

Scaling Up Data-Driven Earthquake Monitoring

Yiyu Ni

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington
2026

Reading Committee:
Marine Denolle, Chair
Brad Lipovsky
Jose Nathan Kutz

Program Authorized to Offer Degree:
Department of Earth and Space Sciences

©Copyright 2026
Yiyu Ni

University of Washington

Abstract

Scaling Up Data-Driven Earthquake Monitoring

Yiyu Ni

Chair of the Supervisory Committee:

Marine Denolle

Department of Earth and Space Sciences

Modern seismology is undergoing a data revolution driven by advances in machine learning, fiber-optic sensing, and cloud computing. This dissertation develops scalable, data-driven approaches to earthquake monitoring that span the full data lifecycle from acquisition and storage to analysis. I compile a curated benchmark seismic dataset from 21 years of Pacific Northwest recordings that improves machine learning phase picker performance in the region. I develop a cloud-native storage framework for large-scale fiber-optic sensing data and demonstrate its effectiveness on an ambient noise cross-correlation workflow using data from the SeaDAS-North deployment. I apply a machine learning wavefield reconstruction method to seafloor fiber-optic data in Cook Inlet, Alaska to enhance earthquake detection and phase picking in an edge computing setting. I present a cloud-native workflow that processes approximately 1.3 petabytes of continuous seismic recordings spanning two decades, producing a publicly accessible global phase pick database. I develop a multi-stage machine learning framework that reconstructs earthquake ground motion fields and estimates probabilistic hypocenter locations from sparse sensor observations, with applications to earthquake early warning. Finally, I provide a comprehensive review of cloud computing and storage in seismology. Together, these contributions advance the scalability of earthquake monitoring and will contribute to the development of next-generation seismic monitoring systems.

Contents

1	INTRODUCTION	I
2	CURATED PACIFIC NORTHWEST SEISMIC DATASET	5
2.1	Introduction	5
2.2	Data Selection and Preparation	7
2.3	Results	18
2.4	Conclusion	28
3	OBJECT STORAGE FOR DISTRIBUTED ACOUSTIC SENSING	30
3.1	Introduction	30
3.2	Background	33
3.3	Object Storage and Format for DAS	36
3.4	Performance Test on DASstore	40
3.5	Application: SeaDAS-N Cross-correlation	45
3.6	Conclusions	49
4	WAVEFIELD RECONSTRUCTION OF DISTRIBUTED ACOUSTIC SENSING: LOSSY COMPRESSION, WAVEFIELD SEPARATION, AND EDGE COMPUTING	51
4.1	Introduction	51
4.2	Data	54
4.3	Methods	56
4.4	Results	60
4.5	Alternative Greedy Data Compression	63
4.6	Offshore Seismological Use Cases	65
4.7	An Edge-Computing Framework	71
4.8	Conclusions	72
5	A GLOBAL-SCALE DATABASE OF SEISMIC PHASES FROM CLOUD-BASED PICKING AT PETABYTE SCALE	74
5.1	Introduction	74
5.2	Methods	76

5.3	Results	81
5.4	Discussion	84
5.5	Conclusions	86
6	RECONSTRUCTION OF EARTHQUAKE GROUND MOTION FIELDS WITH COMPRESSIVE LEARNING AND GENERATIVE REFINEMENT	88
6.1	Introduction	88
6.2	Data	91
6.3	Methods	93
6.4	Results	103
6.5	Conclusions	108
7	A REVIEW OF CLOUD COMPUTING AND STORAGE IN SEISMOLOGY	110
7.1	Introduction	110
7.2	Cloud Storage	113
7.3	Cloud Databases	118
7.4	Cloud Compute	120
7.5	Cloud-Native Applications in Seismology	123
7.6	Community Model Visualization and Dissemination	129
7.7	Education and Training Initiatives	132
7.8	Discussion and Outlook	133
7.9	Conclusion	138
8	CONCLUSION	142
A	ACRONYMS AND ABBREVIATIONS	146
B	SUPPLEMENTARY MATERIALS FOR CHAPTER 2	150
C	SUPPLEMENTARY MATERIALS FOR CHAPTER 4	157
D	SUPPLEMENTARY MATERIALS FOR CHAPTER 5	166
E	SUPPLEMENTARY MATERIALS FOR CHAPTER 7	167
	REFERENCES	170

Listing of Figures

2.1	The event counts of ComCat and exotic catalog included in the AI-ready Pacific Northwest dataset as a function of time.	9
2.2	Number of events arranged by event type of the curated ComCat and Exotic event datasets.	11
2.3	Locations of the events included in the ComCat dataset. The red dashed polygon denotes the authoritative region boundaries of PNSN. The solid lines mark the depth contour of the subduction slab with a 20 km interval ⁹⁶ . Some contours are labeled by the slab depth. The plate boundary between Juan de Fuca and North America Plate (plate depth 0 km) is delineated in the white line. Some events are color-coded white because they are deeper than 60 km. These are intermediate-depth earthquakes. . . .	16
2.4	Distributions of P- and S-wave picking residuals ($t_{ML} - t_{PNSN}$) from the benchmark testing on velocity seismograms. The number in the upper right corner of each subplot shows the mean absolute error (MAE), the root-mean-square error (RMSE) as the uncertainties to the MAE, the mean value of the residual, and the picking completeness in percentage concerning the ground truth. The PNW-retrained Earthquake Transformer outperforms the other four pre-trained models from SeisBench ³⁰⁷ in both picking accuracy and detecting completeness.	18
2.5	Number of streams from each channel type used in the ComCat and exotic event catalogs through time. Short-period (EH?) and broad-band (BH?) sensors were the predominant channels for both ComCat and exotic datasets before 2012, while the recording at higher sampling rates at broadband sensors (HH?) increasingly has become the standard since then. A limited number of streams from strong-motion accelerometer EN? channels is available in the dataset since 2007.	19
2.6	Magnitude types of ComCat events as a function of time. Md and Ml denote duration and local magnitudes, respectively. Mh denotes magnitudes manually inserted by the analysts. Before the PNSN began using the ANSS Earthquake Monitoring System (AQMS) in 2012, 483 events had Ml estimates, and 46,326 events had Md estimates in the dataset.	24
2.7	Waveform from event uw10583988 (M4.8 Brinnon, Washington, 25 April 2003) included in the dataset. Only the vertical component is shown. The blue and red vertical lines show P- and S-wave arrival picked, respectively. The amplitude units are in counts.	25
2.8	Distribution of signal-to-noise ratios (SNR) of the traces from ComCat and exotic events. SNRs are calculated on each component of the three-component streams. . . .	26

