A Methodology for Evaluating the Impact of Data Compression on Climate Simulation Data

Allison H. Baker
National Center for Atmospheric Research
Boulder, CO
abaker@ucar.edu

Haiying Xu
National Center for Atmospheric Research
Boulder, CO
haiyingx@ucar.edu

John M. Dennis
National Center for Atmospheric Research
Boulder, CO
dennis@ucar.edu

Michael N. Levy
National Center for Atmospheric Research
Boulder, CO
mlevy@ucar.edu

Doug Nychka
National Center for Atmospheric Research
Boulder, CO
nychka@ucar.edu

Sheri A. Mickelson
National Center for Atmospheric Research
Boulder, CO
mickelso@ucar.edu

ABSTRACT

High-resolution climate simulations require tremendous computing resources and can generate massive datasets. At present, preserving the data from these simulations consumes vast storage resources at institutions such as the National Center for Atmospheric Research (NCAR). The historical data generation trends are economically unsustainable, and storage resources are already beginning to limit science objectives. To mitigate this problem, we investigate the use of data compression techniques on climate simulation data from the Community Earth System Model. Ultimately, to convince climate scientists to compress their simulation data, we must be able to demonstrate that the reconstructed data reveals the same mean climate as the original data, and this paper is a first step toward that goal. To that end, we develop an approach for verifying the climate data and use it to evaluate several compression algorithms. We find that the diversity of the climate data requires the individual treatment of variables, and, in doing so, the reconstructed data can fall within the natural variability of the system, while achieving compression rates of up to 5:1.

Categories and Subject Descriptors

E.4 [**Coding and Information Theory**]: Data compaction and compression; H.3.m [**Information and Storage Retrieval**]: Miscellaneous; D.2.4 [**Software and Program Verification**]: Validation

Keywords

data compression, high performance computing

HPDC'14, June 23–27, Vancouver, BC, Canada.

http://dx.doi.org/10.1145/2600212.2600217 .

1. INTRODUCTION

The Community Earth System Model (CESM) is an important and widely-used earth system model whose development is centered at the National Center for Atmospheric Research. Model simulations from CESM, which we will refer to as data, are used by scientists around the world as well as by the Intergovernmental Panel on Climate Change (IPCC). To illustrate the potential size of a data set, we note that a recent high-resolution CESM simulation generated on the order of one terabyte of data per compute day (corresponding to half a terabyte of data per simulation year) [17]. CESM simulation data are written to "history files" in time slices at pre-defined sampling rates that vary by variable and model component (e.g., the ocean, the atmosphere, the land, etc.). The history files are in NetCDF format and are used for post-processing analysis. The files contain floating-point data that are truncated from double- to single-precision at the time they are written. Furthermore, in practice, climate scientists are often forced to save variables to history files less frequently than they wish due to storage considerations, and a low sampling rate results in even more data loss (see, e.g., [11]). Despite the fact that the simulation data in the history files has already been subjected to these two lossy processes, climate scientists have resisted applying compression algorithms to the history files. Our goal is to develop a suite of quality metrics that can assess whether or not the loss of information due to the application of lossy data compression is detectable within the context of climate data analysis. In other words, are the complete dataset and the reconstructed dataset (that has undergone lossy compression) statistically distinguishable? We focus on lossy compression as it provides a bigger advantage in terms of data reduction. Note that CESM also writes 'restart files" in full-precision (8-byte floating-point) that are used to continue a stopped simulation (i.e., checkpointing). We do not consider compressing restart files at this time, but will examine lossless techniques for these data in the future (as done in [12] for multi-physics simulations).

We are initially focusing on using compression as a means of reducing online and archive storage requirements, ie. "disk-compression" [12], as opposed to reducing the required I/O bandwidth of our application. Therefore, we examine compression with the intention of integrating it into a post-

processing step that converts the CESM time-slice data history files to time series data files for each variable. It is well known that losslessly compressing floating-point scientific data is difficult (see, e.g., [14, 1, 12, 11]), primarily due to the almost random (highly entropic) nature of the floating-point data. In particular, the significands of the floating point numbers often look random after the first several digits, depending on the type of data and whether the number of digits that are physically significant is less than the precision used by the model and the precision of the input data. Furthermore, while compression techniques have been studied extensively in areas such as image, video, and audio files, compression of scientific data has only received attention relatively recently. In addition, the metrics typical used to evaluate whether a lossy compression technique is acceptable for image compression may be quite different than what is required to verify scientific simulation data (see, e.g., [15]). Therefore, while performance metrics (such as compression speed and compression ratio) and average error metrics (such as peak signal-to-noise-ratio and root mean squared error) are of interest, of foremost importance is accurately accessing the impact of data compression on climate simulation data. In practical terms, if the reconstructed and the original climate simulation data are indistinguishable during the post-processing analysis, which includes both visualization and analytics, then the effects of compression fit within the natural variability of the system and applying compression is certainly a reasonable thing to do. In summary, we make the following key contributions:

- development of a thorough approach for the verification of climate data that has undergone compression and decompression, and
- evaluation of several existing compression techniques on CESM data, and
- demonstration that climate data compressed by as much as 5:1 can be reconstructed to be statistically indistinguishable from the original

This paper is organized as follows. We discuss related work in Section 2. In Section 3, we describe the attributes of compression methods that were chosen for our study. Our verification methodology is presented in Section 4, and results from compressing the CESM data are given in Section 5. We give concluding remarks in Section 6.

2. RELATED WORK

A number of lossless and lossy compression techniques have been proposed that have potential value for climate data. We give a brief overview of some of these approaches and discuss the metrics used to evaluate the lossy methods.

2.1 Lossless Methods

Traditional general-purpose lossless compression techniques (i.e., methods that exactly preserve the data), such as *gzip, bzip2*, and *lmza*, for example, are relatively ineffective on most scientific data and have motivated the development of more recent lossless approaches such as [14, 2, 16, 6]. In [14], the *fpzip* algorithm is presented, with a focus on lossless online compression to reduce bandwidth requirements for scientific data. This method uses predicative coding and can also be used in a lossy manner (by truncating a specified number of least significant bits when the floating-point values to be compressed are converted to integers). Burscher's lossless *FPC* method [2, 3] aims to simultaneously obtain both a good compression ratio and fast compression and decompression speeds with a predictive coding method that targets 64-bit values. However, his results show that the FPC method (as well as some other lossless methods) does not achieve good compression ratios on datasets with a large amount of randomness. An interesting alternative approach for lossless compression is the ISOBAR-compress method (In-Situ Orthogonal Byte Aggregation Scheme) [16]. *ISOBAR* is a preconditioner that operates on the data to be compressed in a manner that makes it more amenable to compression. Another preconditioner-type method is developed in [6] and applies binary masks to the dataset before the compression step. In addition, in [7], the lossless compression technique *MAFISC* is presented and evaluated on climate data from the German Weather Service (GWS) and CMIP5 [10]. *MAFISC* essentially acts as a preconditioner as well by applying multiple filters to the data before a standard compression method is used. *MAFISC* slightly improves upon the standard lossless method *lmza*, compressing the GWS data by about fifty percent.

