Impact of Lossy Compression Errors on Passive Seismic Data Analyses

Abdul Hafiz S. Issah[*1] and Eileen R. Martin[1,2]

Abstract

New technologies such as low-cost nodes and distributed acoustic sensing (DAS) are making it easier to continuously collect broadband, high-density seismic monitoring data. To reduce the time to move data from the field to computing centers, reduce archival requirements, and speed up interactive data analysis and visualization, we are motivated to investigate the use of lossy compression on passive seismic array data. In particular, there is a need to not only just quantify the errors in the raw data but also the characteristics of the spectra of these errors and the extent to which these errors propagate into results such as detectability and arrival-time picks of microseismic events. We compare three types of lossy compression: sparse thresholded wavelet compression, zfp compression, and low-rank singular value decomposition compression. We apply these techniques to compare compression schemes on two publicly available datasets: an urban dark fiber DAS experiment and a surface DAS array above a geothermal field. We find that depending on the level of compression needed and the importance of preserving large versus small seismic events, different compression schemes are preferable.

Cite this article as Issah, A. H. S., and E. R. Martin (2024). Impact of Lossy Compression Errors on Passive Seismic Data Analyses, Seismol. Res. Lett. 95, 1675–1686, doi: 10.1785/0220230314.

Supplemental Material

Introduction

Innovations in seismic instrumentation have given rise to a variety of ways of gathering data. Two major innovations are the rise of low-cost seismic nodes and fiber-optic distributed acoustic sensing (DAS), both of which enable the recording of high-frequency data on many closely spaced sensors. For example, the Penn State Fiber-Optic foR Environment SEnsEing (FORESEE) project recorded urban environmental seismic data at 500 samples per second across 4189 m of fiber-optic cable at 2 m spacing (Zhu et al., 2021; Spica et al., 2023) and the PoroTomo Natural Laboratory at Brady's Hot Springs shared geothermal production and microseismicity data at 1000 samples per second across 8.9 km of cable at 1.021 m spacing (Coleman, 2016; Feigl et al., 2016). A sample of just nine DAS experiments (including these two) produced more than 750 TB of DAS data from 2015 to 2020, and this substantially faster rate of data acquisition (in comparison with long-period seismometers or nodal arrays) has inhibited geoscientists' ability to quickly access, analyze, and visualize these new data sources (Lindsey and Martin, 2021).

One solution to reduce the volume of seismic data is compression, with higher compression ratios commonly achieved through the use of lossy compression techniques. However, lossy compression techniques introduce data error, so we need to quantitatively compare the various options for reducing data storage and data movement during passive seismic processing. Compression of seismic data can be achieved by the transformation of data into a sparse representation so that fewer bytes are needed to capture key features of the data. Efforts in lossy compression of seismic data have been skewed toward active seismic data and in the past have often focused on wavelet decomposition and the discrete cosine transform (Bosman and Reiter, 1993; Donoho et al., 1999; Averbuch et al., 2001). More recently, the advancement of machine learning has given rise to more efforts toward using autoencoders for the compression of active seismic data (Valentine and Trampert, 2012).

In this article, we provide a quantitative assessment of the suitability of three methods—wavelet decomposition, zfp floating point compression, and low-rank singular value decomposition (SVD) approximation—for compressing passive seismic data. This assessment focuses on the errors in analyses rather than the error in the raw data. We choose these methods, in part, because of the potential for analytical bounds on these errors' propagation. We give a brief overview of these methods, their comparison in different metrics for assessing the integrity of reconstructed data, and the effect of compression on event detection. In addition, we have released open-source software to enable easy application of these error analysis workflows to additional compression schemes and datasets in the future.

1. Colorado School of Mines, Golden, Colorado, U.S.A., https://orcid.org/0000-0002-4794-2754 (AHSI); https://orcid.org/0000-0002-3420-4971 (ERM); 2. Virginia Tech, Blacksburg, Virginia, U.S.A.

*Corresponding author: aissah@mines.edu

Background

Here, we provide an introduction to three types of compression that are frequently applied to reduce the size of spatiotemporal scientific data: wavelet compression (1D and 2D), zfp compression, and SVD compression. We use two publicly available DAS datasets for testing and comparing these compression techniques' effects on the characteristics of compressed data and the errors incurred in the results of event detection workflows. These data include one urban dark fiber dataset, called the FORESEE data, and one geothermal microseismicity monitoring dataset, called the Brady's Hot Springs data. Here, we provide an overview of the compression schemes and datasets. Throughout this article, the compression factor is defined as the ratio of the size of the original data to the size of the compressed data.

1D wavelet compression

Given a time-domain signal $f(t)$, we can represent it in terms of a spanning set of wavelet functions. This spanning set is generated using scaled and translated versions of a mother wavelet function, ψ, and a father wavelet function, ϕ. The discrete wavelet transform of f can be computed as follows:

$$f = \sum_{j=L}^{J} \sum_{n=0}^{2^{-j}-1} d_j[n]\psi_{j,n} + \sum_{n=0}^{2^{-J}-1} a_J[n]\phi_{J,n}, \quad (1)$$

in which the inner products $d_j[n] = \langle f, \psi_{j,n}\rangle$ and $a_J[n] = \langle f, \phi_{j,n}\rangle$, denote the detail and approximation wavelet coefficients, respectively. $-L > 0$ is the number of nested subspaces, corresponding to the number of scaling levels represented in the wavelet basis and is related to the number of discrete points in f, that is, $N = 2^{-L}$. The functions $\psi_{j,n}$ and $\phi_{j,n}$ are the scaled and translated versions of the mother and father wavelets, respectively, and are defined as follows:

$$\psi_{j,n}(x) = \frac{1}{\sqrt{2^j}}\psi\left(\frac{x-n}{2^j}\right) \text{ and } \phi_{j,n}(x) = \frac{1}{\sqrt{2^j}}\phi\left(\frac{x-n}{2^j}\right). \quad (2)$$

Typically, this representation has a small number of large-magnitude wavelet coefficients around and leading to areas of high amplitude in time (Mallat, 2009). This property allows the application of the wavelet transform for denoising (Donoho, 1995) or for the preservation of events when used for compression Villasenor et al. (1996). For a predetermined threshold T, we can construct an approximation of our signal,

$$\tilde{f} = \sum_{j=L}^{J} \sum_{n=0}^{2^{-j}-1} d_j[n]\psi_{j,n} + \sum_{n=0}^{2^{-J}-1} a_J[n]\phi_{J,n}$$
$$- \sum_{j=L}^{J} \sum_{n=0}^{2^{-j}-1} \tilde{d}_j[n]\psi_{j,n} + \sum_{n=0}^{2^{-J}-1} \tilde{a}_J[n]\phi_{J,n}$$
$$\tilde{a}_J[n] = \begin{cases} 0, & |a_J[n]| \le T \\ a_J[n] - T \times \mathrm{sign}(a_J[n]), & |a_J[n]| > T \end{cases}$$
$$\tilde{d}_j[n] = \begin{cases} 0, & |d_j[n]| \le T \\ d_j[n] - T \times \mathrm{sign}(d_j[n]), & |d_j[n]| > T \end{cases}. \quad (3)$$

The error in approximation E is then

$$E = \sum_{j=L}^{J} \sum_{n=0}^{2^{-j}-1} \tilde{d}_j[n]\psi_{j,n} + \sum_{n=0}^{2^{-J}-1} \tilde{a}_J[n]\phi_{J,n}. \quad (4)$$

If a large enough T is selected, this approximation has few nonzero coefficients that can be encoded to achieve compression. Hence, the threshold determines the amount of compression and the associated error in the approximation.