3.1	Workflow for DASstore: Distributed Acoustic Sensing (DAS) data is converted into Zarr or TileDB format in the MinIO object storage server. The computing server sends a data request to the MinIO object storage and loads the data directly into the memory. A conventional workflow would be downloading the HDF5 file from the raw data server and loading data to memory on the server where computing is performed (gray dashed arrows). The data attributes converted from the raw HDF5 files are stored in the new format, which in the Ocean Observatories Initiative (OOI) examples are: <code>RawData-2D</code> DAS data, <code>RawDataTime-1D</code> time axis, with four customized datasets: <code>GpBits</code> , <code>GpsStatus</code> , <code>PpsOffset</code> , and <code>SampleCount</code> . These datasets are specific to the OptaSense interrogator of the OOI data. The color version of this figure is available only in the electronic edition.	37
3.2	Chunking data in the three formats. The raw data is stored from the interrogators as a single file for each minute of recording. The Zarr schema separates each HDF5 dataset into a number of chunks; each chunk is a single object. The TileDB schema collects the entire data into a single array; the chunks are “tiles” and are concatenated in time. The red arrow represents the case for a “large-T” analysis with a subarray of the DAS cable. The green arrow represented a “large-N” analysis with a smaller time window. The color version of this figure is available only in the electronic edition.	39
3.3	Results of the reading test for (a) chunked HDF5, (b) Zarr and (c) TileDB format. For chunked HDF5, the compute server downloads the entire HDF5 file, reads a subarray of 5000 channels and then performs the feature extraction on each channel. The compute server requests a subarray of 5000 channels from the object storage and performs the same feature extraction workflow. The upper panels show the time to finish the test, and the lower panels show the total bytes received by the compute server. The line color indicates the chunk/tile size for Zarr/TileDB arrays. All lines are overlapping with each other for the chunked HDF5 test. The black dashed line in the lower section shows the theoretical data size after compression. The color version of this figure is available only in the electronic edition.	45
3.4	Results of the scaling reading test on Amazon Web Service (AWS). Panel (a) shows the reading time averaged over 5 runs. The dotted line indicates a strong scaling fitting, assuming the network transmission time stays constant. Panel (b) shows the total bytes received from the object storage averaged over 5 runs. The line color indicates the data format (Zarr and TileDB). The black dashed line in the right panel (b) shows the theoretical data size after compression. The color version of this figure is available only in the electronic edition.	46

3.5	(a) Map showing the location of the SeaDAS-N cable (solid line) connecting Seattle and Bothell campus of the University of Washington, United States. Only the red segment was interrogated in December 2022 for data collection. (b) One-hour SeaDAS-N raw data of all 2089 channels (in radians unit, a data volume of ~ 2.7 GB). The image starts 2022-12-15 at 13:30:00 UTC. The gray region indicates a one-minute data gap. The clipped data (channel ~ 1100 -1700) is from the segment where the cable is above ground. The color version of this figure is available only in the electronic edition.	47
3.6	An example of ambient noise cross-correlations functions of SeaDAS-N using channel 500 as the virtual source. The channel number is labeled on the left y-axis, and the distance along the cable is labeled on the right y-axis. The black dashed line indicates the location of the virtual source. The color version of this figure is available only in the electronic edition.	48
4.1	(a) Map of the Cook Inlet area in Alaska with two seafloor telecommunication cables shown: the southern span of Kodiak Kenai Fiber Link (KKFL-S) cable and the Terrestrial for Every Rural Region (TERRA) cable. 2615 earthquakes shown are selected from the USGS catalog between 1 July and 31 December 2023. The solid black lines mark the depth contour of the subduction slab with a 20 km interval ⁹⁷ . (b) A 10-minute recording of the raw data after demeaning and normalization on the KKFL-S cable, with corresponding water depth shown on the right and power spectrum in (c). Channel numbering starts from the landing station in Homer, Alaska. Ocean signals dominate the spectrum at about 10 s. The instrumental noise, a common mode, is clearly visible as a long-period vertical band observable along the whole cable and excessively dominating the data on the land cable segments (channel number smaller than ~ 100).	56
4.2	Structure of the wavefield reconstruction model, SHallow REcurrent Decoder (SHRED ³⁰³). Two LSTMs (long-short-term memory) are stacked together as the input branch. The shallow decoder is a three-layer, densely connected multi-layer perceptron (MLP) branch. The LSTM branch maps the time series from sparse DAS channels into a latent state with a fixed dimension. The latent state is then projected to the reconstructed wavefield at all N channels through the shallow decoder network. Note that time series at and before time T_0 is used for input, while only wavefield at time T_0 is reconstructed.	58

4.3	(a-c) A 10-minute example of raw, reconstructed, and residual data from the Cook Inlet DAS experiment on the KKFL-S cable sampling at 25 Hz. The same color scale is used for raw and reconstructed data, while the residual is boosted by a factor of 2. (d-e) The power spectrum of the raw data (black solid line) and reconstruction (red solid line) across the cables, with six selected channels shown in (i-k), suggesting a well-reconstruction in the frequency domain below 0.15 Hz.	60
4.4	(a-b) A 10-minute example of raw and reconstruction data from the Cook Inlet DAS experiment on the KKFL-S cable sampling at 25 Hz. Data from a segment of 2000 channels are shown. The corresponding data in the frequency-wavenumber (F-K) domain is shown in (c) and (d), respectively. The dispersion relation for ocean waves under an intermediate water depth setting ($h = 80$ m) is plotted as red dashed lines. .	63
4.5	Greedy algorithm tests: DAS data compression and reconstruction using Implicit Neural Representation (INR) models: Sinusoidal Representation Network (SIREN) and Random Fourier Feature Network (RFFN) are tested using a 10-minute data from the Cook Inlet DAS experiment (Figure 4.3). Reconstructive waveforms and their residuals are shown in (a,c) and (b,d), respectively.	65
4.6	(a-c) Raw, reconstructed, and residual data of a 40 s DAS recording from the Cook Inlet experiment. The selected window contains P- and S-wave arrivals from a regional M _{3.5} earthquake on 29 December 2023 (USGS event ID <code>ak023godcr3i</code>). Dashed lines show the theoretical arrival time calculated from Tau-P using the catalog hypocenter location. (d-f) With the noise (gray), P- (blue), and S-wave (red) onset selected, we measure the kurtosis for each data on this shorter time window. (g-h) The improvement of peak kurtosis contrast in the P- and S-wave window before and after removing the reconstruction. As each circle corresponds to a single event sized by its magnitude, 48.8% and 76.3% events show an enhanced arrival on P- and S-wave, respectively. The aforementioned event is filled with stars.	67
4.7	A 50 s residual DAS recording after removing the reconstructed wavefields from the raw data. The selected window contains P- and S-wave arrivals from a regional M _{2.0} earthquake on 14 July 2023 (USGS event ID <code>ak0238yvmdxr</code>). A band-pass filter is applied for illustration purposes only, enhancing the earthquake onset. We use the ensemble learning earthquake phase picker (ELEP ³²³) to pick the arrival time on the unfiltered residual data, marked by the blue and red crosses. Dashed lines show the theoretical arrival time calculated from Tau-P using the catalog hypocenter location and IASP-91 reference velocity model. The inset shows the difference between the ELEP picks (T_{ML}) and the theoretical arrival time (T_{1D}) for P- and S-wave onset. The mean absolute difference, the standard deviation as the uncertainties, and the mean value of the difference are reported in the upper right corner.	70