2.2 Lossy Methods

Because of the inherent randomness in scientific data, a lossy technique (i.e., the reconstructed data will differ from the original) is typically needed to achieve useful compression rates. Recently, lossy techniques such as those in [21, 11, 1, 20, 12, 8, 9, 18] are being actively developed and applied to scientific datasets. As mentioned previously, *fpzip* [14] can be used in a lossy fashion, and *fpzip* is compared to the commercial software APAX (APplications AXceleration) [20] in [12]. Like *fpzip*, *APAX* also uses predictive encoding, but the two methods differ in their quantization method. The *fpzip* scheme results in a bounded relative error, while APAX bounds the absolute error. It is notable that in [12], the authors attempt to evaluate the impact of the compression by a 'physics-based' approach that specifically tailors a metric(s) for each of three different physics codes. The work by Iverson et. al in [9] is novel as well in that it achieves very good compression ratios on scientific data on unstructured grids by modeling the grid data as a graph and taking advantage of locality. The error metrics used in their study are the maximum pointwise error, the root mean squared error (RMSE) and the peak signal-to-noise ratio (PSNR). In [21], the main objective is to reduce the transfer time for large climate (more specifically, ocean) datasets. The authors use a JPEG2000 compression scheme (wavelet-based) and prefer to evaluate the error with a maximum pointwise error metric, rather than the RMSE. This wavelet-based compression also requires grid information, and the ocean data is compressed with a commercial JPEG2000 package in multiple 2-D slices after a quantization pre-processing step. The lossy *ISABELA* method in [11] appears attractive because it compresses data locally by first preconditioning the data to increase smoothness and then approximating with B-splines or wavelets. A threshold relative error is defined, and the quality is measured with a normalized RMSE and the Pearson correlation coefficient. Bicer et. al in [1] develop a new lossy compression approach, called *CC* ("Climate Compression"), that they apply to the GCRM (global cloud-resolving model) climate

dataset. Their compression method is a type of delta compression that takes advantage of spatial and temporal neighbor information and can also be used in a lossless manner. They emphasize integrating compression into a data processing or simulation application, and there is little discussion of validation beyond indicating the number of least significant bits that are dropped. In [8], the authors look at compressing climate data from the European climate model ECHAM using *GRIB2, APAX*, and *MAFISC*. *GRIB2* [5] is essentially a bit-oriented file format standard defined by the World Meteorological Organization that results in lossy data compression from the format conversion. These *GRIB2* files can then be further compressed with a standard compression method, typically JPEG2000 (see [8] or [18] for further description). The metrics for comparison in [8] are compression speed, compression ratio, number of bits of precision, and a so-called signal-to-residual ratio (SRR), which is the ratio between the standard deviation of the data and the standard deviation of the point-wise error in the reconstructed data. Finally, we note that in [18], Sullivan explores finding a suitable compression technique for meteorological data, detailing existing compression software options and highlighting the difficulties in meeting all of a user's requirements.

Although compression of simulation data from a climate model has been explored in [8], [21], and [1], the verification of the resulting reconstructed climate dataset has not been sufficiently addressed for our purposes. While quantifying the maximum pointwise error (or number of significant digits) or average error results in useful information, our goal of incorporating compression into the CESM workflow requires a more comprehensive analysis strategy. To our knowledge, the only work on compressing scientific data that, similar to our effort, emphasizes data verification is that of Laney et. al [12], where specific application-based metrics are developed for several physics codes at Livermore National Laboratory.

3. SELECTION OF ALGORITHMS

3.1 Criteria

We considered several factors similar to Sullivan's study of meteorological data in [18] to choose and evaluate compression algorithms. First, climate models typically contain a large and diverse set of variables. Some variables may have small ranges, while others are quite large, and the magnitudes may be quite different for each variable even within a particular component. For example, in the community atmosphere model (CAM), the sulfur dioxide variable (SO2) has a maximum value of $\mathcal{O}(10^{-8})$, whereas the maximum cloud condensation concentration at $S = 0.1\%$ (CCN3) is $\mathcal{O}(10^{3})$. Some variables are smoother than others, and many missing or special values exist. For example, the value of sea-surface temperature in the ocean model component (POP2) for a land point is undefined and set to 10^{35}. Therefore, we need a compression algorithm that can handle different types of data, and ideally one that can be either lossy and lossless and allows specification of error or compression rates. This flexibility is important because our intent is to customize the compression for each variable individually. Second, because compression will eventually be integrated into the I/O package of CESM, the software must be robust and sharable with a diverse user community. Third, achieving good compression ratios while maintaining the integrity of the reconstructed data is critical. Therefore, the compression software for CESM datasets should ideally possess the following attributes:

1. open source or freely available (i.e., no intellectual property restrictions);
2. permits lossy and lossless modes;
3. allows the specification of error or compression rate for a variable;
4. handles both 32- and 64-bit data;
5. does not require grid information; and
6. can accommodate special or missing values.

Items 2-6 are necessary due to the variety of data generated by CESM, and the first item is highly desirable because of the size of the CESM user community.

3.2 Algorithms

With the above selection criteria in mind, we chose several algorithms from those described in Section 2 to evaluate with CESM: *fpzip, ISABELA, APAX*, and *GRIB2* (with JPEG2000 compression). We briefly describe each of these algorithms in more detail and list some of their properties in Table 1. Note that all algorithms chosen have a lossy mode and do not require grid information, but none of the algorithms satisfy all of the desired requirements. Most do not accomodate special or missing values, but we hope that capability could be added.

3.2.1 fpzip

fpzip [14] can perform in both lossy and lossless mode, and in lossy mode the amount of compression achieved is affected by the number of least significant bits truncated. In particular, one can specify the number of bits of precision to retain, which must be a multiple of 8 (i.e., 8, 16, 24, or 32, the latter of which is lossless for single-precision data). Recent results on real physics applications are encouraging [12].

3.2.2 ISABELA

ISABELA [11] is aimed at compressing data that is potentially noisy and enabling random access to that compressed data. This method pre-sorts spatial data so that it is relatively smooth before applying a curve-fitting approximation, such as a B-spline or wavelet. A window size within which to sort as well as the desired per-point relative error must be specified by the user. (We use the recommended window size of 1024.) Because *ISABELA* is a local method, a subset of the data (instead of the entire dataset) can be decoded, which is potentially useful for post-processing analysis tasks. As expected for a wavelet method, it can be sensitive to data at the boundaries of a window, but the results shown in [11] are good.

3.2.3 GRIB2 with JPEG2000 compression

We selected *GRIB2* [5] for further investigation because of its wide acceptance in the meteorological community. Unfortunately, it is non-trivial to use as the compression parameters that control the bits of precision (and indirectly the compression rate) have to be customized for each variable according to the variable's magnitude, which will be discussed further in Section 5. Also, the encoding itself into

Table 1: Algorithm properties.

Method	lossless mode	special values	freely avail.	fixed quality	fixed CR	32- & 64-bit
GRIB2 + jpeg2000	N	Y	Y	N	N	N
APAX	Y[1]	N	N	Y	Y	Y
fpzip	Y	N	Y	N	N	Y
ISABELA	N	N	Y	N	N	Y

the *GRIB2* format is lossy, and, therefore, lossless is not an option even if one uses lossless JPEG2000 compression.

3.2.4 APAX

Because both [12] and [8] show good results for Samplify's *APAX* compressor, we were curious to apply it CESM data, despite its being a commercial product. Also, *APAX* is the only method that allows for the specification of fixed compression rates (with varying quality), which is very useful in practice. Another nice feature is the *APAX* profiler tool that illustrates the quality of the reconstructed data and recommends encoding rates. In fact, *APAX* is the only method where the user can specify a fixed quality mode (with varying compression rates). All three previous methods require the much more effort and tuning on the user's part to achieve a fixed quality.

4. VERIFICATION PROCESS

In this section, we describe metrics to characterize the original data, to quantify the difference (i.e., error) between the reconstructed and original data, and to evaluate the reconstructed data in the context of an ensemble of CESM runs with slight perturbations. In the following discussion, denote the original spatial dataset X as $X = \{x_1, x_2, \ldots, x_N\}$, with x_i a scalar, and the reconstructed dataset $\tilde{X}$ by $\tilde{X} = \{\tilde{x_1}, \tilde{x_2}, \ldots, \tilde{x_N}\}$. We denote the range of X by R_X (i.e., $R_X = x_{max} - x_{min}$). Note that although climate data is both spatial and temporal, we are considering a single temporal step for this preliminary analysis.

4.1 Characterizing the Original Data

First, characterizing the original data is important for gaining insight into what types of compression schemes will or will not be effective for a particular variable. We employ several standard metrics for characterizing X: the minimum (x_{min}) and maximum (x_{max}) values, the mean (μ_X), and the standard deviation (σ_X). We also measure how well lossless compression works on data from a particular variable. We use the lossless compression scheme that is part of the NetCDF-4 library (zlib) and to compress our original file, F_{orig}, which results in the losslessly compressed file F_{comp}. The compression ratio (CR) is defined as the ratio of the size of the compressed file to that of the original file (c.f. [9, 18]):

$$\mathrm{CR}(F) \;=\; \frac{\mathrm{filesize}(F_{comp})}{\mathrm{filesize}(F_{orig})}. \tag{1}$$

[1]Lossless mode is not supported for 64-bit data.

If the CR for the NetCDF-4 lossless compression for a particular variable is close to one, then lossless compression is not effective.