2D wavelet compression

To exploit additional redundancies in 2D datasets, wavelet compression can be performed using 2D wavelet decomposition (Villasenor et al., 1996). In this approach, separable 2D wavelets derived from 1D wavelets are used to decompose the data in both dimensions. For 2D data, f, the wavelet decomposition can be computed as follows:

$$f = \sum_{j=L}^{J} \sum_{m=0}^{2^{-j}-1} \sum_{n=0}^{2^{-j}-1} \left(d_j^1[m,n]\psi^1_{j,m,n} + d_j^2[m,n]\psi^2_{j,m,n} + d_j^3[m,n]\psi^3_{j,m,n}\right) + \sum_{m=0}^{2^{-J}-1} \sum_{n=0}^{2^{-J}-1} a_J[m,n]\Phi_{J,m,n}, \quad (5)$$

in which ψ^1, ψ^2, ψ^3, and $\Phi(x,y)$ are the 2D wavelets and are computed from the 1D wavelets as $\psi^1(x,y) = \phi(x)\psi(y)$, $\psi^2(x,y) = \psi(x)\phi(y)$, $\psi^3(x,y) = \psi(x)\psi(y)$, and $\Phi(x,y) = \phi(x)\phi(y)$. The coefficients $d_j^1[m,n]$, $d_j^2[m,n]$, and $d_j^3[m,n]$ are referred to as detail coefficients, and $a_J[m,n]$ are called approximation coefficients. Similar to the 1D case, we can threshold the coefficients to achieve compression with approximation error.

High-dimensional wavelet compression has been studied to achieve higher compression rates for active source seismic data organized by streamer number, shot number, sensor, and time dimensions (Villasenor et al., 1996). For passive DAS data, we investigate how 1D wavelet compression does in comparison to 2D compression considering the different channels as a dimension in space in addition to the recording in time.

zfp floating-point compression

The zfp compression technique uses processes such as block transforms and embedded coding, commonly used in image compression, to perform compression that is suitable for a variety of floating-point scientific data, as detailed by Lindstrom (2014). We briefly outline the process here. For d-dimensional data, the data array is sectioned into blocks of 4^d values, which are assumed to be approximately continuous within any block. Each block is compressed separately; the following are the steps during compression that may introduce some errors:

1. Convert floating point values in the 4^d block to scaled integers with a common exponent.

2. Perform a block transform to introduce some sparsity into the integer representation.
3. Encode the numbers in the sparse representation only. This encoding takes sparsity into consideration and only uses as many bits as required to save nonzero entries. The process also allows for specified bits to be allocated for each block (fixed rate), the number of binary exponents to encode (fixed precision), and the maximum error allowed for each floating point value (fixed accuracy).

Although Wade (2020) released a zfp format and implementation designed to handle triggered active-source seismic data formats, here we use the general zfp implementation to align with a wider variety of zfp error analysis studies. Although this article focuses on analyzing the errors introduced in passive seismic data workflows, other teams have previously studied the numerical error due to zfp (Diffenderfer et al., 2019) and its effect on workflows for several fluid dynamics and plasma physics problems as well as climate models (Laney et al., 2013; Baker et al., 2016; Poppick et al., 2020).

SVD compression

The SVD can be used to decompose data represented in matrix form into the product of three matrices: the left singular vectors, a diagonal singular value matrix, and the right singular vectors, in which the singular values represent the amount of variation in the data explained by each singular vector. SVD has been widely used in various scientific fields (e.g., signal processing, image compression, data mining, and machine learning) because of its ability to identify and capture the highest possible amount of variability in the data. Seismic data collected through DAS can be organized as "channels by samples" to obtain seismic data matrices. The application of SVD to these matrices offers a unique opportunity to decouple the channels and samples and enable efficient processing (Martin, 2019).

Although constructing the full SVD can be computationally expensive, randomized SVD is a powerful and efficient algorithm for computing partial SVD of large-scale matrices using randomized projection methods to quickly approximate the dominant singular vectors and singular values. One of the primary benefits of randomized SVD is its ability to efficiently decompose large matrices storing high-dimensional data (Halko et al., 2011). This makes it an attractive tool for a wide range of scientific applications. Given data stored a matrix $D \in R^{N_c \times N_t}$, in which N_c is the number of channels, r is the rank; and N_t is the number of time samples. To achieve compression, we construct a low-rank (rank, r) approximation, D_r $U \Sigma V$ using randomized SVD, in which $U \in R^{N_c \times r}$, $\Sigma \in R^{r \times r}$, $V \in R^{r \times N_t}$. Then we combine U and Σ into an $(N_c \times r)$ matrix. Storing $(N_c \times r)$ and $(r \times N_t)$ matrices provides a compression factor of $\frac{N_c N_t}{r N_t N_c}$. "Compression factor" as used here is defined as the ratio of the size of the original data to the size of the compressed data.

The FORESEE data

To study the use of lossy compressed data for passive data exploration, we used data from the FORESEE urban DAS study. This study's publicly available data were continuously recorded between April 2019 and October 2021, resulting in ~46 TB of data (Zhu et al., 2021). The data examples in Figure 1 illustrate two instances of recordings in the dataset, one with a few events due to thunder earthquakes, and the other with 30 min of passive data with vehicles and noise due to infrastructure. These examples provide visual insights into the characteristics of the recorded data, facilitating subsequent data exploration, analysis, and interpretation. This dataset contains recordings of signals like these and others from both natural and anthropogenic sources, providing a comprehensive representation of the seismic activity in the urban environment (Zhu and Stensrud, 2019; Zhu et al., 2021).

The Brady's Hot springs data

To study the use of lossy compressed data for microseismicity detection and extraction of arrival times for tomographic imaging, we used data from the Brady's Hot Springs geothermal field. This was recorded in 2016 in Nevada as part of an investigation into the feasibility of using DAS for cost-effective monitoring of geothermal reservoirs. The data consist of ~8.7 km of fiber-optic cable deployed horizontally in a shallow backfilled trench, with 1.021 m channel spacing. The data, which were shared publicly, are sampled at 1000 samples per second and recorded ~40 TB of data in 15 days (Coleman, 2016). The dataset recorded microseismic activities that have been previously studied and cataloged (Li and Zhan, 2018). We take advantage of the existing workflow and resulting catalog to compare the detectability of events and arrival times in various types of compressed data.

Computational Experiments

Errors are expected in lossy compressed seismic data, and these errors can affect the quality of the data as well as the results obtained from seismic processing workflows. These errors can cause distortions in the seismic waveforms and may propagate through the seismic processing workflow, affecting subsequent steps such as imaging and inversion. Ultimately, this can lead to interpretation errors based on seismic properties, imaging, and inversion results. Therefore, it is crucial to carefully evaluate the effects of lossy compression on seismic data and the extent to which these propagate through processing workflows. Here, we present several experiments designed to explore how errors from varying compression types and ratios propagate into (1) the level of noise in the data, (2) the distribution of error across frequency ranges, and (3) errors in key metrics for microseismic event detection.

Norm of error in data

To investigate the level of noise introduced by different compression schemes at various compression rates, we studied

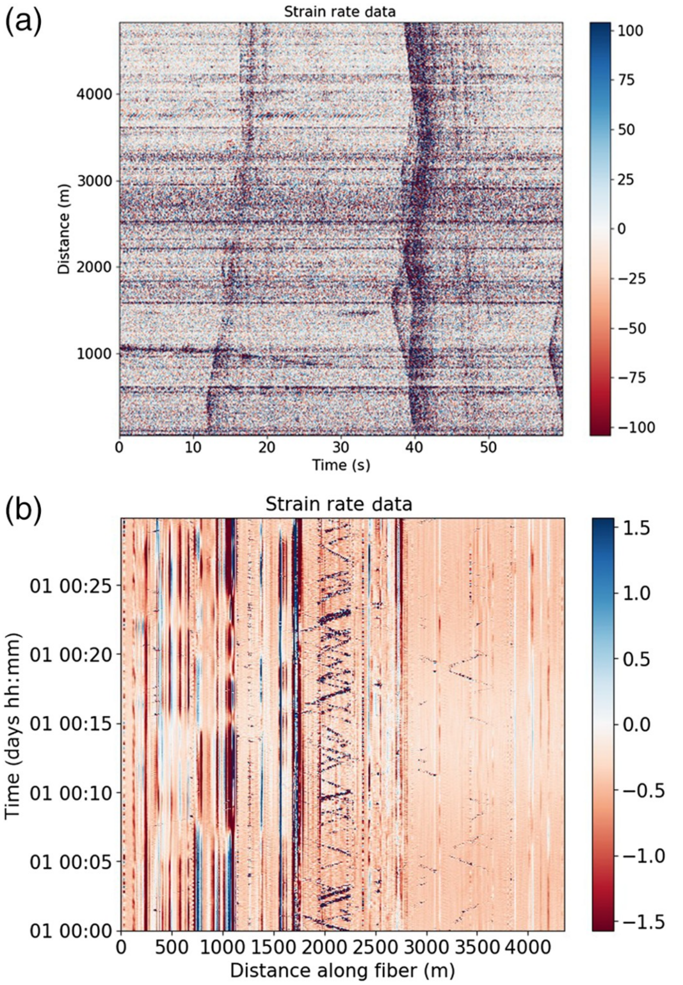

Figure 1. Data from the Penn State Fiber-Optic foR Environment SEnsEing (FORESEE)-urban project. (a) An example of 60 s of continuous data recorded at 03:33:35 on 15 April 2019. This shows recordings of multiple thunder earthquakes starting at 10 and 40 s. It also shows some traffic noise around 1000 m from 0 to 25 s. (b) Twenty min data used for frequency preservation experiment recorded from UTC 00:00:04 to 00:20:04 on 1 August 2019. The coherent signals here are traffic noises between 1500 and 2500 m along the fiber. These visual representations provide valuable insights into the characteristics of the recorded data, offering us a better understanding of what to anticipate during the analysis and interpretation of the compressed data. The color version of this figure is available only in the electronic edition.