4.8	Correlated wavefield and dispersion analysis. Cross-channel correlation on (a) raw, (c) reconstructed, and (e) residual data. Correlation functions are calculated over 100 continuous channels and are stacked linearly over one hour, with slant stacked phase velocity dispersion map shown on the right, respectively. All cross-correlation functions and dispersion maps share the same color range, respectively, highlighting little leakage of coherent seismic ambient wavefield in the reconstruction. The dispersion suggests a highly dispersive fundamental-mode surface wave on this cable segment.	71
5.1	Map of stations, displayed as triangles and color coded according to the Data availability in station-years included in EarthScope, NCEDC, and SCEDC archives. Panel (a) shows the distribution of global station data availability. Panel (b) shows a detailed view of stations in the United States, while Panel (c) shows data availability in California.	77
5.2	The scalable cloud-native workflow for seismic phase picking. Containerized jobs are submitted to AWS Batch, which loads miniSEED seismic waveforms directly from AWS S3 buckets. Phase arrivals were identified with PhaseNet through the SeisBench implementation. A DocumentDB cluster is employed to store job metadata, picks, and checkpoints. Finally, an EC2 instance is used to provide a public database query service.	79
5.3	Detailed job run history for the EarthScope, NCEDC, and SCEDC dataset. Panel (a) shows the Job ID as a function of time, color-coded by year the data was recorded, for the EarthScope dataset. Panel (b) shows the job progression for the NCEDC and SCEDC datasets. Panels (c) and (d) show the pending jobs as a function of time. Panels (e) and (f) show the running jobs as a function of time. The horizontal dashed line represents the job quota.	82
5.4	Daily picks for selected stations. Stations are indicated by triangles on the central map, annotated with location and channel codes. For each of the ten example stations, the detail plots show a time series of the number of picks per day and a 28-day moving average in red.	83
5.5	Omori-type decays of the number of picks per day (blue) and the number of events per day (red) after the mainshock. We use picks at the reference station provided in the figure title. Event counts for the Illapel earthquake are from the Chilean Seismic Network (CSN) catalog, for the other two examples from the International Seismological Centre (ISC) and United States Geological Survey (USGS) catalog. In each case, we count all events with at most a 1.5-degree difference in latitude and longitude from the reference station.	84

- 6.1 (a) Three-dimensional shear wave velocity (V_s) structure of the simulation domain ($D_x = D_y = D_z = 9.6$ km). The red outline marks the surface observation area where virtual sensors record ground motion. (b) Example surface wavefield snapshots from a single simulation (sample index 100015). The nine panels show the three-component (East-West, North-South, and Vertical) displacement (cm), velocity (cm/s), and acceleration (m/s^2) field snapshot, with the projected source location marked by the gray star. The right panels show the corresponding Peak Ground Velocity (PGV, cm/s) and Peak Ground Acceleration (PGA, %g) maps at the same time stamp. 92
- 6.2 Architecture of the multi-stage wavefield reconstruction framework (EQ-SHRED and EQ-FLOW). Three-component velocity trajectories recorded at sparse sensors over a lag window $[t_0, t_0 + N_{lag}]$ are encoded by a stacked GRU-based recurrent encoder into a shared latent space. In Stage 1A (wavefield reconstruction), MLP decoder branches map the latent space to the compressive space, which are then expanded to the full spatial wavefield through their respective left singular mode matrices as the low-resolution reconstructions. In Stage 1B (source location estimation), the Gaussian Mixture Model (GMM) decode the latent space into probabilistic distributions of the hypocenter locations ($\rho_{x_s}(\cdot)$, $\rho_{y_s}(\cdot)$, $\rho_{z_s}(\cdot)$). In Stage 2 (wavefield refinement), a flow matching conditioned U-Net^{2.2.4} takes the low-resolution wavefield reconstructions as conditioning inputs and generates high-resolution refinements. 94
- 6.3 Example of the randomized Singular Value Decomposition (SVD) applied to the velocity field (vertical component). The high-dimensional tensor $\mathbf{x}_{vel}^Z \in \mathbb{R}^{N_{sim} \times N_x \times N_y \times N_t}$ is decomposed into the left singular vectors $\mathbf{U} \in \mathbb{R}^{N_x \times N_y \times r}$ (spatial modes), the diagonal matrix of singular values $\mathbf{\Sigma} \in \mathbb{R}^{r \times r}$, and the right singular vectors $\mathbf{V}^T \in \mathbb{R}^{r \times N_{sim} \times N_t}$ (temporal modes). The center panel shows the singular value decay (red) and the relative information content (RIC; blue) as a function of the number of retained modes. The left and right panels show the 10 leading spatial and temporal modes, respectively. 97
- 6.4 Schematic of the EQ-FLOW generative refinement stage based on conditional flow matching. A Gaussian noise sample $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is progressively shifted to the target distribution p_1 of high-resolution wavefields $\mathbf{x}_1 = \mathbf{x}_{HR}$, constrained by the low-resolution wavefield $\mathbf{x}_{LR}$ reconstructed by EQ-SHRED from Stage 1A. Wavefield snapshots at two intermediate time steps ($\mathbf{x}_{\tau_1}$, $\mathbf{x}_{\tau_2}$) illustrate the gradual emergence of fine-scale wavefield features as the integrating path evolves from noise toward the high-resolution target. 100