4.2 The Original and Reconstructed Data

Second, we characterize the difference between the original and reconstructed datasets via measures of pointwise error, average error, and correlation. The pointwise error at point i between the original and the reconstructed data is denoted by e_i, where $e_i = x_i - \tilde{x_i}$. The maximum absolute pointwise error, or maximum norm, is denoted by e_{max} and its magnitude indicates the minimum precision that has been achieved. Because our simulation data varies quite a bit in magnitude, depending on the variable, we define the normalized maximum pointwise error as

$$e_{nmax} \;=\; \frac{\max_{i=1:N} |e_i|}{R_X}, \tag{2}$$

which facilitates comparisons of error between variable types. We decide whether or not e_{nmax} is an acceptable size based on our ensemble results described in Section 4.3. To evaluate the average error in the reconstructed data, we evaluate the popular root mean squared error (RMSE):

$$\mathrm{rmse} \;=\; \sqrt{\frac{1}{N}\sum_{i=1}^{N}(e_i)^2}. \tag{3}$$

Again, due to the diversity of variables, we prefer the *normalized* RMSE (NRMSE) measure:

$$\mathrm{nrmse} \;=\; \frac{RMSE}{R_X}. \tag{4}$$

Looking at the pointwise error in combination with the average error for a large dataset is important as the average error could be quite small despite a relatively large error at one or more points. We note that another commonly used average error metric for evaluating compression schemes, particularly in visualization, is the peak signal-to-noise ratio (PSNR). The PSNR evaluates the size of the RMSE relative to the peak size of the signal (see. e.g., [15, 9]), but we do not report on the PSNR as it conveys the same type of error information as the NRMSE.

To evaluate the correlation between the values of a particular variable in the original and reconstructed datasets, we utilize the Pearson correlation coefficient (ρ):

$$\rho \;=\; \frac{\mathrm{cov}(X,\, \tilde{X})}{\sigma_X \sigma_{\tilde{X}}}, \tag{5}$$

where $\mathrm{cov}(X,\, \tilde{X})$ is the covariance. This coefficient indicates the strength of the linear relationship between the variables, where $\rho \in [-1\,, 1]$ and $\rho = 1$ indicates a perfect positive correlation. For context, the APAX profiler recommends that the correlation coefficient be .99999 (or better) between the original and reconstructed data. We currently use .99999 as the acceptance threshold for our tests.

4.3 Data in the Context of an Ensemble

Finally, we evaluate the reconstructed data in the context of the new CESM port-verification tool (CESM-PVT) [13], which we describe briefly in this section. The purpose of the CESM-PVT is to determine whether a change in CESM

that does *not* result in bit-for-bit agreement with the previous result is statistically distinguishable (i.e., is it "climate-changing" or not). The motivation for developing this tool was to create a straightforward way to verify the CESM code after porting it to a new machine architecture. Although running a CESM simulation on a new machine will not give the same bit-for-bit results as on the original "trusted" machine, the results should not be climate-changing. Previous to the development of the CESM-PVT, CESM was evaluated on a new architecture by comparing the results of a single 500-year simulation run on each of the two architectures. The previous methodology for comparing the two long runs was resource intensive and not entirely rigorous because of its use of subjective evaluations.

The CESM-PVT uses the following approach. First, an ensemble $E = \{E_1, E_2, \ldots, E_{101}\}$ consisting of 101 one-year climate simulations is run with annual averages of output for a selected grid resolution on a "trusted" machine. These 101 simulations differ only in a random perturbation of the initial atmospheric temperature condition of $\mathcal{O}(10^{-14})$. Such a perturbation should not be climate-changing over a one-year timeframe. Due to the nonlinear properties of this model, the trajectories of the ensemble members will rapidly diverge, but the statistical properties of the ensemble members are expected to be the same. Then, for each ensemble member m, the mean and standard deviation are calculated at every grid point x_i in the sub-ensemble $\{E \setminus m\}$ (consisting of the remaining 100 members) for each variable X and are denoted by $\bar{x}_i^{E\setminus m}$ and $\sigma_{x_i}^{E\setminus m}$, respectively. The Z-score that compares the value of x_i of ensemble member m (x_i^m) to sub-ensemble $\{E \setminus m\}$ is

$$Z_{x_i}^m = \frac{x_i^m - \bar{x}_i^{E\setminus m}}{\sigma_{x_i}^{E\setminus m}} \tag{6}$$

Therefore, the root mean squared Z-score for dataset X of ensemble m is given by

$$\mathrm{RMSZ}_X^m = \sqrt{\frac{1}{N_X}\sum_i (Z_{x_i}^m)^2}, \tag{7}$$

where N_X is the total number of grid points in X. This process, applied to each ensemble member in E, results in a distribution of 101 RMSZ scores for each output variable. In addition, global means of each of the variables are calculated.

The second step in the verification is to run a small number (generally three is sufficient) of randomly selected ensemble runs on the new architecture. The global means from the new runs are compared against the global mean of ensemble E in order to detect whether there has been a range shift (which would indicate a changed climate). In addition, we calculate the RMSZ scores for the variables in the new ensemble runs and check whether or not they fall within the z-score distribution from E.

If a variable's RMSZ score falls within the distribution, then that variable is considered to have "passed". A subject of future work is determining which variables are "critical" and *must* pass and which variables may be allowed to fall outside of the distribution (by some specified amount). At present, because the ensemble runs are one-year in length, only the output variables from the atmospheric model are evaluated, as the atmosphere model will be affected by feedback sooner than the ocean or ice models, for example.

One can also imagine that even on the benchmark machine, a non-bit-for-bit change in CESM could be the result of compiling the code with different compiler options (or a different compiler) [4], of making an incremental improvement to the code, or of redesigning an algorithm such that the order of operations is affected in parallel. It makes sense to eventually apply the CESM-PVT to these scenarios as well. For our purposes in evaluating the reconstructed data after compression, we employ the CESM-PVT in a slightly different way. We verify a compression utility by requiring that if we choose three members at random from ensemble E, denoted by index m, and apply the compression/decompression to each m, then the RMSZ score of the reconstructed result must at minimum fall within the distribution of the RMSZ values from the ensemble E, as with the CESM-PVT test. More stringently, though, we also require the difference between the original and reconstructed RMSZ score to be "small" as compared to the range of the ensemble RMSZ scores. For the variables we test in the next section, the range of RMSZ scores is $\mathcal{O}(1)$ (and is less than two, in most cases), and we require the difference between the two RMSZ values to be less that 1/10 for the reconstructed data to pass this test.

$$|RMSZ_X^m - RMSZ_{\tilde{X}}^m| \leq \frac{1}{10} \tag{8}$$

When the reconstructed RMSZ score satisfies the CESM-PVT requirements of lying within the distribution, this shows that the impact of the compression on the solution is on par with that of a bit perturbation on the initial conditions. However, when the difference between the original and reconstructed values satisfy (8), then this implies that the distribution itself is essentially unchanged (statistically indistinguishable) by replacing the original data with the reconstructed data.

Additionally, we use the CESM-PVT to evaluate whether the compression utility has added any bias to the climate data. First, we compress and decompress all 101 members of ensemble E, resulting in the new reconstructed ensemble $\tilde{E}$. Next, for each ensemble member m in $\tilde{E}$, we calculate the RMSZ score of the reconstructed ensemble in the same manner as before with equations (6) and (7), substituting $\tilde{E}$ for E. Then, for each variable, we compare the 101 RMSZ scores of ensembles $\tilde{E}$ and E via a simple linear regression plot. An accurate reconstruction will yield a strong linear relationship, and we can estimate the standard deviation for the fit. For an unbiased reconstruction, the fitted line would have a slope of 1 and an intercept of 0, otherwise we have introduced bias. Therefore, a simple scatterplot of slope versus intercept that contains each compression method with its 95% confidence region (indicated by a rectangle) allows us to evaluate which methods have introduced bias. This plot will be clarified in the next section with data for several variables (Figure 4). The degree of uncertainty for each compression method is quite important. For example, if the line of best fit has a slope of (nearly) one and small uncertainty, but a non-zero intercept, then bias has been introduced uniformly, and this will be detected by the RMSZ ensemble test. On the other hand, if the uncertainty is relatively large, then even if the slope is close to one, the RMSZ ensemble test may not have caught the bias error (depending on the random samples chosen). Therefore we evaluate the distance between the ideal and worst case values for the slope, s_I

and s_{WC}, respectively, based on the 95% confidence region. We require that this distance be less than .05 for the method to be acceptable for compression:

$$|s_I - s_{WC}| \leq .05. \quad (9)$$

This criteria may be stricter than necessary, and we plan to explore the detection of bias further in subsequent work.