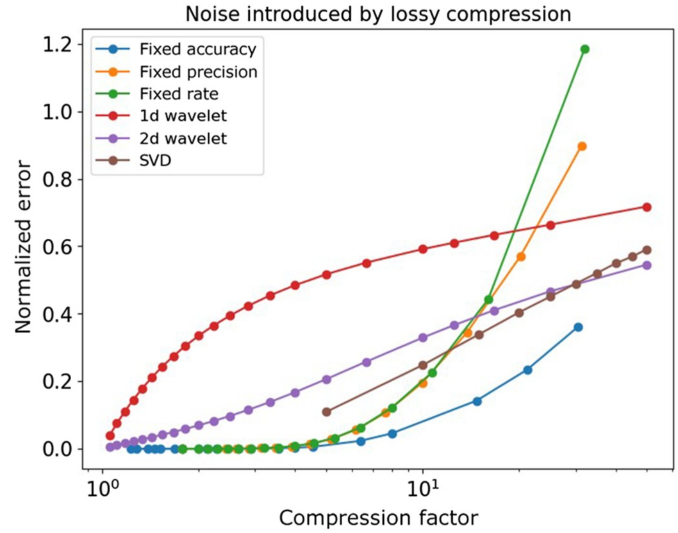

Figure 2. Frobenius norm of noise introduced in data at various levels of compression for compression with wavelet decomposition, singular value decomposition, and zfp floating point compression (three modes—fixed accuracy, fixed precision, and fixed rate). The color version of this figure is available only in the electronic edition.

10 days of data from the FORESEE urban project recorded from UTC 23:51:35 on 04/09/2019 to UTC 00:07:35 on 04/21/2019. We selected this particular time range because it contains a diverse range of recorded signals that we wish to preserve when compressing passive seismic data. For each compression scheme, the comparison workflow is as follows:

1. Begin with data $D \in R^{N_c \times N_t}$.
2. Compress D.
3. Reconstruct compressed data $\tilde{D} \in R^{N_c \times N_t}$.
4. Compute the normalized Frobenius norm error defined as $\|D - \tilde{D}\|_F / \|D\|_F$.

The results of this test are presented in Figure 2. Among the three modes of zfp compression, fixed accuracy maintains lower errors than fixed precision and fixed rate at similar compression factors. This can be attributed to the fixed accuracy mode encoding as many bit planes as necessary to achieve a specific absolute error; the other modes encode a fixed amount of bit planes irrespective of error level. The drawback to the fixed accuracy mode is that it requires complete knowledge of the range of values in the data to provide an absolute error tolerance, making it unsuitable for some on-the-fly compression of streaming data. When comparing 2D and 1D wavelet compression, it is observed that 2D wavelet compression results in a lower error at the same compression rate. This improved performance could be attributed to the additional dimension available for compression, which allows for more effective use of space and time redundancy (Villasenor et al., 1996).

Regarding how the different compression schemes compare, we found that the three modes of zfp compression introduce the least noise up to a compression rate of ~15×, followed by SVD, 2D wavelet, and then 1D wavelet compression in the same range. However, at higher compression rates, zfp compression begins to incur errors at a higher rate, with only fixed accuracy mode maintaining lower errors than SVD and 2D wavelet compression. The errors incurred by wavelet compression may be attributed to its denoising quality. However, at

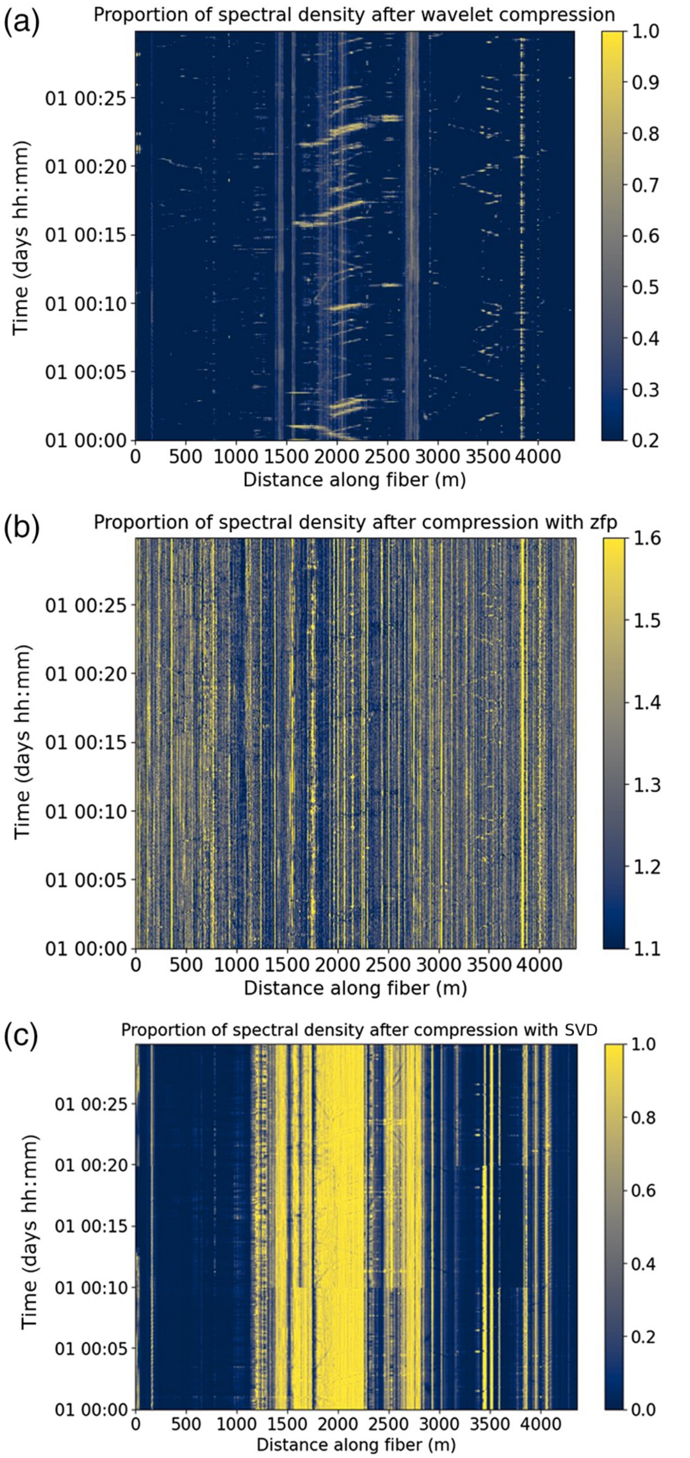

Figure 3. Frequency preservation with respect to compression for (a) wavelet compression, (b) zfp compression, and (c) singular value decomposition (SVD) compression. These images are $N_t = T_w \times N_c$ data in time and channel, respectively, obtained by averaging $P_{\tilde{D}=D}$ along the frequency axis. The color version of this figure is available only in the electronic edition.

higher compression rates, compression may be depleting the real signals to some extent. SVD produces errors that grow at a nearly linear rate relative to the log of the compression factor. The factors governing the error bound is an area that may be explored in future studies.