6.5	Displacement wavefield reconstruction snapshot for a representative test event (sample index 100044). (a) Spatial maps of the displacement field (cm) for three components (East-West, North-South, Vertical) and for ground-truth simulation, Stage 2 refinement, and Stage 1A reconstruction, respectively. White triangles mark the 33 virtual sensor locations that are used as input. (b) Compressive state amplitudes for modes 1–100 per component, comparing ground-truth (red), Stage 1A reconstruction (light blue), and Stage 2 refinement (dark blue). Same comparison for (c) modes 101–200, and (d) vector wavenumber power spectral density (PSD) as a function of wavenumber k (km^{-1}).	104
6.6	Velocity wavefield reconstruction snapshot for the same representative test event as Figure 6.5. (a) Spatial maps of the velocity field (m/s) for three components (East-West, North-South, Vertical) and for ground-truth, Stage 2 refinement, and Stage 1A reconstruction, respectively. White triangles mark the 33 virtual sensor locations that are used as input. (b) Compressive state amplitudes for modes 1–100 per component, comparing ground-truth (red), Stage 1A reconstruction (light blue), and Stage 2 refinement (dark blue). Same comparison for (c) modes 101–1000, and (d) vector wavenumber power spectral density (PSD) as a function of wavenumber k (km^{-1}).	104
6.7	Peak Ground Velocity (PGV) and Peak Ground Acceleration (PGA) reconstruction snapshot for the same representative test event as Figure 6.5. (a) Spatial maps of the PGV field (cm/s) and PGA field (% g) showing ground-truth, Stage 2 refinement, and Stage 1A reconstruction, respectively. Gray triangles mark the 33 virtual sensor locations that are used as input. (b) Compressive state amplitudes for modes 1–100, comparing ground-truth (red), Stage 1A reconstruction (light blue), and Stage 2 refinement (dark blue). Same comparison for (c) modes 101–200, and (d) vector wavenumber power spectral density (PSD) as a function of wavenumber k (km^{-1}).	106
6.8	Source location estimation results for a representative test event (sample index 100035; see panel (d)) and across the held-out test set. (a) Time-evolving probability density maps for each hypocenter location axis (x_s, y_s, z_s). The solid red vertical line marks the time when the event is first detected by a sensor. The white dashed horizontal line marks the ground-truth source location, and the white cross marks the first model prediction with a minimum probability density of 5, at the time indicated by the red dashed vertical line. (b) Velocity waveforms (vertical component) from the selected sensors, ordered by their hypocenter distance, with the first arrivals marked by the red triangles. (c) Response time as a function of ground-truth source location for all test events. Each dot represents one event, with the red curve showing the running mean value. (d) Wavefield snapshot of this representative event showing the 33 virtual sensor locations (triangles), the source location (gray star), and the corresponding source parameters used for the simulation.	107

7.1	Evolution of cloud computing in seismology: before 2020, most computational workflows involved downloading data from the Incorporated Research Institutions for Seismology Data Management Center (IRIS DMC) and other data centers and working locally. Between 2020 and 2025, seismologists have investigated the use of elastic computing by pulling data from existing archives and processing directly on the cloud (e.g., MacCarthy et al. ¹⁵²). At the same time, two regional seismic networks copied their archives of earthquake catalogs and seismic waveforms to Amazon Web Services (AWS): the Southern California Seismic Network (SCSN) and the Northern California Seismic Network (NCSN). Since 2025, the EarthScope Consortium has migrated its petabyte-scale archive to the cloud, enabling researchers to pull and compute directly on the cloud.	113
7.2	Station map of cloud-hosted data: A total of about 1.3 PBs of seismometer miniSEED data is hosted on Amazon Web Services (AWS) cloud storage: the EarthScope Consortium seismic data archive, and the Northern and Southern California Earthquake Data Center. Each triangle indicates a seismic station and is color-coded by its data availability.	115
7.3	Two canonical workflows in seismology: Event-based seismology that reads large volumes of seismic data (high I – high Input) but outputs low volumes of data in databases (low O – low Output), and structure-based seismology that reads large volumes of seismic data (high I) and outputs large volumes of seismic data (high O). The first workflow outlines the basic steps in generating a seismic event catalog. The second workflow describes how to extract seismic properties of the subsurface with ambient field seismology, which generates high data volumes of ambient noise cross-correlations.	124
7.4	Ambient Noise Cross Correlation using 1 year of data from the Southern California Earthquake Data Center: all data are publicly available in the SCEDC AWS cloud storage (s3://scedc-pds). The Z-Z component cross-correlation functions are bandpass filtered between 1 and 10 seconds.	127
7.5	Evolution of job statistics in the 2025 Data Mine experiment: top panel has a line between the start and end time of each job that ran on Fargate. Each year’s worth of data is sent manually as a separate batch of jobs, color-coded for EarthScope. The middle panel shows the number of pending jobs in the Fargate queue. Jobs enter the queue and are scheduled to run until the quota of 1500 jobs is reached. The dips in the number of active jobs are attributed to our slower job orchestration. One that may be attributed to SPOT’s intermitency is at hour 60. Upon success, the experiment on the NCEDC and SCEDC went smoothly, maximizing the number of jobs until completion.	130
B.1	Number of streams from the ComCat event catalog per station. The red dashed polygon denotes the authoritative region boundaries of PNSN.	152

B.2	Number of streams from the Exotic event catalog per station. The red dashed polygon denotes the authoritative region boundaries of PNSN. Note that 96% of the exotic events are surface events.	152
B.3	Example of a stream with a step offset from short-period EH channel. FDSN network and station code, ComCat event ID, and source origin time are labeled on the top. . .	153
B.4	Example of a stream with a step offset from broad-band HH channel. FDSN network and station code, ComCat event ID, and source origin time are labeled on the top. . .	153
B.5	Histogram of P- and S-wave picking residuals ($t_{ML} - t_{PNSN}$) from the benchmark test data set of strong motion channels. The number in the upper right corner of each subplot shows the mean absolute error (MAE), the root-mean-square error (RMSE) as the uncertainties to the MAE, and the picking completeness in percentage with respect to the ground truth. The PNW-retrained Earthquake Transformer outperforms than other four pre-trained models from SeisBench ³⁰⁷ in both picking accuracy and detecting completeness.	154
B.6	Fitting of Gutenberg-Richter (GR) power law distribution of magnitudes (solid lines) with 3 years of earthquake events cataloged by PNSN (black dots). The plot indicates that the minimum magnitude of completeness is around 2.	154
B.7	Location of the events with location horizontal uncertainty larger than 20 km. The red polygon denotes the authoritative region boundaries of PNSN.	155
B.8	Location of the events with location horizontal uncertainty larger than 20 km. The red polygon denotes the authoritative region boundaries of PNSN.	155
B.9	Number of picked P-wave polarity as a function of time. The red line marks the positive-negative ratio of the P-wave polarities.	156
C.1	Loss function during the SHallow REcurrent Decoder (SHRED) model training process. Mean-Squared-Error (MSE) is used to measure the misfit between the reconstructed and the raw data.	158
C.2	A 100 s long example of the raw, reconstructed, and residual data from the Cook Inlet DAS experiment on the KKFL-S cable sampling at 25 Hz. Figures show the arrivals from an M1.9 earthquake near Anchor Point, Alaska, on 22 July 2023 (USGS event ID ak0239btba15). The amplitude coefficient b of this event is estimated to be 0.704. The same color scale is used for raw and reconstruction data, while the residual is boosted by a factor of two.	158

