\tilde{f}\|_{L^\infty}/[\max(f)-\min(f)]$ (and likewise for the QoIs). Note that by keeping the relative L^∞ error smaller than ϵ we guarantee that the NRMSE of the density field will be smaller than ϵ as well. The results are shown in Figure 9. The density is successfully preserved to within the error tolerance at every timestep. We also plot the error in a nonlinear QoI, $T_{\|}$, which

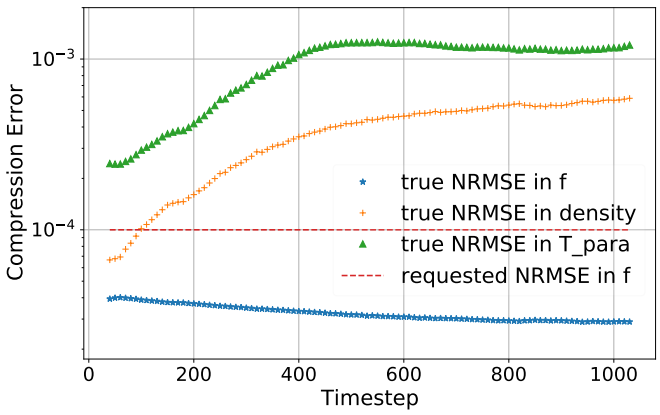


Fig. 8: Illustration of the error in f, density n, and parallel temperature $T_{\|}$ when compressing 1000 timesteps of f using MGARD with an NRMSE tolerance of 10^{-4}. The NRMSE of f stays relatively flat over the simulation, but the NRMSE of density and temperature increase as the turbulence builds up. The compression ratio of f ranges from 25× to 50× and rises over time.

MGARD cannot control. As expected, bounding the error in density does not guarantee that the error in $T_{\|}$ will stay below the requested error tolerance.

6 Future Work

State-of-the-art compressors that support compression on arbitrary unstructured meshes do not exist. To enable the compression of XGC data, we convert the unstructured 2D RZ plane to nonuniform 1D grid, potentially lowering the compression efficacy. In the previous work [15], we develop compression algorithms for data defined on meshes formed by particular types of triangulations/tetrahedrations. We plan to expand that work to support data defined on arbitrary unstructured meshes.

Furthermore, the different error trend of f and QoIs, as shown in Figure 8, suggests that empirically error control for QoIs is unreliable. In our previous work on MGARD, we develop algorithms that guarantee the preservation of certain linear QoIs, but the error control for complicated nonlinear QoIs (e.g., $T_{\|}$ in Figure 9) remains to be understood and developed. Our future work will include formulating new theories to provide guaranteed error bounds for multiple and nonlinear QoIs.

This work does not discuss the throughput performance of lossy compressors as the trade-offs between throughput and reduction ratios is not the focus of the paper. Our evaluations in this paper were conducted using sequential implementations, but compression techniques can be accelerated on different

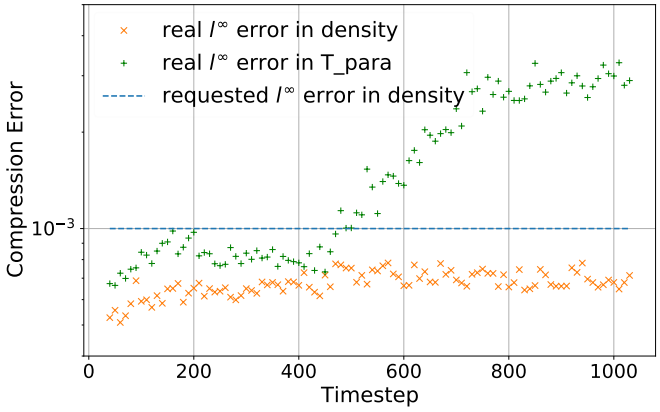


Fig. 9: Illustration of the error in density n and parallel temperature $T_\parallel$ when compressing 1000 timesteps of f using MGARD with an error bound set on density. The L^∞ error in density remains under the requested threshold and is relatively flat. The L^∞ error in $T_\parallel$, which is a nonlinear QoI that MGARD cannot control, rises above the requested error tolerance. The compression ratio ranges from 16.5× to 19.5× and decreases over time.

hardware [29–31]. Due to the page limit, we leave the evaluation on accelerated lossy compression to the future work.

7 Conclusion

Lossy compression can be used to reduce the cost of scientific data transmission and storage. To ensure that the lossy compression does not weaken the integrity of downstream analysis, the compression must be able to control the uncertainties in both 'raw' data and derived QoIs. In this paper we focus on the data generated by XGC, which is high in dimension, nonuniform in spacing, and can be used to compute many simple and complex, linear and nonlinear QoIs. Choosing the particle distribution function f and its five derived QoIs as the example, we show that the better compression quality can be achieved when high-dimensional data correlation is utilized, and the decorrelation and quantization steps of the compression algorithm are adapted to the nonuniform grid structure and to the characteristics of QoIs. We discuss the importance of having mathematically guaranteed error control for linear and nonlinear QoIs. We demonstrate that we can conserve the accuracy of linear QoI (i.e., density) over 1000 timesteps of XGC simulation using the QoI error control theory of MGARD, whereas the empirical studies with error control on 'raw' data cannot guarantee the error in QoIs.

Acknowledgement

This research was supported by the ECP CODAR, Sirius-2, and RAPIDS-2 projects through the Advanced Scientific Computing Research (ASCR) program of Department of Energy, and the LDRD project through DRD program of Oak Ridge National Laboratory.