To facilitate meaningful comparisons, we selected a specific mode for wavelet compression and zfp compression, assuming that the various modes within each compression method yield comparable error characteristics. For wavelet compression, we opted for the 2D mode because of its observed tendency to introduce a lower norm of noise, as evidenced by the previous analysis. On the other hand, for zfp compression, we chose the fixed precision mode instead of the fixed accuracy mode, despite the latter's exhibiting the least norm of noise. The fixed precision mode allows for relative error control and can be applied to unexplored datasets without requiring specific fine-tuning for optimal tolerable error and compression ratio. This consideration becomes crucial if the selected compression method is intended for use on data as they are being gathered.

Errors in frequency content

To assess the extent to which the frequency content of the data is preserved following lossy compression, a methodology was used whereby the proportion of the power spectrum retained at each frequency was calculated and then averaged across frequencies. The step-by-step process is as follows for each file containing a window of data in time:

1. Obtain $\tilde{D}$ previously defined by compression and subsequent decompression of D.
2. Identify the power spectrum, $P(\tilde{D})$ of $\tilde{D}$ in predetermined time windows, T_w (5 s for this experiment) such that $P(\tilde{D}) \in N_c \times N_t=2 \times N_t=T_w$.
3. Divide the power spectrum at each frequency by the original power spectrum, that is, $P_{\tilde{D}=D} = \frac{P(\tilde{D})}{P(D)}$.

The outcome was a 3D data cube for $P_{\tilde{D}=D}$ that captured information across time windows of size $N_t=T_w$, frequencies of size $N_t=2$, and channels N_c. Weighted averaging was then performed along each dimension to investigate the patterns of frequency preservation across the various dimensions.

The data obtained by averaging the ratios of the compressed to original energy across all frequencies can be represented in two dimensions (time by channels), revealing the variations in frequency preservation. As depicted in Figure 3, the patterns observed in this 2D matrix closely resemble the trends in the data shown in Figure 1. In the case of wavelet and low-rank compression, a general reduction in energy can be attributed to the thresholding operation used in both the compression methods. In addition, whereas wavelet decomposition exhibits better preservation of small events, SVD compression tends to preserve mostly high-amplitude events. On the other hand, zfp compression leads to a general increase in energy but preserves the energy around the identified events to levels close to pre-compression levels. This may be due to the spurious frequencies introduced at multiples of a quarter of the sampling rate as

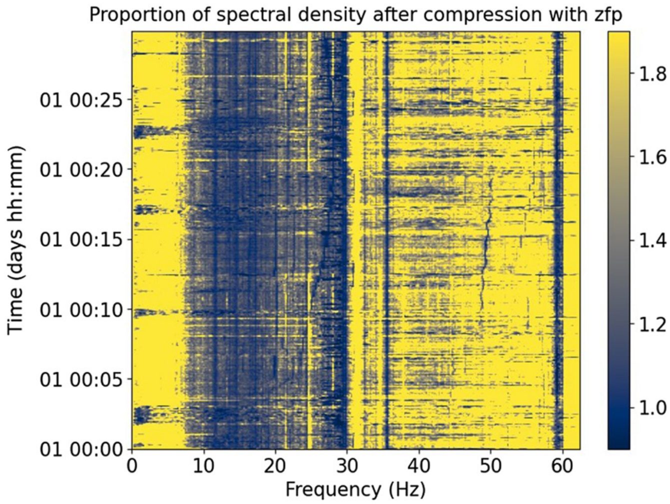

Figure 4. Frequency preservation with respect to compression for zfp compression. This image is constructed from $N_t = T_w \times N_t = 2$ data in time and frequency, respectively, obtained by averaging $P_{\tilde{D}=D}$ along the channel axis. The color version of this figure is available only in the electronic edition.

depicted in Figure 4. This effect can be attenuated when there are other strong frequencies corresponding to events but tends to be exaggerated in noisy parts when there are not many other strong frequencies. In contrast, such false frequencies are not observed in wavelet or low-rank compression.

Finding a low-rank approximation of the data can result in varying levels of representation fidelity for different channels (i.e., columns of the data matrix). This discrepancy may explain the observed trend of certain channels exhibiting better-preserved frequency characteristics compared with others when those channels have high-amplitude events (e.g., vehicles). Notably, the channels that were better preserved through compression as evaluated in the time domain also showed improved representation in the frequency domain. Despite compression being applied in smaller time windows, the channels that demonstrated good preservation remained consistent throughout the entire time period under consideration (i.e., well-preserved channels continued to be well-preserved in most other time windows). This consistency suggests that these specific columns likely account for a significant portion of the overall variation of the data. However, when many spatially distributed ray paths are needed (e.g., many source–receiver pairs for imaging), normalization prior to compression may be required to ensure that many channels' compressed data are high quality.

Changes in template matching event detection

Template matching enables the detection of similar events by comparing known seismic events, referred to as templates, with continuously recorded seismic data. The underlying principle of template matching involves calculating the cross-correlation coefficient of the template waveform and recorded seismic data at different time intervals. This correlation coefficient is normalized to account for variations in signal amplitude and noise levels and provides a quantitative measure of similarity between the template and the recorded data. The significance of template matching lies in its ability to enhance the detection capability for microseismic events. By applying cross-correlation analysis, even weak events that may be buried within the background noise can be identified and accurately located (Gibbons and Ringdal, 2006).

To evaluate the impact of noise introduced by different lossy compression schemes on event detection, we conducted a series of experiments using a template matching workflow. In particular, we investigate two questions:

1. To what extent is the array-wide detection significance of varying-size events impacted?
2. Are there biases or increases in the variability of event times picked across the array as higher compression ratios are used for any types of compression?

To test microseismic event detection via template matching, we use the Brady's Hot springs data. This dataset has been previously studied, and various microseismic activities have been cataloged providing a baseline catalog for comparison (Li and Zhan, 2018). Our goals in this study are to apply a similar workflow as outlined by Li and Zhan (2018) and to compare the performance of event detection at various compression rates for 2D wavelet, zfp, and SVD compression. By analyzing the results of these experiments, we can gain valuable insights into the trade-off between compression rates and event detection performance for the compression schemes considered.

To provide a detailed view of the template matching process, we calculated the normalized cross correlation of each channel's recording of the template event in Figure 5 with its recording of the noise from UTC 14 March 2016 beginning at 08:39:13. The resulting array-wide normalized cross correlation is shown in Figure 5.

Typically, template matching would be carried out on small collections of point sensors, with large amplitude normalized cross correlations being considered a possible event and potentially a voting system among sensors to ensure multiple sensors detected the same possible event. With thousands of sensors, we use a simple metric: calculating the array-wide average of the normalized cross correlations at each time, then calculating the envelope of the resulting time series. Despite time delays in the original event as it moves across the array, if a similar event occurs in the same location, the large normalized cross correlations across the array are expected to occur at the same time lag. Thus, averaging across all sensors does not cause destructive interference.

The average envelope for each type of compression at multiple compression levels is shown in Figure 6. Three events were