Finally, to evaluate whether the normalized maximum pointwise error, e_{nmax}, between the original and reconstructed data is reasonable, we extend the CESM-PVT to include a distribution of the maximum pointwise error. To do this, we follow a procedure similar to that for building the distribution of RMSZ scores. In particular, for each ensemble member m, at each grid point x_i, we calculate the maximum pointwise error between ensemble m and that grid point in all members of the sub-ensemble $\{E \setminus m\}$. The maximum pointwise error for member m is then the maximum of the maximum pointwise error over all grid points. Therefore, for the variable in dataset X of ensemble m, the normalized maximum pointwise error is given by

$$E^{mX}_{nmax} = \frac{\max_i(\max_{n \in E \setminus m} |x^m_i - x^n_i|)}{R^m_X}, \quad (10)$$

where R^m_X indicates the range of X for ensemble member m. Following this procedure for all 101 ensemble members creates a distribution that then can be used to determine whether or not the value of e_{nmax} (between the original and reconstructed data) falls within the variability of the ensemble. The acceptance criteria for this test is similar to that for the RMSZ ensemble test. We choose three ensemble members at random, and at minimum, the value of e_{nmax} for those must certainly be smaller than the range between the maximum and minimum values of E^{mX}_{nmax}, which is noted bu $R_{E^X_{nmax}}$. However, we additionally require that the maximum pointwise error between the original and reconstructed datasets to be an order of magnitude less than that range so that replacing the original data with the reconstructed data would have little effect on the ensemble distribution:

$$\frac{e_{nmax}}{R_{E^X_{nmax}}} \leq \frac{1}{10}. \quad (11)$$

Note that we are careful not to include any special values (such as the 10^{35} mentioned in Section 3.1) when calculating our metrics.

5. EXPERIMENTAL RESULTS

In this section, we explain the experiments that we performed on various datasets from CESM. We detail the results and apply the metrics outlined in Section 4.

5.1 Preliminaries

The results in this work were obtained from the 1.1 release version of CESM, using an active CAM5 atmosphere and CLM land model. We concentrate on the CAM history files, which contain a total of 83 two-dimensional and 87 three-dimensional variables, that are currently supported by the CESM-PVT. This spectral-element version of CAM uses a $ne = 30$ resolution, which corresponds to a 1-degree global grid, containing a total of 48,602 horizontal grid-points and 30 vertical levels.

We apply the four lossy methods described in Section 3.2 to the 170 CAM variables. For *fpzip*, we use two different levels of precision: 16- and 24-bit precision, which we denote as fpzip-16 and fpzip-24. We apply the B-spline variant of *ISABELA* with three different per-point relative error values: 1.0, 0.5, 0.1. The three options are called ISA-1.0, ISA-0.5, and ISA-0.1, respectively. As noted previously, applying *GRIB2* with JPEG2000 compression requires variable-level customization to achieve reasonable results due to the diversity (e.g., magnitude and range) of variables in CAM, and, therefore, we only show one result, which we denote by GRIB2. Finally, we evaluate the *APAX* compressor using the fixed compression rates 2, 4 and 5, which we refer to as APAX-2, APAX-4, and APAX-5, respectively.

Table 2: Characteristics of the datasets for variables U, FSDSC, Z3, and CCN3.

Variable	units	x_{min}	x_{max}	μ_X	σ_X	CR
U	m/s	-2.56e1	5.45e1	6.39e0	1.22e1	.75
FSDSC	W/m^2	1.24e2	3.26e2	2.43e2	4.83e1	.66
Z3	m	4.12e1	3.77e4	1.12e4	1.01e4	.58
CCN3	$\#/cm3$	3.37e-5	1.24e3	2.66e1	5.57e1	.71

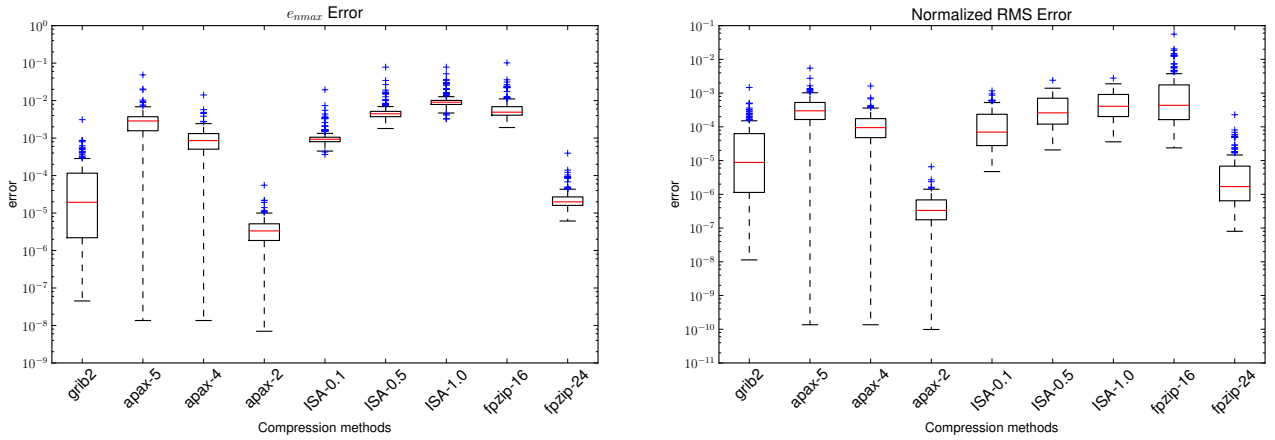
Datasets from several CAM variables will be presented in more detail: geopotential height above sea level (Z3), clear sky downwelling solar flux at surface (FSDSC), cloud condensation nuclei concentration at S=0.1% (CCN3), and zonal wind (U). Note that FSDSC is a 2D field and the rest are 3D. We selected these variables to represent the differing effects of compression on climate variables, and the characteristics described in Section 4.1 are given in Table 2. Note that "CR" in the table indicates the compression ratio for the lossless method from NetCDF-4, and the "units" column indicates the scientific units for the variable.

5.2 The Original and Reconstructed Data

We first compare the original and reconstructed data as described in Section 4.2. The two box plots in Figure 1 demonstrate how the datasets of all 170 variables respond to the four compression methods and the variants within each method. In the left plot, the y-axis indicates the normalized maximum pointwise error between the original and reconstructed data, and in the right plot, the y-axis is the NRMSE, as given in (4). The x-axis lists the compression methods evaluated, with the rectangle in each column indicating the range of the lower to upper quartiles and the red line denoting the median. The "whiskers" extending from the top and bottom of the rectangles denote the full range

Table 3: NRMS errors (and compression ratio CR) between the original and reconstructed datasets for U, FSDSC, Z3, and CCN3.

Comp. Method	U	FSDSC	Z3	CCN3
		NRMSE (CR)		
GRIB2	3.6e-4 (.10)	1.4e-4 (.22)	7.8e-8 (.32)	2.3e-8 (.37)
APAX-2	5.8e-7 (.50)	8.3e-7 (.50)	7.0e-8 (.50)	1.6e-7 (.50)
APAX-4	1.4e-4 (.25)	2.1e-4 (.26)	2.0e-5 (.25)	4.1e-5 (.25)
APAX-5	4.3e-4 (.20)	5.4e-4 (.21)	5.1e-5 (.19)	9.9e-5 (.20)
fpzip-24	2.2e-6 (.39)	1.8e-5 (.34)	5.1e-6 (.19)	6.5e-7 (.36)
fpzip-16	5.7e-4 (.15)	4.6e-3 (.10)	1.2e-3 (.04)	1.7e-4 (.12)
ISA-0.1	8.7e-5 (.57)	4.1e-4 (.37)	3.8e-5 (.39)	2.8e-5 (.37)
ISA-0.5	2.7e-4 (.44)	9.1e-4 (.36)	9.8e-5 (.37)	1.2e-4 (.38)
ISA-1.0	3.7e-4 (.41)	1.1e-3 (.36)	1.5e-4 (.36)	2.0e-4 (.37)



(a) Normalized maximum pointwise error.