C.3	A 100 s long example of the raw, reconstructed, and residual data from the Cook Inlet DAS experiment on the KKFL-S cable sampling at 25 Hz. Figures show the arrivals from an Mww7.2 earthquake near Sand Point, Alaska, on 16 July 2023 (USGS event ID at00rxvmw1). The amplitude coefficient b of this event is estimated to be 0.643. The same color scale is used for raw and reconstruction data, while the residual is boosted by a factor of two.	159
C.4	(a-c) A 10-minute long example of the raw, reconstructed, and residual data from the Cook Inlet DAS experiment on the TERRA cable sampling at 25 Hz. The selected window does not contain earthquakes according to the USGS catalog. The same color scale is used for raw and reconstruction data, while the residual is boosted by a factor of two. (d-e) The power spectrum of the raw data (black solid line) and reconstruction (red solid line) across the cables, with six selected channels shown in (i-k).	160
C.5	Generalizability test of the SHRED model. (a) An 80 s long example of the raw data from the Cook Inlet DAS experiment on the TERRA cable sampling at 25 Hz. Figures show the reconstruction results of (b) the model trained by data from KKFL-S cable, (c) the model in (b) but transfer-learned with data from TERRA cable, and (d) the model trained by data from TERRA cable.	161
C.6	(a-b) A 10-minute example of raw and reconstruction data from the Cook Inlet DAS experiment on the KKFL-S cable sampling at 25 Hz. The SHallow REcurrent Decoder (SHRED) model is trained using 101 out of 1000 decimated channels. Data from a segment of 2000 channels are shown. The corresponding data in the frequency-wavenumber (F-K) domain is shown in (c) and (d), respectively. Dispersion relation for ocean wave under an intermediate water depth setting ($h = 80\text{ m}$) is plotted as red dashed lines. The first overtone is modeled as dashed gray lines.	162
C.7	(a-b) A 10-minute example of raw and reconstruction data from the Cook Inlet DAS experiment on the KKFL-S cable sampling at 25 Hz. The SHallow REcurrent Decoder (SHRED) model is trained using 151 out of 1000 decimated channels. Data from a segment of 2000 channels are shown. The corresponding data in the frequency-wavenumber (F-K) domain is shown in (c) and (d), respectively. Dispersion relation for ocean wave under an intermediate water depth setting ($h = 80\text{ m}$) is plotted as red dashed lines. The first overtone is modeled as dashed gray lines.	163

C.8	A 50 s raw DAS recording from the Cook Inlet experiment. The selected window contains P- and S-wave arrivals from a regional M2.0 earthquake on 14 July 2023 (USGS event ID ak0238yvmdxr). A band-pass filter is applied for illustration purposes only, enhancing the earthquake onset. We use the EQCCT to pick the arrival time on the unfiltered residual data, marked by the blue and red crosses. Dashed lines show the theoretical arrival time calculated from Tau-P using the catalog hypocenter location and IASP-91 reference velocity model. The inset shows the difference between the ML picks (T_{ML}) and the theoretical arrival time (T_{1D}) for P- and S-wave onset. The mean absolute difference, the standard deviation as the uncertainties, and the mean value of the difference are reported in the upper right corner.	164
C.9	A 50 s raw DAS recording from the Cook Inlet experiment. The selected window contains P- and S-wave arrivals from a regional M2.0 earthquake on 14 July 2023 (USGS event ID ak0238yvmdxr). A band-pass filter is applied for illustration purposes only, enhancing the earthquake onset. We use the ensemble learning earthquake phase picker (ELEP) to pick the arrival time on the unfiltered raw data, marked by the blue and red crosses. Dashed lines show the theoretical arrival time calculated from Tau-P using the catalog hypocenter location and IASP-91 reference velocity model. The inset shows the difference between the ML picks (T_{ML}) and the theoretical arrival time (T_{1D}) for P- and S-wave onset. The mean absolute difference, the standard deviation as the uncertainties, and the mean value of the difference are reported in the upper right corner.	165
D.1	Number of phase picks grouped by network codes, with the top 40 networks shown. Bars are color-coded by the hosting data center: EarthScope (red), SCEDC (orange), and NCEDC (blue).	166
E.1	Training (red line) and testing (dashed blue line) loss for Stage 1A wavefield reconstruction (epoch 1-150) and Stage 1B source location estimation (epoch 151-300).	168
E.2	Training (red line) and testing (dashed blue line) loss for Stage 2 wavefield refinement of (a) displacement, (b) velocity, (c) PGV, and (d) PGA field, respectively.	169

Listing of Tables

2.1	Number of included ComCat events in each magnitude range. The magnitude used here includes duration (Md), local (Ml), and hand (Mh) magnitude. The number of streams includes both velocity and acceleration channels. Also provided is the number of included events as a percentage of downloaded ComCat events. Numbers in the parentheses show the events and streams from the 20 December 2022 Northern California earthquake sequence and are not included in the total number of events/streams.	15
2.2	Attributes in the metadata file. Some source attributes are not available for exotic events.	21
2.3	Description of network FDSN code and their references. Networks annotated by an asterisk mark (*) are maintained by the PNSN. The number of streams shown for each network is from ComCat events, exotic events, and noise, respectively. PB and HW network does not have a registered FDSN network DOI.	28
3.1	Servers used in the tests: role, network, CPU cores, RAM, and primary storage. These specifications emulate a researcher-level DAS setup and include an example AWS EC2 instance for cloud benchmarking.	40
3.2	OOI DAS data conversion from HDF5 into Zarr and TileDB formats. HDF5 files are converted directly to the object storage from the raw data server. The green numbers highlight the writing time shorter than the data duration, indicating the configurations that would allow real-time conversion.	42
5.1	DocumentDB Collections and Schema Summary.	80
B.1	The event types and labels used in PNSN’s ANSS Quake Monitoring System (AQMS). Several event types are merged into one when being reported to the ComCat.	151
B.2	The events selected from the 20 December 2022 Northern California earthquake sequence that are included in the data set. Source origin time, event ID, and magnitude are reported by the California Integrated Seismic Network (CISN), with corresponding PNSN event ID.	151
C.1	Performance comparison of SHRED under different configurations on a 10-minute long KKFL-S test recording. N_{LSTM} : the number of the LSTM layer. N_{latent} : the dimension of the latent space. N_{hidden} : the number of neurons in the hidden layer.	157

Acknowledgments

First and foremost, I would like to express my deepest gratitude to my advisor, Marine Denolle, for her continuous support, guidance, and encouragement throughout my Ph.D. journey. I got to know Marine when I was doing a summer internship at Harvard in 2019. Being both very shy and uncertain about my academic interests, I did not reach out to her or simply say hi. I was finally ready to apply for graduate school in 2020 and developed further interests in high-performance computing and seismology. Luckily, Marine admitted me to the ESS program, though I still knew nothing about the Pacific Northwest (e.g., I thought “PNW” was a typo because NW Pacific Ocean must be near Kamchatka?). I always received full support from Marine: I never got a negative response when I wanted to attend conferences and workshops, apply for funds, do fieldwork or internships, take a break, or even renew my visa in Mexico and Canada. No matter how busy she is, Marine is always happy to meet with me to discuss research progress or simply brainstorm. She has been a consistent source of encouragement to stay curious and passionate about scientific questions, to be rigorous and critical in my work, to think big, and to seize every opportunity to present. She shaped who I am as a researcher, and I continue to learn from her.