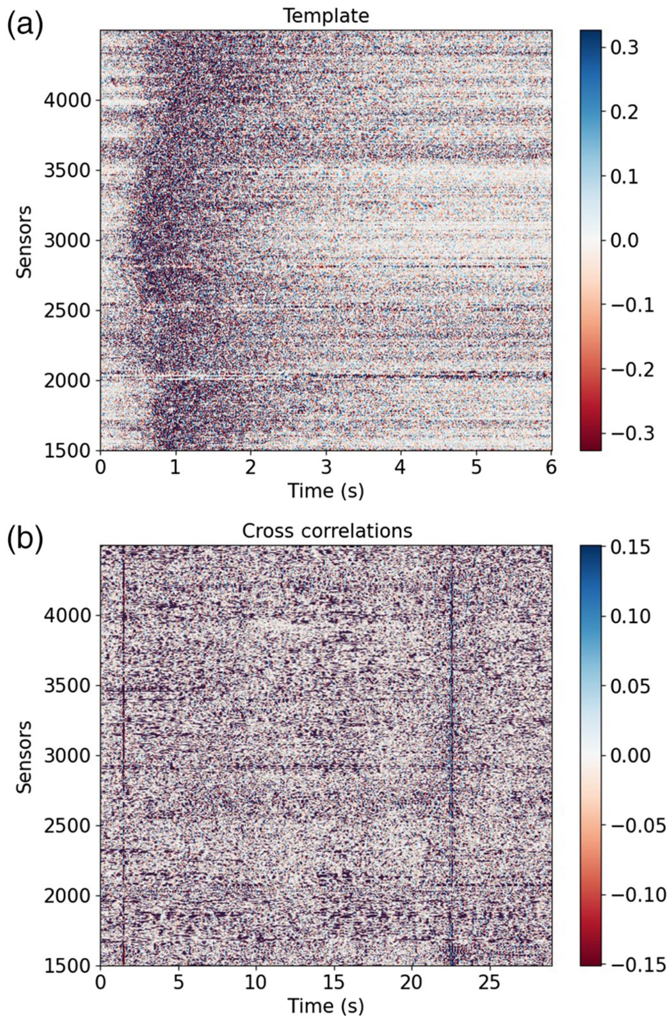

Figure 5. (a) An example template event recorded at Brady Hot Springs at 08:39:05.24 on 14 March 2016. (b) The array-wide template matching results show the normalized cross correlation of each channel's continuous recording with its template event recording. Vertical lines indicate many channels with high similarities at a particular time. The color version of this figure is available only in the electronic edition.

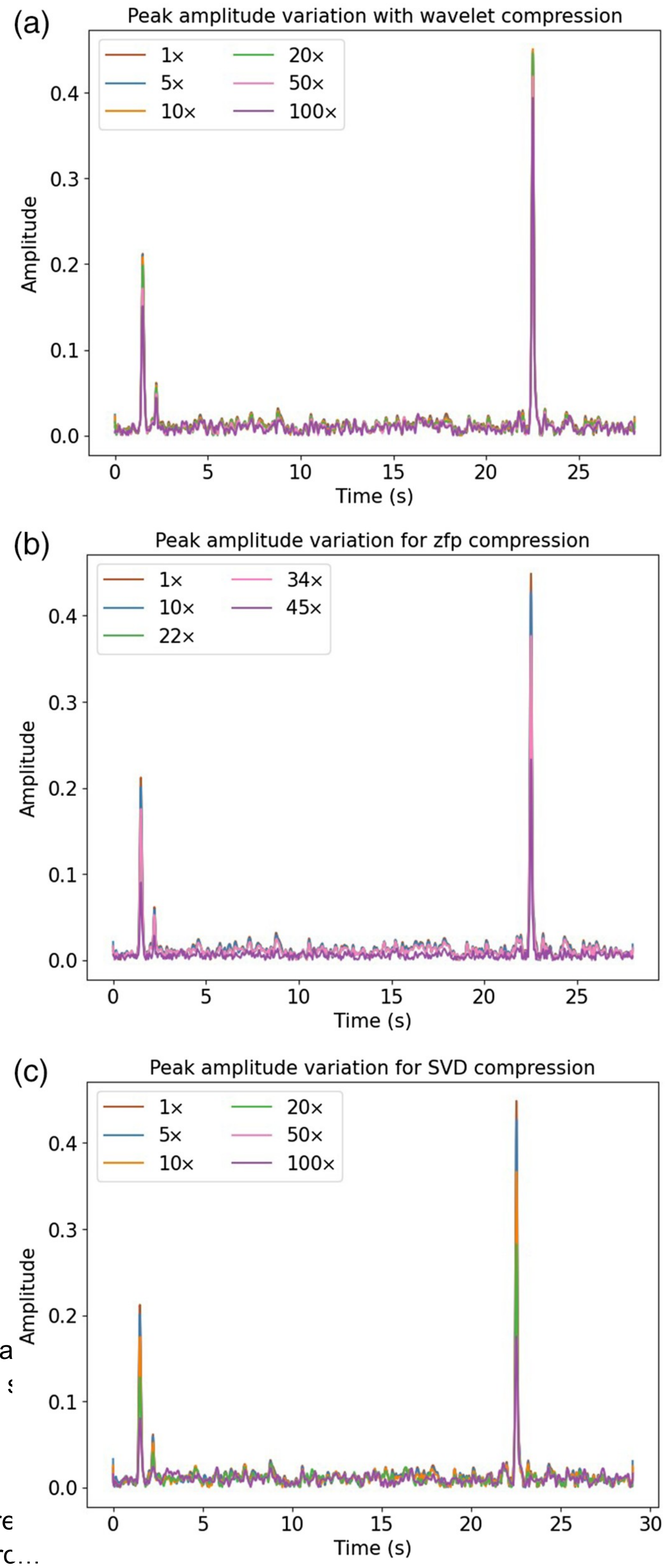

Figure 6. The envelope of the average normalized cross correlations during three events in Figure 5 shows that the event picks are largely similar for various levels of compression. This was tested with (a) 2D wavelet-compressed data, (b) zfp-compressed data, and (c) SVD-compressed data. The color version of this figure is available only in the electronic edition.

detected, even though these events were barely distinguisha in the raw data. We refer to the first event (between 1 and 2 s as event 1, which is the midsize event, the second event (between 2 and 3 s) as event 2, which is the small event, and the third event (between 22 and 23 s) as event 3, which is the largest event. We see that for all three types of compre sion, the three events appear to be largely distinguishable fr… the background noise level. 2D wavelet compression maintains a more constant peak amplitude across compression ratios (1× original, 5×, 10×, 20×, 50×, and 100×) than do zfp and SVD compression, although there is a small amount of amplitude loss at higher compression ratios (noticeable at 20×, 50×, and 100×). The zfp compression shows some amplitude loss in the events, which particularly makes it difficult to

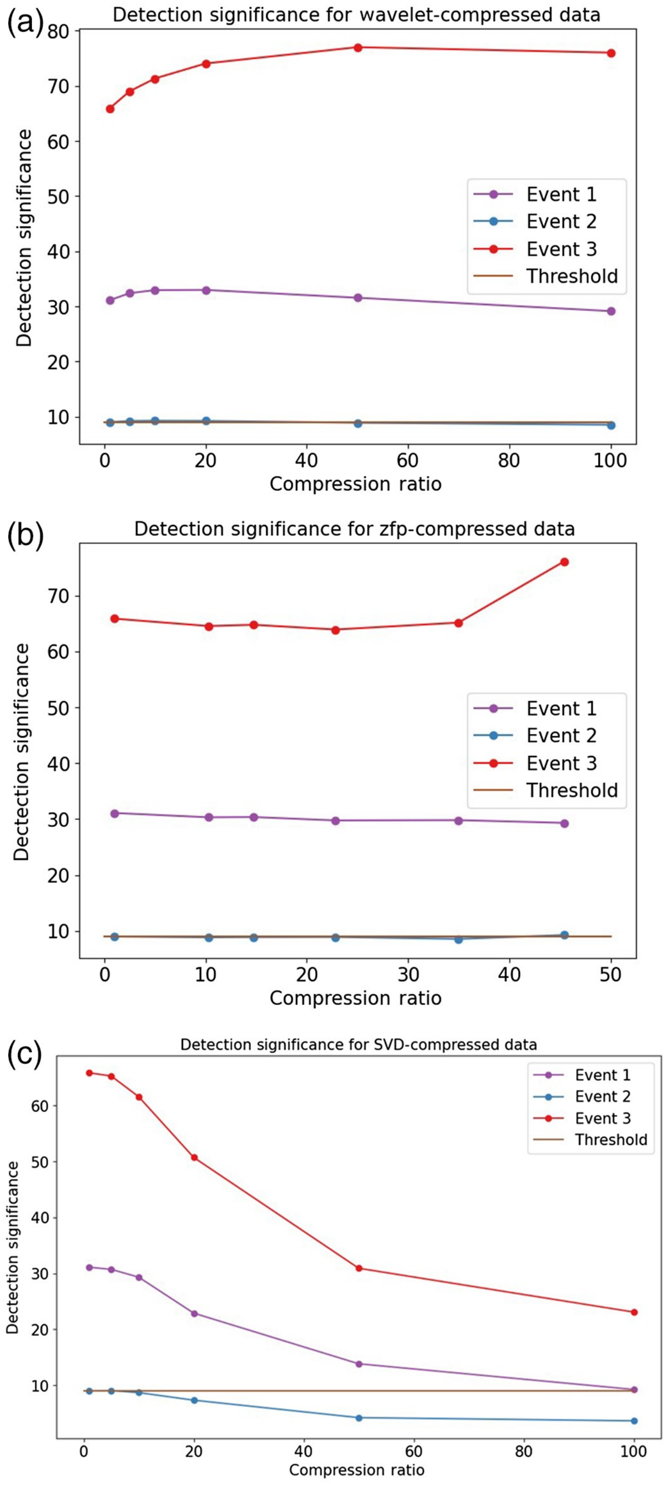

Figure 7. Change in detection significance with compression rate for (a) 2D wavelet-compressed data, (b) zfp-compressed data, (c) low-rank SVD-compressed data. The color version of this figure is available only in the electronic edition.

distinguish event 2, but events 1 and 3 are clearly above the noise level at all compression levels (1× original, 10×, 22×, 34×, and 45×). The SVD compression leads to significant energy loss, particularly at the higher compression ratios (e.g., the peak of events 1 and 3 for 100× compressed data are less than half their original amplitudes).