(b) Normalized RMSE.

Figure 1: Normalized maximum pointwise and normalized RMS errors for all 170 variable datasets.

Table 4: Maximum relative pointwise errors (and compression ratio) between the original and reconstructed datasets for U, FSDSC, Z3, and CCN3.

Comp. Method	U	FSDSC	Z3	CCN3
		e_{nmax} (CR)		
GRIB2	6.2e-4 (.10)	2.5e-4 (.22)	1.6e-7 (.32)	4.9e-8 (.37)
APAX-2	3.3e-6 (.50)	4.7e-6 (.50)	3.3e-6 (.50)	2.9e-6 (.50)
APAX-4	9.0e-4 (.25)	1.1e-3 (.26)	8.3e-4 (.25)	7.5e-4 (.25)
APAX-5	2.7e-3 (.20)	2.7e-3 (.21)	3.1e-3 (.19)	1.9e-3 (.20)
fpzip-24	1.2e-5 (.39)	3.9e-5 (.34)	3.3e-6 (.19)	2.4e-5 (.36)
fpzip-16	3.1e-3 (.15)	9.9e-3 (.10)	6.8e-3 (.04)	5.3e-3 (.12)
ISA-0.1	6.4e-4 (.57)	1.6e-3 (.37)	9.8e-4 (.39)	8.7e-4 (.37)
ISA-0.5	2.9e-3 (.44)	7.6e-3 (.36)	4.9e-3 (.37)	3.9e-3 (.38)
ISA-1.0	4.9e-3 (.41)	1.5e-2 (.36)	9.9e-3 (.36)	7.9e-3 (.37)

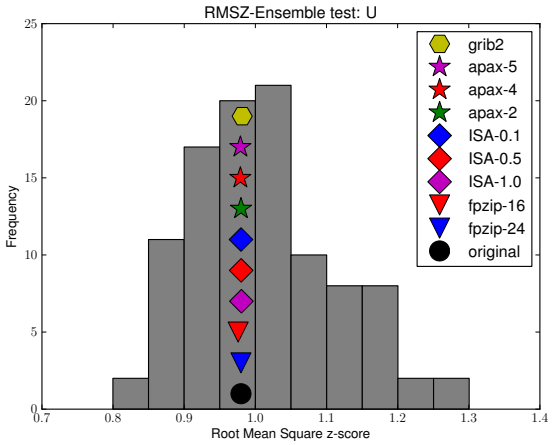
of the distribution (i.e., the maximum and minimum). As expected, these two error metrics vary quite a lot between the 170 variables. For example, if we consider the APAX-4 method, the NRMSE ranges from $\mathcal{O}(10^{-3})$ to $\mathcal{O}(10^{-10})$. Also, the methods with higher levels of compression clearly result in higher errors, as expected. These plots suggest that to achieve a fixed quality on our diverse set of CESM variables, treating the variables individually in terms of choosing a compression method is required. (We note that because *APAX* offers both a profiler and a fixed quality mode, this task is considerably simpler for *APAX* than for the other methods.) In fact, some variables may need to be compressed with a lossless variant to achieve acceptable quality, but we do not include lossless results in this figure as *ISABELA* and *GRIB2* cannot be run losslessly.

Now we take a closer look at the datasets for variables U, Z3, FSDSC, and CCN3 to better illustrate the effects of the various compression methods. Tables 3 and 4 show the average and pointwise error metrics, NRMS and e_{nmax}, respectively, between the original and reconstructed datasets, as well as the compression ratio. The effectiveness of the different compression techniques on the datasets for a particular variable varies quite a bit, as does the effectiveness of a single method across variable datasets. For example, *ISABELA* is clearly not doing as well as the other methods in terms of both the average and pointwise errors, reflecting perhaps on *ISABELA*'s focus on enabling random access. Also the difference between the three *ISABELA* variants is small for all four variables because the amount of storage needed for the sort index is a higher percentage of the total for single-precision data (we would expect *ISABELA* to obtain better compression ratios on double-precision data). Note that the *APAX* methods are operating at a fixed compression rate, which is convenient in practice. Z3 generally compresses the most (i.e., has the smallest compression ratios), as with the lossless compression in Table 2. The method providing the lowest compression ratios overall is fpzip-16, but the errors are also the largest. GRIB2 performs well in terms of obtaining good compression ratios and small errors. For example, comparing GRIB2 with fpzip-24 on CCN3 indicates that, for a similar CR, the GRIB2 errors are about three orders of magnitude smaller. Similarly for U, GRIB2 yields smaller errors and a smaller CR than fpzip-16. Finally we note that for these variables, the NRMSE and e_{nmax} roughly correlate; the NRMSE tends to be an order of magnitude smaller.

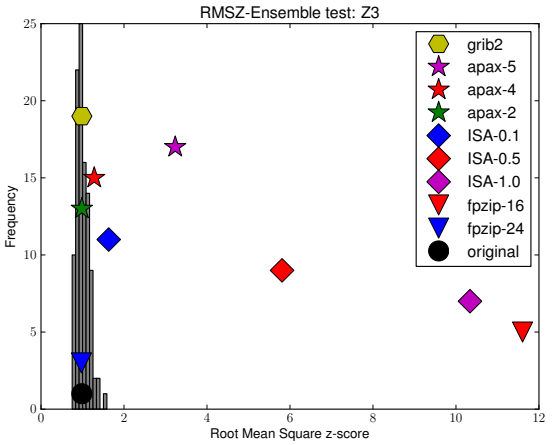
While the focus of this manuscript is on the quality of the compressed data, performance of the methods cannot be ignored, particularly given the volume of climate data that will be compressed. Table 5 lists the time to compress and reconstruct the datasets for variables U and FSDSC, as well as the achieved compression ratio, for all the methods. The (*) by some of the CR values for variable FSDSC indicates that the reconstructed data from that method was not of sufficient quality to pass test metrics. The time to compress and reconstruct our data depends on the variable, and variable U took more time to compress than FSDSC. The *APAX* method is clearly the fastest of the methods, sometimes by a couple orders of magnitude.

5.3 Comparing to an Ensemble

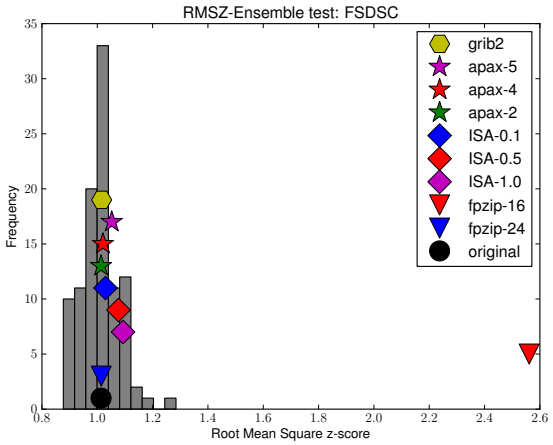
By simply comparing the original data to the reconstructed data with standard metrics, as done in Section 5.2, it is difficult to evaluate the effectiveness of the methods relative to each other (as well as the quality of the reconstructed data). Therefore, we now look at the reconstructed data in the context of the CESM-PVT ensemble, as described in Section 4.3. First, the four plots in Figure 2, one for each variable, show the distribution of the RMSZ scores for the 101



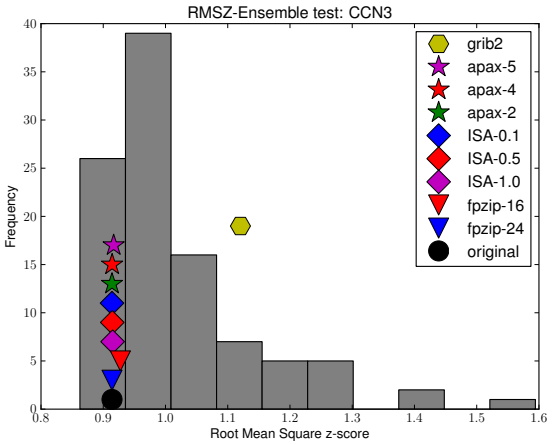
(a) Variable U (zonal wind).



(b) Variable Z3 (geopotential height above sea level).