I am thankful to my supervisory committee members, Brad Lipovsky, Renate Hartog, Nathan Kutz, and Brett Maurer. Each of them brought a distinct perspective to my work, from fiber-optic sensing to ground motion and geotechnics, and from earthquake monitoring to data-driven modeling. I have learned a lot from them during every step of my Ph.D. journey. Much of the work presented in this dissertation would not be possible without their support and guidance. I am also deeply inspired by their dedication in their respective fields and their passion for scientific rigor.

I am grateful to have shared ATG 202 with fellow ESS students who made the time in the office so enjoyable and inspiring. I have enjoyed chatting with them about any random topic. In particular, Barrett Johnson sat back-to-back with me for three years, and we had many conversations about research, life, and everything in between, until he moved downstairs to a PNSN office and temporarily lost his window access. I teamed up with Barrett during my first seismology fieldwork in Tillamook, Oregon: it was an unforgettable experience. Manuela Köpfler goes to the office almost every day, while I am only in the office sporadically in the afternoon. Beyond random chats, there were many times I sold NoisePy to her for ambient noise cross-correlation on DAS and nodes, and we would sit together coding to build new features for NoisePy. Emily Wilbur always brings in snacks (while Manuela and I are perhaps the major consumers of *Nerds*) and remembers everyone’s birthday with cakes to celebrate. Thank you all for your company.

I have enjoyed working with current and former Denolle Lab, Fiber Lab, and Pacific Northwest Seismic Network colleagues, who have been a great source for discussion and seeking feedback. Congcong Yuan and Qibin Shi have provided invaluable suggestions on my research and career development. As Congcong wrote in his dissertation, “This is the dedication”—and from my perspective, they both demonstrate full dedication to scientific inquiry. I also enjoyed the GCI DAS fieldwork in Homer,

Alaska, with Qibin, Verónica Gaete-Elgueta, and Ethan Williams: during a postdoc interview, I did not hesitate to say that the GCI DAS project was my favorite. Manuela and Dominik Gräff are both great post-lunch coffeemates: we discussed random topics including research, news, culture, and most recently, how to write a good acknowledgment. The duty seismologist team, in particular Paul Bodin, Alex Hutko, Amy Wright, Steve Malone, and many others, have been great teachers, helping me learn how to read seismograms for earthquake monitoring.

I would like to thank my friends at UW Oceanography, Civil and Environmental Engineering, Computer Science and Engineering, and Applied Mathematics. Through you, I have broadened my interests across many fields and discovered new possibilities for my career and interdisciplinary research. I am also grateful to the staff and researchers at the eScience Institute for building a dynamic data science community where I always felt welcome. I have met many friends through seminars and the incubator projects that eScience has organized. I have learned so much from Scott Henderson and Carlos Garcia Jurado Suarez, who have constantly and generously shared the best practices of data science and code development. There are also many friends who have offered me enormous personal support: thank you all for the joy you have brought to my life.

Above all, I want to thank my family for their unconditional support. I have been given complete freedom to pursue my interests, to make decisions, and to become the person I want to be. I also want to thank my dearest girlfriend Sihan (Sarah) Wang for her love, care, company, and support. I am so lucky to have met you, to share a wonderful life with you, and to have you by my side.

1

Introduction

Seismology, rooted in the first principles of elasticity, is becoming an increasingly data-driven subdiscipline of geoscience. While seismometers continuously measure ground motion, for a long time, digital data were archived and made available only as event segments at major data centers. With advances in computer science and engineering, continuous seismic data archiving became common practice for most seismic networks in the early 2000s. Driven by the rapid expansion of regional seismic networks, dense nodal deployments, ocean-bottom seismometers, and trends toward higher sampling rates and co-located instruments, continuous seismic archives are growing rapidly¹¹⁰. As of 2024, data hosted by the EarthScope Consortium (formerly the Incorporated Research Institutions for Seismology (IRIS))

Data Management Center (DMC) have exceeded 1 petabyte¹⁹². Additionally, the emergence of Distributed Acoustic Sensing (DAS) technology has further expanded seismic data collection in various environments by utilizing fiber-optic cables as dense seismic arrays³²⁷. This proliferation has opened new opportunities to leverage the full potential of continuous seismic recordings, further advancing our understanding of the Earth's structure and dynamics¹⁶.

One major motivation for expanding seismic observation is to enhance earthquake monitoring across scales. Earthquake monitoring serves two complementary roles. In near-real-time, it is critical for earthquake early warning (EEW) systems that provide rapid alerts before strong shaking arrives, guiding automated protective actions that can potentially save lives and protect critical urban infrastructure^{10,11}. The same system works as a broadband environmental sensor, enabling the real-time detection and characterization of a wide range of non-tectonic seismic sources such as quarry blasts and explosions¹², landslides¹⁹, and glacial processes¹⁹⁶. Beyond real-time applications, complete and high-resolution earthquake catalogs resulting from retrospective data analysis fuel a broad range of Earth science research, including seismic tomography^{328,329}, seismic hazard analysis¹⁷⁵, and volcano monitoring^{135,170}.

However, fully utilizing the large volume of seismic data to further enhance earthquake monitoring comes with notable challenges. First, processing continuous seismic recordings requires developing algorithms capable of efficiently identifying useful signals from seismic ambient noise. Second, many scientifically important regions remain beyond the reach of dense seismic stations, either due to logistical inaccessibility or prohibitive deployment costs, leaving critical gaps in coverage. Third, the sheer scale of modern seismic datasets demands advanced computing infrastructure capable of supporting data-intensive workflows that exceed the capacity of conventional on-premises systems. In this dissertation, I address these challenges through three recent innovations in seismology: machine learning, distributed acoustic sensing, and cloud computing. While individually distinct, these innovations are complementary and will collectively enable a scalable, data-driven earthquake monitoring framework.

Machine Learning (ML) has increasingly benefited research in geoscience and seismology^{26,171}. The recent surge of ML applications is fueled by the need to automate feature extraction and pattern recognition in ground-motion time series data. ML has been widely applied across tasks such as earthquake detection and phase picking^{174,203,225,259,337}, phase association^{161,177,227,340}, catalog building work-

flows^{285,320,330}, event classification^{131,239}, denoising^{246,277,338}, structural imaging and inversion^{28,258,312}, source characterization^{128,178,270}, and earthquake early warning^{47,132,179}. While the growth of ML research is transforming scientific understanding of the Earth's dynamics, it is still necessary to further develop and refine machine learning algorithms, especially with more high-quality data and computational resources becoming available.