As in Li and Zhan (2018), we use the detection significance of each event pick to provide a single array-wide value to quantitatively compare the preservation of each event when using various compression schemes and ratios. The detection significance is defined as $\frac{CC_i - M}{MAD}$ for a value CC_i on the array-wide average of the normalized cross correlations at the ith time sample. M is the median of CC and MAD is the median absolute deviation defined as $\mathrm{median}|CC - M|$. We set a detection significance minimum threshold of 9, which was set to provide a reasonable bound on false detections following Li and Zhan (2018). The detection significance for each of the three events across all compression schemes and compression ratios is shown in Figure 7. All three compression schemes preserve events 1 and 3 (the midsize and large events) as picked events above the threshold, even at high-compression ratios (e.g., 45× and 50× for all, and 100× for wavelet and SVD). The SVD-compressed data have the largest drop in detection significance relative to 2D wavelet-compressed and zfp-compressed data. At 100× SVD compression, midsize event 1 is barely above the detection significance threshold. Even in the original uncompressed data, the small event 2 starts out very close to the detection threshold. With SVD compression at 20× and higher compression and with 2D wavelet compression at the 100× level, event 2 drops substantially below the threshold for detection significance. It is not surprising that a barely detectable event in the raw data could be lost in some of the highly compressed data. Although all events compressed with SVD show a drop in detection significance, the 2D wavelet compression and zfp compression of the large event 3 data actually show an uptick in detection significance at high-compression ratios, likely indicating some denoising occurring to further emphasize this largest event during the compression process.

Figure 8 shows the event detected using the template matching workflow discussed in the original data. When compressed data are used for all the events in this catalog, we see similar trends to the one explained by the small-scale case. The results of this are summarized in Figure 9 showing the variation in detection significance with compression. In these plots, we expect a trend with a slope of 1 in the situation in which there are not any changes in detection significance. For wavelet compression (Fig. 9a), we have a lot of points close to this trend even at high-compression rates, although points with smaller detection significance show a slight decrease and points with high-detection significance show some increase indicating some denoising effect. This leads to smaller events eventually being missed at higher compression rates. SVD compression (Fig. 9c) shows a similar trend but shows more reduction in detection significance for the mid- and small-size events. The image for zfp (Fig. 9b) shows less predictability; some small events increase in detection significance even though most events are observed to be reducing in detection significance.

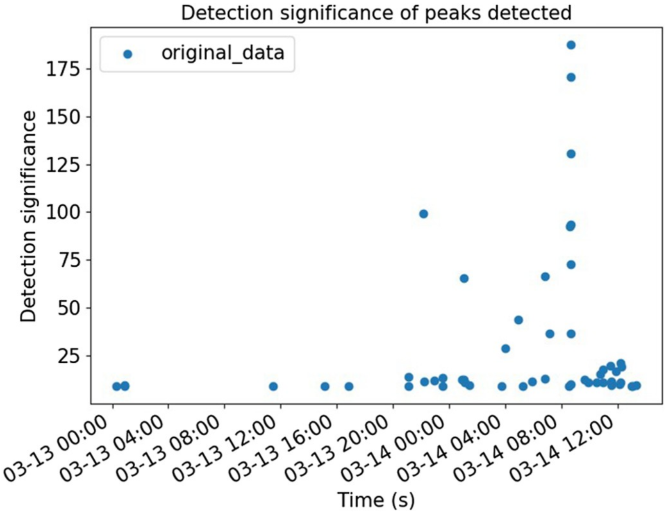

Figure 8. Catalog of events detected by template matching using the uncompressed data. The color version of this figure is available only in the electronic edition.

To address the question of whether event pick times are r able, we need to quantify the distribution of how each event's pick times change throughout the array of compressed data compared with the event pick times on the original data. In particular, we need to know if the median pick time shift is staying very close to 0 (indicating there is no substantial array-wide bias) and compare the rate at which the minimum Q1, Q3, and maximum values are spreading apart for higher compression ratios. For each type of compression and for ea compression ratio, we created a box and whiskers plot for the distribution of each event's changes in pick times across all channels, which is shown in Figure 10. For each type of compression, these boxplots are overlaid for all three events (col coded) so that the spread for a small, midsized, and large ev can be compared easily.

There is not an apparent median bias for any of the events or compression types with a compression ratio <100×. At 100×, there appear to be a slightly (<0.05 s) late median of picks in SVD compression and a positive skew distribution (based on investigation of min, max, Q1, and Q3) to the picks in 2D wavelet compression at 100×, though the median of the wavelet distribution appears to be unbiased. We see that for types of compression, the extremes and quartiles spread out more with higher compression ratios. In the range of 5×–15× compression, all three compression schemes perform reasonably well, with 5× 2D wavelet, 14× zfp, 5× and 10× SVD compression all yielding distributions for all three events that are completely contained within a ±0.1 second shift. Among these low-ratio compression schemes, 2D 10× wavelet compression does perform the worst with the largest spread for all three events, and the small event 2 distribution only has its inner quartile range (Q1–Q3) contained within ±0.1 s,