(c) Variable FSDSC (clear sky downwelling solar flux at surface).



(d) Variable CCN3 (CCN concentration at S=0.1%).

Figure 2: Ensemble plots for variables U, Z3, FSDSC, and CNN3 with results off all data compression methods indicated.

Table 5: Compression and reconstruction timings (in seconds) and compression ratios (CR) for variables U (3D) and FSDSC (2D).

Comp.	U			FSDSC		
Method	comp.	reconst.	CR	comp.	reconst.	CR
GRIB2	0.284	0.264	0.10	0.020	0.017	0.22
APAX-2	0.068	0.044	0.50	0.002	0.001	0.50
APAX-4	0.062	0.042	0.25	0.002	0.001	0.26
APAX-5	0.059	0.042	0.20	0.002	0.001	0.21(*)
fpzip-24	0.123	0.114	0.39	0.004	0.004	0.34
fpzip-16	0.101	0.097	0.15	0.003	0.003	0.10(*)
ISA-0.1	4.016	0.508	0.57	0.053	0.016	0.39
ISA-0.5	2.455	0.530	0.44	0.042	0.016	0.36(*)
ISA-1.0	1.852	0.478	0.41	0.042	0.016	0.35(*)

ensemble runs. The different markers indicate the RMSZ scores from one of the reconstructed members, whose original RMSZ score is indicated by the black circle. Note that the y-axis value (frequency) is not relavant for the markers; they are stacked for aesthetic reasons. Recall that the goal of using the CESM-PVT tests is to ensure that the reconstructed data falls not only within the original distribution, but is also quite close, e.g. equation (8), to the original value. Theses plots show that in terms of obtaining a similar RMSZ value with the reconstructed data, all compression methods do well for the variable U and most do well for FSDSC. These two variables both have a small range, and, because U represents wind speed, it typically varies quite smoothly. The *ISABELA* methods and fpzip-16 are the worst performers for FSDSC. Interestingly, from the NRMSE result in Table 3, the results for *APAX* and *ISABELA* for FDSDC are similar, but Figure 2 shows that the ISA-0.5 and ISA-1.0 RMSZ scores are much further away from the original for FSDSC. Also fpzip-16 has a similar NRMSE to that of ISA-1.0, but this plot shows that it does very poorly in the ensemble test. All of the methods perform the worst on variable Z3,

despite its obtaining the lowest compression ratios in Tables 3 and 4. The dataset for this variable has quite a large standard deviation (Table 2), and though the lossless method obtained the best CR, the lossy methods have difficulties. Finally, all methods do reasonably well with CCN3, except for GRIB2. CCN3 has quite a large range, and we find that GRIB2 does not perform well on such variables in terms of the RMSZ score of the reconstructed data, something which is not apparent for GRIB2 and CCN3 from Tables 3 and 4.

Next, we evaluate e_{nmax} in the context of the CESM-PVT ensemble, as described in Section 4.3, to determine whether the normalized maximum pointwise error between the original and reconstructed data (Table 4) is acceptable. Figure 3 displays four box plots, one for each variable. In each box plot, the y-axis indicates the value of e_{nmax} (note that the range shown for the y-axis varies for each plot). The leftmost column on the x-axis shows the distribution of the ensemble as calculated by equation (10). The rectangle indicates the range of the lower to upper quartiles, with the red line denoting the median. The whiskers extending from the top and bottom of the rectangle denote the full range of the distribution. The remaining columns represent the values of e_{nmax}, as calculated by equation (2), between the original ensemble member and the reconstructed data for each compression method. The plots in this figure show that all methods do quite well on the dataset for variable U in terms of the pointwise error. For FSDSC, the *ISABELA* methods show some larger errors, and several methods have some difficulty with Z3. For CCN3, we again see that GRIB2 does much worse than the other methods.

Finally, we use the ensemble to evaluate whether the compression process has introduced bias into the reconstructed datasets. The four plots in Figure 4 show the slope-versus-intercept data described in Section 4.3. A 95% confidence region is shown for each compression method. Note that both the y-axis and x-axis ranges are quite different for each subplot in the figure. Also, the sizes of the uncertainty regions vary quite a lot between the compression methods for a particular variable. A larger uncertainty in the estimation implies that all ensemble members are responding differently to the compression process. We have chosen 95% as a useful level of confidence and are evaluating the slope based on Equation 9. For U, it appears that most of the compression methods are introducing bias because the rectangles do not contain (1,0). However, the scale of the x-axis shows that the amount of bias added is so small that it is insignificant to us. FSDSC is similar in that we only see one rectangle containing the origin, but four of the other methods are acceptable because their uncertainty is within our tolerance. For CCN3, *GRIB2* does much worse that the other methods and is not shown on the plot (its slope ranges from .93 to .97). Variable Z3 has some significant outliers as well.

5.4 Customizing by Variable

We now explore customizing each of the four compression methods by variable to perform optimally in terms of quality and CR on each variable's dataset. To begin, Table 6 summarizes the results of the compression methods on the datasets for all the variables with our four metric tests. For each method, we indicate how many variables (out of 170 total) "passed" each of the tests, according to the criteria defined in the previous section. The second column, labeled "ρ", is the Pearson correlation coefficient value test.

Table 6: Number of passes for all compression methods on 170 variables.

Comp. Method	Number of passes ρ	RMSZ ens.	E_{rmax} ens.	bias	all
GRIB2	167	163	170	124	121
APAX-2	170	170	170	146	146
APAX-4	167	163	165	126	122
APAX-5	130	152	160	111	85
fpzip-24	170	164	170	167	163
fpzip-16	122	129	138	126	113
ISA-0.1	168	160	164	160	152
ISA-0.5	140	154	145	161	123
ISA-1.0	63	154	112	161	43

Table 7: Results from customizing each compression method by variable and forming a hybrid method.

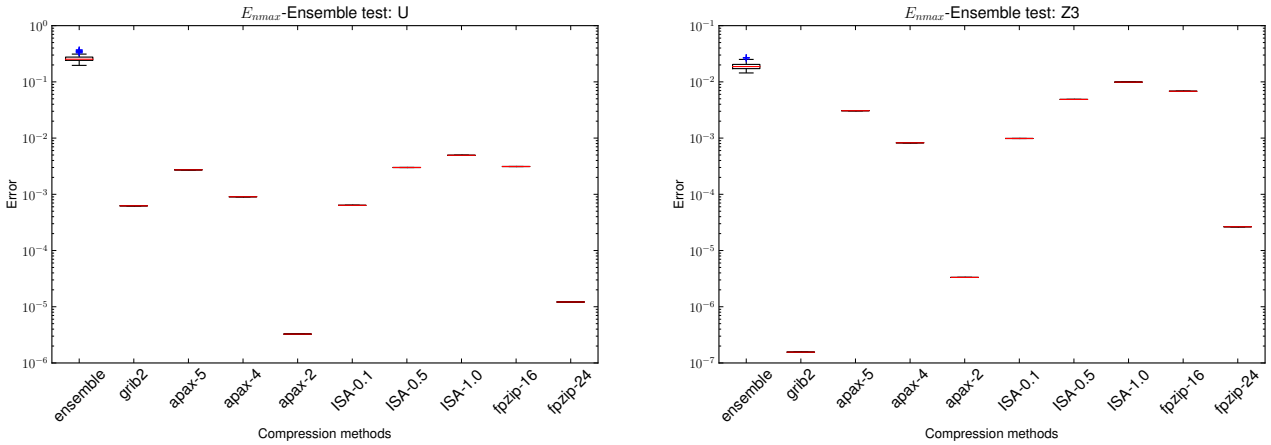
	GRIB2	*ISABELA*	*fpzip*	*APAX*	NC
avg. CR	0.37	0.42	0.18	0.29	0.61
best CR	0.03	0.20	0.02	0.06	0.07
worst CR	0.86	0.77	0.68	0.80	0.86
avg. ρ	.9999999	.9999991	.9999995	.9999991	1.0
avg. nrmse	5.73e-5	3.22e-4	2.35e-4	2.61e-4	0.0
avg. e_{nmax}	1.01e-4	5.56e-3	2.76e-3	1.83e-3	0.0

The third column, "RMSZ ens.", contains the results of the RMSZ ensemble test, and the fourth and fifth columns are the E_{nmax} ensemble and bias tests, respectively. The rightmost column indicates the number of variables that pass all four tests in columns 2-5. As expected, methods with higher compression rates generally have fewer test passes. Because none of the methods has perfect passing rate, we will need to use a lossless method on some of the variables.