References

1. Chang, Choong-Seock, et al. "Spontaneous rotation sources in a quiescent tokamak edge plasma." Physics of Plasmas 15.6 (2008): 062510.
2. Chang, Choong-Seock, et al. "Compressed ion temperature gradient turbulence in diverted tokamak edge." Physics of Plasmas 16.5 (2009): 056108.
3. Hager, Robert, et al. "Gyrokinetic study of collisional resonant magnetic perturbation (RMP)-driven plasma density and heat transport in tokamak edge plasma using a magnetohydrodynamic screened RMP field." Nuclear Fusion 59.12 (2019): 126009.
4. Jesse, Stephen, et al. "Using Multivariate Analysis of Scanning-Rochigram Data to Reveal Material Functionality." Microscopy and Microanalysis 22.S3 (2016): 292-293.
5. https://www.olcf.ornl.gov/2021/02/18/scientists-use-supercomputers-tostudy-reliable-fusion-reactor-design-operation. (2021) Online.
6. Rebut, P-H. "ITER: the first experimental fusion reactor." Fusion engineering and design 30.1-2 (1995): 85-118.
7. Ku, Seung-Hoe, et al. "Full-f gyrokinetic particle simulation of centrally heated global ITG turbulence from magnetic axis to edge pedestal top in a realistic tokamak geometry." Nuclear Fusion 49.11 (2009): 115021.
8. Dominski, J., et al. "Spatial coupling of gyrokinetic simulations, a generalized scheme based on first-principles." Physics of Plasmas 28.2 (2021): 022301.
9. Wolfram Jr, et al. "Global to Coastal Multiscale Modeling via Land-river-ocean Coupling in the Energy Exascale Earth System Model (E3SM)." No. LA-UR-20-24263. Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2020.
10. Ratanaworabhan, et al. "Fast lossless compression of scientific floating-point data." Data Compression Conference (DCC'06). 2006.
11. Liang, Xin, et al. "Error-controlled lossy compression optimized for high compression ratios of scientific datasets." 2018 IEEE International Conference on Big Data (Big Data). IEEE, 2018.
12. Lindstrom, Peter. "Fixed-rate compressed floating-point arrays." IEEE transactions on visualization and computer graphics 20.12 (2014): 2674-2683.
13. Ainsworth, Mark, et al. "Multilevel techniques for compression and reduction of scientific data—the multivariate case." SIAM Journal on Scientific Computing 41.2 (2019): A1278-A1303.
14. Ainsworth, Mark, et al. "Multilevel techniques for compression and reduction of scientific data-quantitative control of accuracy in derived quantities." SIAM Journal on Scientific Computing 41.4 (2019): A2146-A2171.
15. Ainsworth, Mark, et al. "Multilevel Techniques for Compression and Reduction of Scientific Data—The Unstructured Case." SIAM Journal on Scientific Computing 42.2 (2020): A1402-A1427.

16. Choi, Jong, et al. “Generative fusion data compression.” Neural Compression: From Information Theory to Applications–Workshop ICLR (2021)
17. https://github.com/CODARcode/MGARD/blob/master/README_MGARD_GPU.md
18. https://github.com/LLNL/zfp
19. https://github.com/szcompressor/SZ
20. Hines, Jonathan. “Stepping up to summit.” Computing in science & engineering 20.2 (2018): 78-82.
21. Faghihi, Danial, et al. “Moment preserving constrained resampling with applications to particle-in-cell methods.” Journal of Computational Physics 409 (2020): 109317.
22. Jackson, Matthew, et al. “Reservoir modeling for flow simulation by use of surfaces, adaptive unstructured meshes, and an overlapping-control-volume finite-element method.” SPE Reservoir Evaluation & Engineering 18.02 (2015): 115-132.
23. Alted, Francesc. “Blosc, an extremely fast, multi-threaded, meta-compressor library.” (2017).
24. Burtscher, Martin, et al. “FPC: A high-speed compressor for double-precision floating-point data.” IEEE Transactions on Computers 58.1 (2008): 18-31.
25. https://facebook.github.io/zstd/ Online (last accessed 2021)
26. Chen, Jieyang, et al. “Understanding Performance-Quality Trade-offs in Scientific Visualization Workflows with Lossy Compression.” 2019 IEEE/ACM 5th International Workshop on Data Analysis and Reduction for Big Scientific Data (2019).
27. Lu, Tao, et al. “Understanding and modeling lossy compression schemes on HPC scientific data.” 2018 IEEE International Parallel and Distributed Processing Symposium (IPDPS). IEEE, 2018.
28. Liang, Xin, et al. “MGARD+: Optimizing Multi-grid Based Reduction for Efficient Scientific Data Management.” to appear in IEEE Transactions on Computers (2021)
29. Chen, Jieyang, et al. “Accelerating Multigrid-based Hierarchical Scientific Data Refactoring on GPUs.” arXiv preprint arXiv:2007.04457 (2020).
30. Tian, Jiannan, et al. “Cusz: An efficient gpu-based error-bounded lossy compression framework for scientific data.” Proceedings of the ACM International Conference on Parallel Architectures and Compilation Techniques. 2020.
31. Lindstrom, Peter, et al., cuZFP: https://github.com/LLNL/zfp/tree/develop/src/cuda_zfp
32. Wallace, Gregory K. “The JPEG still picture compression standard.” IEEE transactions on consumer electronics 38.1 (1992): xviii-xxxiv.
33. Rabbani, Majid. “JPEG2000: Image compression fundamentals, standards and practice.” Journal of Electronic Imaging 11.2 (2002): 286.