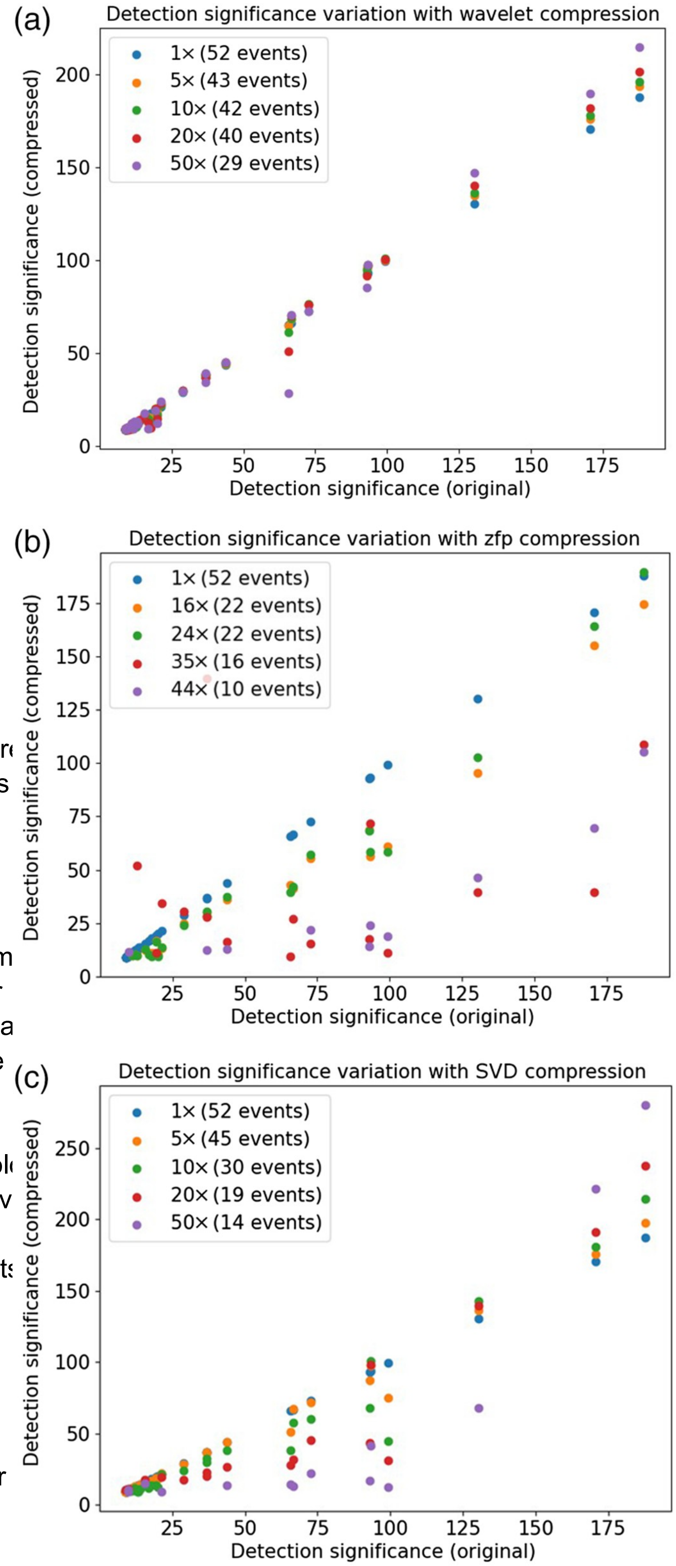

Figure 9. Trends in detection significance and events detected as compression rate is increased for (a) 2D wavelet, (b) zfp, and (c) SVD compressions. The color version of this figure is available only in the electronic edition.

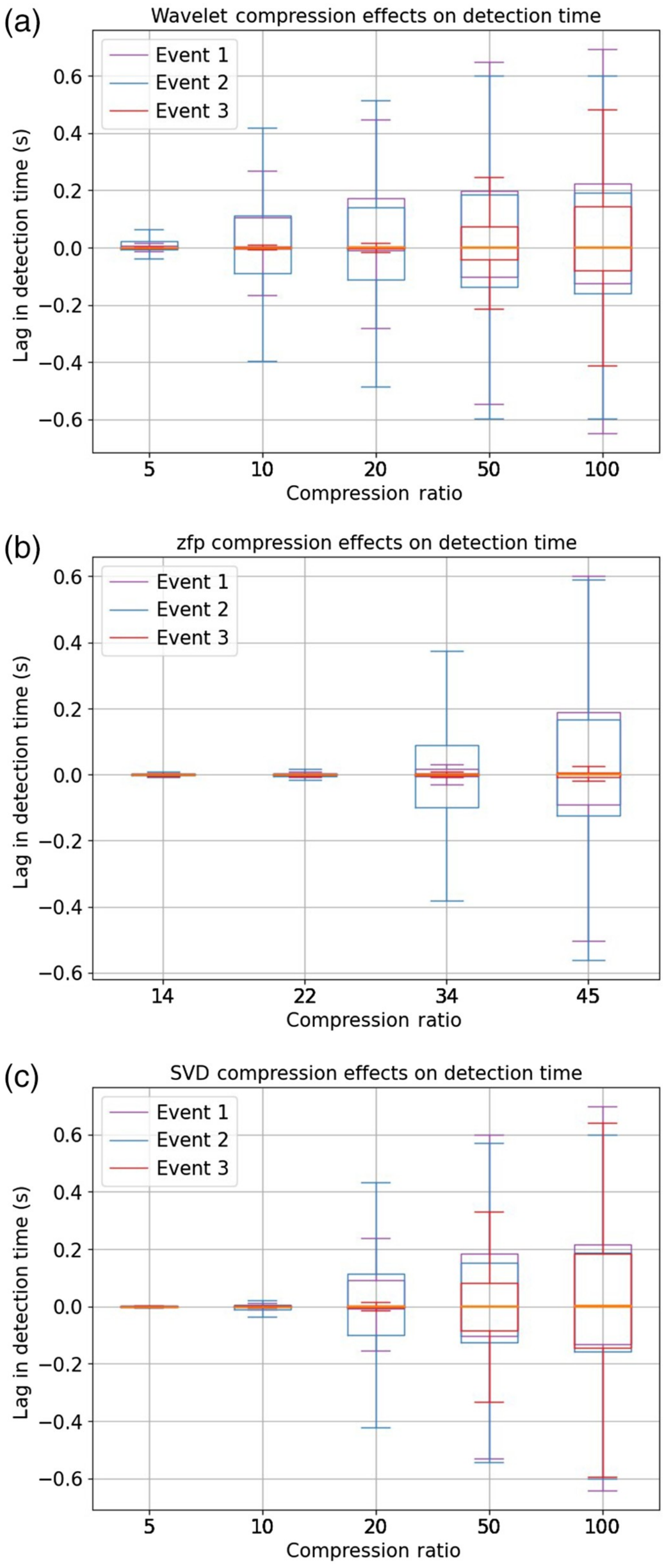

Figure 10. Boxplots show the distribution across all channels in the array of picked event times from template matching applied to the (a) 2D wavelet-compressed data, (b) zfp-compressed data, and (c) low-rank SVD-compressed data. These are shown at various compression ratios for three events. The color version of this figure is available only in the electronic edition.

whereas its extremes have errors bounded by 0.3 s. In the range 20×–35×, we see that zfp at 22× has the only distributions for all three events that are completely contained within ±0.1 s, although both 2D wavelet and SVD perform similarly well at 20× compression for the large event 3. In the range 45×–50×, we see that 2D wavelet at 50× and zfp at 45× have very compact distributions for the large event 3, whereas SVD at 50× has a substantial spread in event pick shifts. All three compression schemes in this compression ratio range perform similarly for events 1 and 2, with extremes that exceed ±0.4 s and interquartile ranges that are bounded within ±0.2 s. The zfp did not allow higher compression ratios, but 2D wavelet and SVD compression were tested at the 100× ratio. 2D wavelet compression outperformed SVD compression on the large event 3 in the sense of providing a more compact distribution of time shifts, although 2D wavelet's distribution of pick time shifts is skewed to have larger positive shifts. Both schemes performed similarly at the 100× level on events 1 and 2, with extreme shifts around ~±0.6 s.

Discussion

Overall, we see that although zfp has the lowest data errors at lower compression ratios, wavelet compression (especially in 2D) has lower errors at higher compression ratios, and low-rank SVD has an error growth that sits in between zfp and wavelets. Wavelets strongly improve broadband representation of strong events over quiet noise, and SVD tends to have better broadband representation of louder signals, but zfp tends to more evenly distribute errors in the frequency content across loud and quiet events. Using a microseismicity dataset, we could see that after feeding compressed data through a template matching workflow, all types of compression could preserve the events at smaller compression ratios. SVD compression tended to have the largest drop in detection significance at high-compression ratios, although it still preserved the detection significance of a midsize and large event even at 100× compression. The unbiased picks with increasing variability caused by higher compression ratios from Figure 10 suggest the opportunity to design a postprocessing scheme that promotes spatial coherency in event picks across the array. In this way, more reliable picks can be used from any highly compressed data, particularly as an input to the event location or for tomographic imaging using microseismic events. This analysis workflow was extended to a 36-hour period recording with 52 events of varying detection significance, and we found that wavelet compression preserved the detection significance better than zfp and SVD compression at similar compression ratios and tended to increase detection significance for larger events and higher compression ratios (i.e., emphasizing and denoising large events). Zfp compression typically led to a reduction in detection significance across all event sizes. SVD compression tends to reduce detection significance for smaller to midsize events and tends to increase detection significance for larger events, with more decrease or increase in significance for higher compression levels. Ultimately, SVD and wavelet representations have been integrated into a larger number of analysis workflows that can operate on data directly in their compressed representation, which may lead us to prefer these in some contexts, but zfp is being increasingly used in scientific computing, so algorithms incorporating zfp may be on the horizon in the coming years. Beyond the scope of this study, further analysis should be done on the effects of lossy compression in ambient noise interferometry for imaging.