Distributed Acoustic Sensing (DAS), as an emerging technology for sensing Earth vibrations on optical fibers, has revolutionized seismic observation. This technology utilizes repeated laser pulses along optical fibers to measure phase changes of light that are scattered from imperfections along the cable, which are proportional to the rate of extensional strain along the axis of the fiber. DAS has dramatically expanded the capability of dense seismic observation with reduced field deployment effort and has been deployed across a vast range of applications, including detecting tectonic faults¹⁴⁰, subsurface imaging^{18,41,164,280,313,314}, urban seismic site characterization²⁵³, cataloging regional earthquakes^{137,185,341}, aquifer and hydrological monitoring²²¹, cryosphere and permafrost sensing⁴², ocean current and surface gravity wave monitoring³⁰², marine acoustic and ship-noise detection^{64,299}, and monitoring vehicular traffic and infrastructure state of health^{141,289}. DAS has the potential to transform geophysical sensing and push the frontiers of seismological research, while efforts are still needed to further scale up DAS for real-time seismic monitoring across diverse environments.

Cloud computing has revolutionized scientific computing by making large datasets and on-demand computing resources more accessible^{84,192}. Cloud adoption in seismology spans a broad range of applications, including cloud-hosted seismic data archives at major data centers (e.g., Southern California Earthquake Data Center³²¹), cloud-native and cloud-optimized data formats and storage³, scalable streaming and batch data processing^{91,152,342}, large-scale machine learning training and dataset dissemination³⁴³, and cloud-based educational platforms and tutorials^{59,127}. Cloud computing has the potential to transform how seismologists curate, share, and process large-scale datasets. However, many challenges remain for cloud seismology, such as designing cloud-native seismological workflows, developing community standards for cloud-hosted data and metadata, and making cloud resources more accessible to the broader research community¹²⁴.

This dissertation addresses the aforementioned challenges and opportunities through six research

chapters spanning the full seismic data lifecycle, from acquisition and storage to large-scale analysis, with an emphasis on scaling data-driven earthquake monitoring. The remainder of this dissertation is organized as follows. In Chapter 2, I present a curated seismic dataset compiled from 21 years of Pacific Northwest Seismic Network (PNSN) recordings, providing a platform for training and benchmarking machine learning models across a broad range of seismological tasks. In Chapter 3, I develop a cloud-native object storage system that leverages cloud-optimized formats such as Zarr and TileDB to enable efficient storage and processing of DAS data. In Chapter 4, I develop a machine learning algorithm for DAS wavefield reconstruction and lossy compression, applied to data from Cook Inlet, Alaska to enhance earthquake detection at the edge. In Chapter 5, I present a scalable cloud-native workflow for petabyte-scale event detection and phase picking, processing 1.3 PB of continuous seismic recordings spanning two decades. In Chapter 6, I develop EQ-SHRED and EQ-FLOW, a multi-stage machine learning framework that reconstructs multi-physics earthquake ground motion fields and estimates probabilistic hypocenter locations from sparse sensor observations, with potential applications to earthquake early warning. Finally, in Chapter 7, I provide a comprehensive review of cloud computing and storage in seismology, showcasing representative big-data workflows and discussing opportunities, costs, and challenges of cloud adoption.

*This chapter was published as Ni, Y., Hutko, A., Skene, K.,
Denolle, M. A., Malone, S., Bodin, P., Hartog, J. R., & Wright,
A. (2023). Curated Pacific Northwest AI-ready Seismic Dataset.
Seismica, 2 (1). doi: 10.26443/seismica.v2i1.368.*

2

Curated Pacific Northwest Seismic Dataset

2.1 Introduction

The Pacific Northwest (PNW) region of the United States is a dynamic tectonic plate boundary between the North American continental plate and the Juan de Fuca oceanic plate. The active margin between the two plates is a subduction zone that hosts a wide variety of earthquake behaviors: fast and large megathrust earthquakes³⁰⁵, intraslab earthquakes⁸³, crustal earthquakes⁸⁷, slow repeating earthquakes^{23,222,292}, tectonic tremor²⁹³, and low-frequency events². The PNW has over twenty active

volcanoes that have experienced eruptions in the historical record. The PNW has hundreds of glaciers in the Cascades, the Olympic Peninsula, and sitting atop the Cascade Volcanoes. Due to the active tectonics and the particular mid-latitude climate, the PNW also experiences hundreds of landslides every year¹⁴⁹. Such geohazards generate seismic waves that are well recorded^{13,14,102}.

The Pacific Northwest Seismic Network (PNSN) is the authoritative seismic network in the states of Washington and Oregon as part of the Advanced National Seismic System (ANSS), which is coordinated by the United States Geological Survey (USGS). PNSN started in 1969 with 5 seismometers and has more than 600 active seismic stations as of 1 November 2022. The authoritative boundaries of the seismic network are geographical (see Figure 2.3), but the Cascadia subduction zone is also active in Northern California and southern British Columbia^{65,66}. The longevity of the seismic records and the richness of the active geohazards in the PNW form a unique opportunity to explore a vast range of seismic signatures. Comprehensive investigations that relate seismic signature to specific geohazards^{14,33,76,102} benefit from curated datasets.

In recent years, machine learning has increasingly been used for applications in geosciences and seismology in particular. The rise of machine learning, especially deep learning, is largely due to the curation of several computer vision (e.g., ImageNet⁵⁸) and natural language processing (e.g., GLUE²⁸⁶) datasets. There is a clear surge of machine-learning workflows in seismological research^{26,122,156,171} that is driven by the high dimensionality of seismological data, the dramatic growth in data volumes¹¹⁰, and the effort by the community to curate seismic datasets. There exists today several curated datasets that have become standards for machine-learning seismological research: STEAD (a dataset of local and regional earthquakes and high-frequency noise recorded globally, Mousavi et al.¹⁷²), INSTANCE (Italian seismic dataset for machine learning, Michelini et al.¹⁶⁵), ETHZ (Eidgenössische Technische Hochschule Zürich, Woollam et al.³⁰⁷), SCEDC (Southern California Earthquake Data Center, SCEDC²³²), and Iquique (a data collection of subduction-zone earthquakes and regional recordings, Woollam et al.³⁰⁶). These datasets contain earthquake and noise time series recorded by various seismometers. The typical data attributes are basic earthquake source and receiver characteristics, including locations, magnitudes, focal mechanisms, and waveforms. The majority of the earthquake sources in these datasets are of tectonic origins: transform plate boundaries such as in California, subduction zone, and intra-continental crustal

earthquakes^{165,306}. Such datasets are considered “AI-ready” since their data and attributes are packaged in data formats commonly used by the Machine Learning community.