Based on the per-variable test results for the compression methods, we now construct the best "hybrid" option for each of our four methods. In particular, we choose the variant of each method (i.e., level of compression) for each variable that yields the best CR and passes all of our tests, choosing a lossless variant if necessary. For example, if fpzip-16 is not of acceptable quality for variable Z3 (i.e., does not pass all four tests in Table 6), then we evaluate fpzip-24. If fpzip-24 is not acceptable, then we use fpzip's lossless compression option (fpzip-32). We expect this selection process to be more sophisticated in the future. Note that because *ISABELA* and

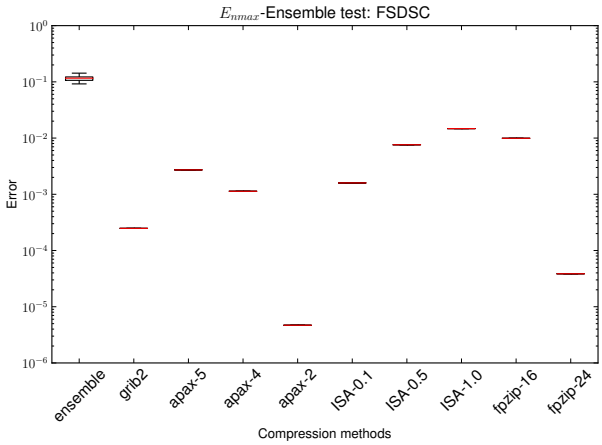
Table 8: Number of variables (out of 170 total) that each variant of each compression method uses to form the hybrid method results in Table 7.

Method	Variant	Number of Variables
GRIB2	GRIB2	121
	NetCDF-4	49
ISABELA	ISA-1.0	43
	ISA-0.5	80
	ISA-0.1	29
	NetCDF-4	18
fpzip	fpzip-16	113
	fpzip-24	50
	fpzip-32	7
APAX	APAX-5	85
	APAX-4	37
	APAX-2	24
	NetCDF-4	24

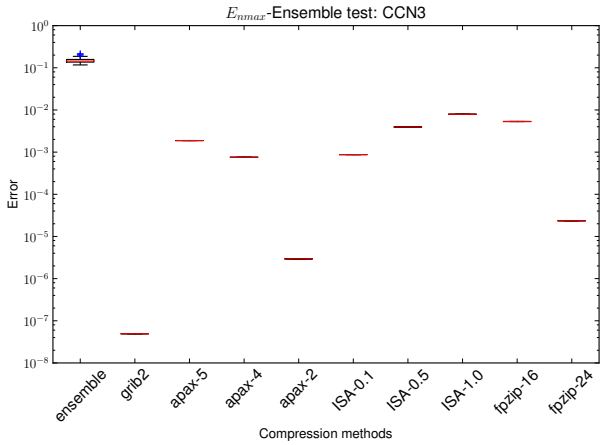


(a) Variable U

(b) Variable Z3



(c) Variable FSDSC



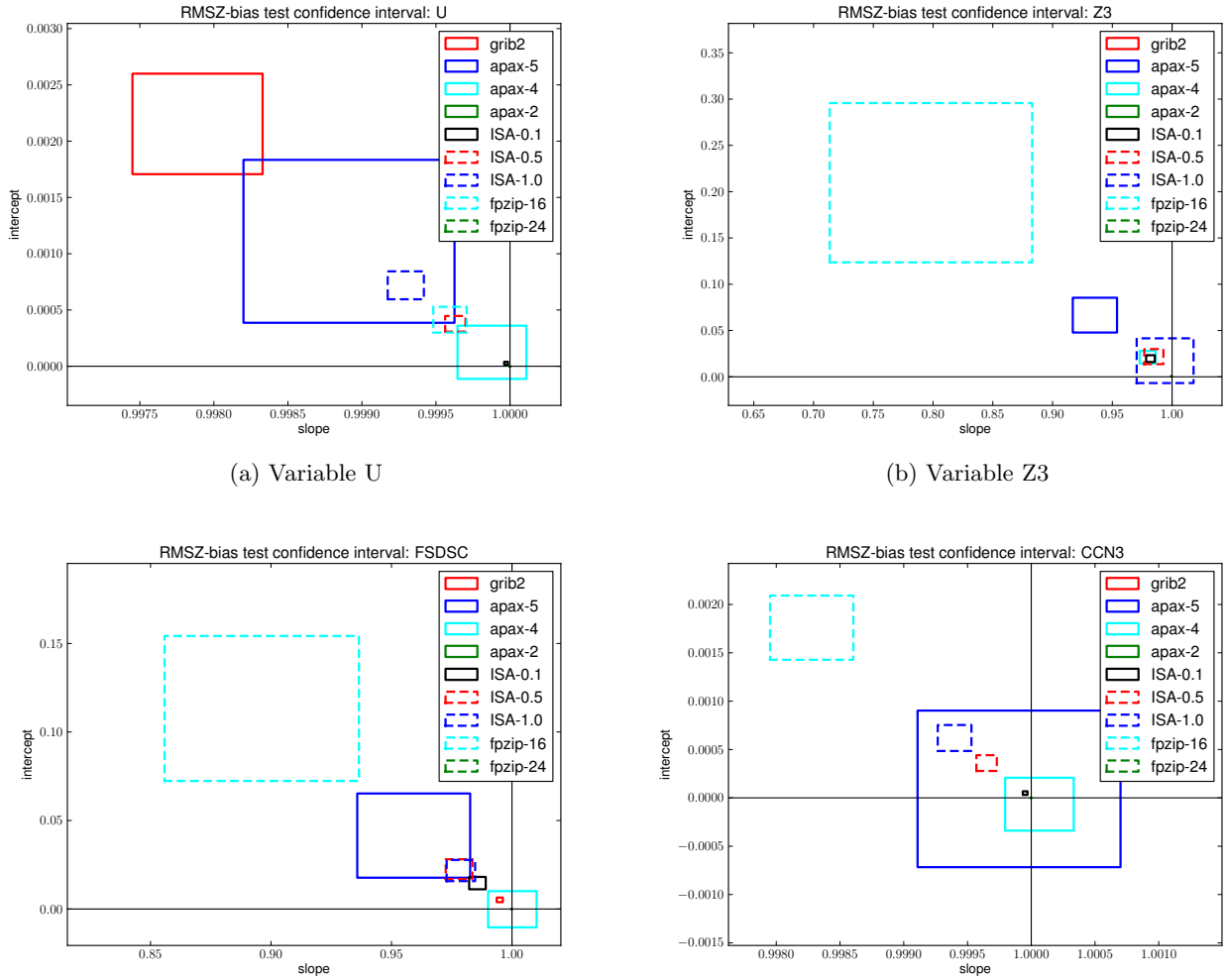
(d) Variable CCN3

Figure 3: Ensemble plot for E_{nmax} with all data compression methods indicated.

GRIB2 cannot be lossless, we use NetCDF4 compression for any variable that requires lossless treatment. The results of this customization for each of the four methods are listed in Table 7, which gives statistics that compare the original and reconstructed datasets. For comparison, the right-most column, labeled "NC", indicates lossless NetCDF4 compression on all variables. The first three rows list the average, best, and worst compression ratios, respectively, over all 170 variables. The fourth, fifth, and sixth rows list the average Pearson correlation coefficient, nrmse, and e_{nmax} values, respectively. To illustrate the composition of the four hybrid methods in Table 7, Table 8 indicates the number of variables used by each variant of each method (summing to 170).

Note that because *GRIB2* requires a compression parameter for each variable that indicates bits of precision, we have essentially already undergone this customization process for *GRIB2* in the results shown. (Though at this point, we do select lossless compression if needed.) When we initially ran *GRIB2*, we selected the same decimal scale factor (D), which is used to achieve desired precision, for each variable, and our initial results for *GRIB2* were quite poor. The results did improve by specifying a D for each variable that depended on the magnitude (and range) of that variable. However, we were only able to achieve the more competitive results presented here for *GRIB2* by using the RMSZ ensemble test as a guide for choosing an optimal D.