Conclusions

New technologies to continuously collect high-resolution seismic data for long periods of time are pushing us to consider lossy compression as a means of reducing data movement time, archival requirements, and processing or visualization time (particularly when interactive workflows are desirable). In this report, we compare the benefits and drawbacks of wavelet compression, zfp compression, and SVD compression at compression ratios ranging between 5× and 100×. These are tested on two public DAS datasets: an urban dark fiber experiment as well as a geothermal field microseismicity monitoring experiment. We see that different compression schemes have the lowest errors at low-compression rates versus high-compression rates, and we compare these errors because they propagate through an entire template matching microseismicity detection workflow.

Data and Resources

The Penn State Fiber-Optic foR Environment SEnsEing (FORESEE) data are publicly available on PubDAS, which can be accessed through Globus Spica et al. (2023). The Brady's Hot springs data are publicly available on the U.S. Department of Energy's Geothermal Data Repository, which is provided alongside example code notebooks for accessing the data through Amazon Web Services (AWS) discussed by Coleman (2016) and Feigl et al. (2016). The Python software to reproduce tests and figures from this article is publicly available at https://github.com/aissah/Issah-SRL-compression-2023.git and is archived at https://zenodo.org/badge/latestdoi/505936568 in its present form. This software is built using the following open-source packages: PyWavelets (Lee et al., 2019), ZFPy (Lindstrom, 2014), Numpy, Matplotlib, and h5py. The supplemental material contains more detailed representations of the compressed data for the various compression types considered in the article. These include the time domain data from the Errors in frequency content section reconstructed after lossy compression and the channelwise normalized cross correlation of each channel's continuous recording with its template event recording used in the event detection analyses. These are pictured for all three compression types considered here. All websites were last accessed in September 2023.

Declaration of Competing Interests

The authors acknowledge that there are no conflicts of interest recorded.

Acknowledgments

The authors thank the National Science Foundation Grant Number 2046387, the Air Force Research Laboratory Subaward Number 62681767-227888 through Stanford University, and the sponsor companies of the Center for Wave Phenomena (CWP); their support made this research possible. The authors thank the Virginia Tech Advanced Research Computing (ARC) and Colorado School of Mines CyberInfrastructure and Advanced Research Computing (CIARC) for computing resources. The authors also thank the students and faculty of CWP and the Martin Group for helpful discussions throughout this work.

References

Averbuch, A. Z., F. Meyer, J.-O. Stromberg, R. Coifman, and A. Vassiliou (2001). Low bit-rate efficient compression for seismic data, IEEE Trans. Image Process. 10, no. 12, 1801–1814, doi: 10.1109/83.974565.

Baker, A. H., D. M. Hammerling, S. A. Mickelson, H. Xu, M. B. Stolpe, P. Naveau, B. Sanderson, I. Ebert-Uphoff, S. Samarasinghe, F. De Simone, et al. (2016). Evaluating lossy data compression on climate simulation data within a large ensemble, Geosci. Model Dev. 9, no. 12, 4381–4403.

Bosman, C., and E. Reiter (1993). Seismic data compression using wavelet transforms, in SEG Technical Program Expanded Abstracts, Washington, DC, 26–30 September 1993, Society of Exploration Geophysicists, 1261–1264, doi: 10.1190/1.1822354.

Coleman, T. (2016). Brady's geothermal field—metadata for DTS and DAS surveys, available at https://www.osti.gov/servlets/purl/1261983/ (last accessed February 2023).

Diffenderfer, J., A. L. Fox, J. A. Hittinger, G. Sanders, and P. G. Lindstrom (2019). Error analysis of zfp compression for floating-point data, SIAM J. Sci. Comput. 41, no. 3, A1867–A1898.

Donoho, D. L. (1995). De-noising by soft-thresholding, IEEE Trans. Inform. Theory 41, no. 3, 613–627, doi: 10.1109/18.382009.

Donoho, P. L., R. A. Ergas, and R. S. Polzer (1999). Development of seismic data compression methods for reliable, low-noise, performance, in SEG Technical Program Expanded Abstracts, Society of Exploration Geophysicists, 1903–1906, doi: 10.1190/1.1820919.

Feigl, K., N. Taverna, and M. Rossol (2016). Porotomo natural laboratory horizontal and vertical distributed acoustic sensing data, available at https://gdr.openei.org/submissions/980 (last accessed February 2023).

Gibbons, S. J., and F. Ringdal (2006). The detection of low magnitude seismic events using array-based waveform correlation, Geophys. J. Int. 165, no. 1, 149–166,

Halko, N., P. G. Martinsson, and J. A. Tropp (2011). Finding structure with randomness: probabilistic algorithms for constructing approximate matrix decompositions, SIAM Rev. 53, no. 2, 217–288, doi: 10.1137/090771806.

Laney, D., S. Langer, C. Weber, P. Lindstrom, and A. Wegener (2013). Assessing the effects of data compression in simulations using physically motivated metrics, in Proc. of the International Conf. on High Performance Computing, Networking, Storage and Analysis, Denver, Colorado, 17–22 November 2013, 1–12.

Lee, G. R., R. Gommers, F. Waselewski, K. Wohlfahrt, and A. O'Leary (2019). Pywavelets: A python package for wavelet analysis, J. Open Source Software 4, no. 36, 1237, doi: 10.21105/joss.01237.

Li, Z., and Z. Zhan (2018). Pushing the limit of earthquake detection with distributed acoustic sensing and template matching: a case study at the Brady geothermal field, *Geophys. J. Int.* **215**, no. 3, 1583–1593, doi: 10.1093/gji/ggy359.

Lindsey, N. J., and E. R. Martin (2021). Fiber-optic seismology, *Annu. Rev. Earth Planet. Sci.* **49**, no. 1, 309–336, doi: 10.1146/annurev-earth-072420-065213.

Lindstrom, P. (2014). Fixed-rate compressed floating-point arrays, *IEEE Trans. Vis. Comput. Gr.* **20**, no. 12, 2674–2683, doi: 10.1109/TVCG.2014.2346458.

Mallat, S. G. (2009). *A Wavelet Tour of Signal Processing: The Sparse Way*, Third Ed., Elsevier/Academic Press, Amsterdam, Boston.

Martin, E. R. (2019). A scalable algorithm for cross-correlations of compressed ambient seismic noise, in *SEG Technical Program Expanded Abstracts*, San Antonio, Texas, 15–20 September 2019, Society of Exploration Geophysicists, doi: 10.1190/segam2019-3216637.1.

Poppick, A., J. Nardi, N. Feldman, A. H. Baker, A. Pinard, and D. M. Hammerling (2020). A statistical analysis of lossily compressed climate model data, *Comput. Geosci.* **145**, 104,599.

Spica, Z. J., J. B. Ajo-Franklin, G. C. Beroza, B. Biondi, F. Cheng, B. Gaite, B. Luo, E. R. Martin, J. Shen, C. Thurber, et al. (2023). Pubdas: A public distributed acoustic sensing datasets repository for geosciences, *Seismol. Res. Lett.* **94**, no. 2A, 983–998.

Valentine, A. P., and J. Trampert (2012). Data space reduction, quality assessment and searching of seismograms: Autoencoder networks for waveform data, *Geophys. J. Int.* **189**, no. 2, 1183–1202, doi: 10.1111/j.1365-246X.2012.05429.x.

Villasenor, J. D., R. A. Ergas, and P. L. Donoho (1996). Seismic data compression using high-dimensional wavelet transforms, in *Proc. of the Data Compression Conference—DCC '96*, Snowbird, Utah, U.S.A., 31 March–3 April 1996, IEEE Computer Society Press, 396–405, doi: 10.1109/DCC.1996.488345.

Wade, D. (2020). Seismic-ZFP: Fast and efficient compression and decompression of seismic data, in *First EAGE Digitalization Conference and Exhibition*, Vienna, Austria, November 30–December 3 2020, 1–5, doi: 10.3997/2214-4609.202032080.

Zhu, T., and D. J. Stensrud (2019). Characterizing thunder-induced ground motions using fiber-optic distributed acoustic sensing array, *J. Geophys. Res.* **124**, no. 23, 12,810–12,823.

Zhu, T., J. Shen, and E. R. Martin (2021). Sensing earth and environment dynamics by telecommunication fiber-optic sensors: an urban experiment in Pennsylvania, USA, *Solid Earth* **12**, no. 1, 219–235, doi: 10.5194/se-12-219-2021.

Manuscript received 18 September 2023
Published online 2 January 2024