The hybrid method results in Table 7, particularly the average compression ratio, indicate that *fpzip*, followed by *APAX*, performed the best on this climate data while maintaining an acceptable level of quality. In choosing a method to continue forward, we recall Table 1 and consider several factors. Both *APAX* and *fpzip* are lacking support for special values, but we assume that could be either easily incorporated into the algorithm or handled through our pre- and post-processing. It is noteworthy that the method with the best compression rate, *fpzip*, is also freely available. However, we also note that the fastest method, *APAX*, has some attractive features that are quite useful in practice, such as fixed compression mode, fixed quality mode, and the inclusion of a profiler. Also, we have not yet tried fixed compression rates 6 and 7 for *APAX*, which, given that 85 variables were able to use APAX-5, may lower the average CR for *APAX*. The biggest drawback for *APAX* is the fact that it's a commercial product. On the other hand, the biggest draw-



(a) Variable U

(b) Variable Z3

(c) Variable FSDSC

(d) Variable CCN3

Figure 4: Bias plots for variables U, Z3, FSDSC, and CCN3 with all data compression methods indicated.

back for *fpzip* is the restriction that the precision specification be a multiple of 8, leaving less room for customization.

6. CONCLUDING REMARKS

The disparity between the rapidly decreasing cost to compute floating-point operations and the more slowly decreasing cost of data storage is problematic for the data-intensive earth system modeling community. The rising relative cost of storing the output data from climate simulations can no longer be ignored, and the length and size of certain large-scale climate simulations are already being constrained by disk storage limitations. The use of lossy data compression could enable the simulation of longer time periods or more frequent output for a fixed on-line storage cost. This paper presents a preliminary, but thorough, study of several state-of-the-art compression techniques on data from CESM. These compression techniques are evaluated based on the development of a more comprehensive set of verification metrics than are used in a typical data compression study. Our metrics serve to ensure that the reconstructed data is indistinguishable from the natural variability of the system. We show that individually addressing the compression needs of each climate variable's dataset is a necessity, given the diversity of data in a climate model, and our preliminary effort at this achieved compression rates of up to 5:1. While decisions about "correctness" are specific to an application domain, our methodology is certainly applicable to other domain scientists working with model simulation data.

In subsequent work, we will fine tune the variable-by-variable customization needed to obtain maximum compression while ensuring that our reconstructed data is virtually indistinguishable from the original in terms of the post-processing data analysis that occurs. This customization will be investigated in close collaboration with climate scientists who can provide valuable insight into variable attributes and analysis. We plan to extend our verification metrics to evaluate the impact of compression on global energy budget calculations as well as on field gradients. In addition, because climate scientists visualize subsets of their simulation data as part of the post-processing analysis workflow, it is important that the reconstructed data produces quality images. We intend to utilize the structural similarity (SSIM) index [19], a recent and meaningful metric of image

quality, as it relates to human perception. Finally, exploring different grid resolutions, particularly finer ones, is critical.

7. ACKNOWLEDGMENTS

Special thanks to Steve Sullivan and Kevin Paul.

8. ADDITIONAL AUTHORS

Jim Edwards (NCAR, email::jedwards@ucar.edu) and Mariana Vertenstein (NCAR, email::mvertens@ucar.edu) and Al Wegener (Samplify, email:awegener@samplify.com) .

9. REFERENCES

[1] T. Bicer, J. Yin, D. Chiu, G. Agrawal, and K. Schuchardt. Integrating online compression to accelerate large-scale data analytics applications. *Parallel and Distributed Processing Symposium, International*, 0:1205–1216, 2013.

[2] M. Burtscher and P. Ratanaworabhan. High throughput compression of double-precision floating-point data. In *Data Compression Conference*, pages 293–302, 2007.

[3] M. Burtscher and P. Ratanaworabhan. FPC: A high-speed compressor for double-precision floating-point data. In *IEEE Transactions on Computers*, volume 58, pages 18–31, January 2009.

[4] M. J. Corden and D. Kreitzer. Consistency of floating-point results using the Intel® compiler or Why doesn't my application always give the same answer? Technical report, Software Solutions Group, Intel Corporation, 2012.

[5] C. F. Day, C. Sanders, J. Clochard, J. Hennessy, and S. Elliott. Guide to the WMO table driven code form used for the representation and exchange of regularly spaced data in binary form, November 2007. http://www.wmo.int/pages/prog/www/WMOCodes/Guides/GRIB/GRIB2_062006.pdf.

[6] L. A. B. Gomez and F. Cappello. Improving floating point compression through binary masks. In *IEEE BigData*, Santa Barbara, CA, 2013.

[7] N. Hübbe and J. Kunkel. *Computer Science - Research and Development*, chapter Reducing the HPC-datastorage footprint with MAFISC - multidimensional adaptive filtering improved scientfic data compression. Springer, Hamburg, Berlin, Heidelberg, 2012.

[8] N. Hübbe, A. Wegener, J. M. Kunkel, Y. Ling, and T. Ludwig. Evaluating lossy compression on climate data. In *Proceedings of the International Supercomputing Conference (ISC '13)*, pages 343–356, 2013.

[9] J. Iverson, C. Kamath, and G. Karypis. Fast and effective lossy compression algorithms for scientific datasets. In *Proceedings of the 18th International Conference on Parallel Processing*, Euro-Par'12, pages 843–856, Berlin, Heidelberg, 2012.

[10] K.E.Taylor, R. Stouffer, and G. Meehl. An overview of CMIP5 and the experiment design. *Bulletin of the American Meteorogical Society*, 93:485–498, 2012.

[11] S. Lakshminarasimhan, N. Shah, S. Ethier, S. Klasky, R. Latham, R. Ross, and N. F. Samatova. Compressing the incompressible with ISABELA: In-situ reduction of spatio-temporal data. In *Proceedings of the 17th International cCnference on Parallel Processing*, Euro-Par'11, Bordeaux, France, Aug 29 - Sep 2 2011.

[12] D. Laney, S. Langer, C. Weber, P. Lindstrom, and A. Wegener. Assessing the effects of data compression in simulations using physically motivated metrics. In *Supercomputing 2013 (SC'13)*, 2013.

[13] M. Levy, A. H. Baker, J. Anderson, J. M. Dennis, J. Edwards, A. Mai, D. Nychka, J. Tribbia, S. Vadlamani, M. Vertenstein, D. Williamson, and H. Xu. A new verification tool for the Community Earth System Model. *In preparation*, 2014.

[14] P. Lindstrom and M. Isenburg. Fast and efficient compression of floating-point data. *IEEE Transactions on Visualization and Computer Graphics*, 12:1245–1250, 2006.

[15] K. Sayood. *Introduction to Data Compression*. Morgan Kaufmann, fourth edition, 2012.

[16] E. R. Schendel, Y. Jin, N. Sha, J. Chen, C. Chang, S.-H. Ku, S. Ethier, S. Klasky, R. Latham, R. Ross, and N. F. Samatova. ISOBAR preconditioner for effective and high-throughput lossless data compression. In *IEEE 28th International Conference on Data Engineering (ICDE)*, 2012.

[17] J. Small, J. Bacmeister, D. Bailey, A. H. Baker, F. Bryan, J. Caron, J. Dennis, E. Munoz, J. Edwards, M. Holland, D. Lawrence, A. Mai, T. Scheitlin, B. Tomas, J. Tribbia, M. Vertenstein, and Y. Tseng. A new high-resolution global climate simulation using Community Atmosphere Model version 5 and an eddy-resolving ocean model. *In preparation*, 2014.

[18] S. Sullivan. Wavelet compression for floating point data - Sengcom. Technical report, University Corporation for Atmospheric Research, 2012. http://www.unidata.ucar.edu/software/netcdf/papers/sengcom.pdf.

[19] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[20] A. Wegener. Adaptive compression and decompression of bandlimited signals. US Patent 7009533, March 2006. http://www.patentlens/patentlens/patent/US_7009533.

[21] J. Woodring, S. M. Mniszewski, C. M. Brislawn, D. E. DeMarle, and J. P. Ahrens. Revisting wavelet compression for large-scale climate data using JPEG2000 and ensuring data precision. In D. Rogers and C. T. Silva, editors, *IEEE Symposium on Large Data Analysis and Visualization (LDAV)*, pages 31–38. IEEE, 2011.