Surface processes may also generate seismic waves. Environmental seismology is a blooming field that utilizes seismic waves to understand surface and environmental processes. There is a body of research done on the seismic signatures of landslides events^{43,103,310}, avalanche signals³³, and debris flows⁴³, most of which investigate specific case studies. Catalogs of such events are available in the Incorporated Research Institutions for Seismology (IRIS) Exotic Seismic Event Catalog (ESEC) (e.g., Allstadt et al.¹⁵, Bahavar et al.¹⁹, Collins et al.⁵¹); these refined and ground-truth catalogs only contain a few (~ 100) events.

Our study provides a novel curated AI-ready dataset of event and waveform data for a diverse range of short-duration seismic sources that include tectonic earthquakes, explosions, surface events such as ice/rock falls and avalanches, sonic booms, and thunderstorms. Not included are phenomena such as non-volcanic tremors or low amplitude low-frequency earthquakes (LFEs). We leverage the 21 years of data curation by the PNSN seismic analysts and researchers to measure the event P- and S-phase arrival times and other attributes. To enable optimal re-usability of our dataset for machine learning studies, we organized the dataset using the SeisBench data format³⁰⁷ to improve accessibility in the machine learning ecosystem. We acknowledge the accompanying human biases that often pollute AI-ready datasets²⁰¹ are well present in our catalog of event and waveform attributes. Some of these identified biases are discussed below and are obvious topics of future investigations.

2.2 Data Selection and Preparation

The PNSN has been monitoring the seismicity in the PNW since 1969. However, seismic waveform data from PNSN were recorded on film and paper until 1980, when digital data became available. From 1980 to 2002, event-triggered waveform data (often with a limited duration) were saved, but continuous archiving did not start until 2002.

For machine-learning applications, long seismic traces as input data are preferred to allow user flexibility when trimming and shifting the data in future investigations (e.g., data augmentation³³⁹). The data must also have the same dimensions, i.e., the same number of samples. To get waveforms that

are long enough (i.e., 150 seconds and longer in this study), we start the curation when continuous data are available from IRIS Data Management Center (DMC) since 2002. The drawback of this choice is that it excludes the largest tectonic earthquakes in the region because they occurred before 2002 (e.g., Nisqually Earthquake of 28 February 2001). In addition, we require that both a P-wave arrival time and an S-wave arrival time information are available for the same station for each event. This requirement removes some of the smaller, older earthquakes for which no S-picks were available. In the context of AI-ready datasets, the associated metadata (labels or attributes) include event-derived parameters, station parameters, and waveform parameters. We use the SeisBench metadata format: Table 2.2 lists the attributes that we associate with each set of waveforms.

2.2.1 Event Parameters

The detection of new events is both automated and manually reviewed by the regional seismic network staff. The PNSN monitors and reports on the seismicity in the region using data from seismic stations (Figures B.1 and B.2). A trigger at a station occurs when the short-term-average-long-time-average of the seismic data (STA/LTA⁹) exceeds a threshold. When a few stations from a designated geospatial group of seismic stations, called a subnet, experience a trigger, events are automatically saved. The PNSN analysts review all automatically detected events and remove erroneous ones by visual inspection of the event waveforms, a process they refer to as “trigger review”. Teleseisms are also identified but not further processed.

If the waveform has a clear but emergent signal, does not contain distinct P and S arrivals, and the frequency content is relatively low, the PNSN assigns a “surface event” label (su) to the source type. Most surface events are “ice” quakes or avalanches associated with glaciers in the Cascades and on the volcanoes; however, some may be debris flows or rock falls. Other non-earthquake phenomena occasionally saved by analysts are recordings of sonic booms, thunderstorms, and other “interesting” events. Such waveforms are picked at very few nearby stations (one or two), and we gather the phase pick information in a catalog that we refer to as the “Exotic Event” catalog.

Once the trigger review identifies an event as an actual earthquake, the PNSN analysts further process the data. First, the automated system picks the arrival times of seismic phases from the recorded

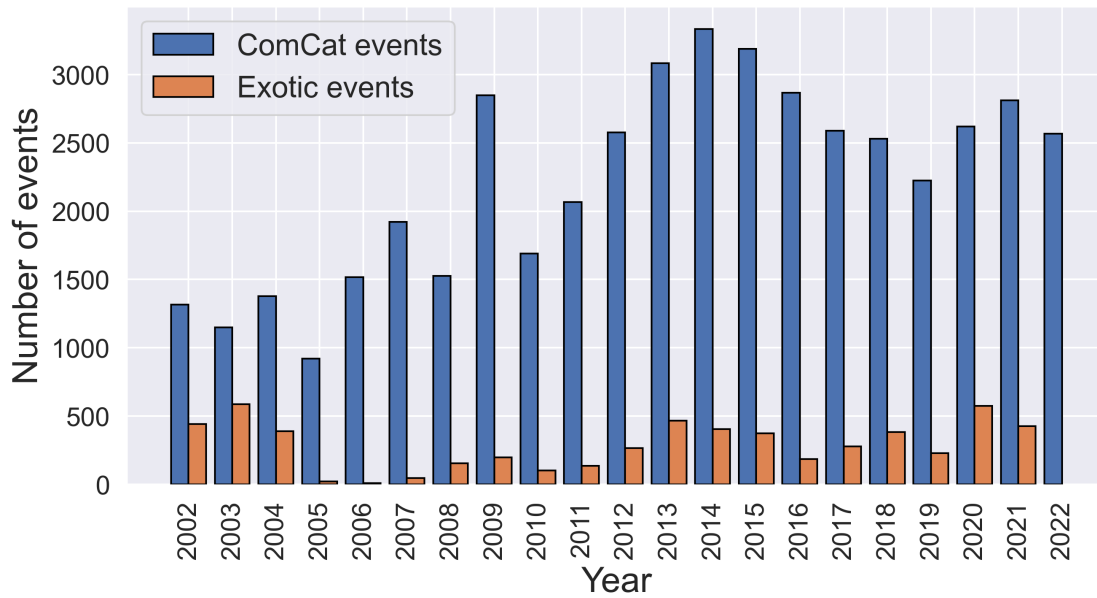


Figure 2.1: The event counts of ComCat and exotic catalog included in the AI-ready Pacific Northwest dataset as a function of time.

seismograms, which are one of the most important and primary data products extracted from the raw waveforms. The analyst reviews and modifies the picks.

Seismic phase picking is the cornerstone of seismological research. With accurate phase arrival information, the analysts can locate the event and estimate its origin time. At the PNSN, the first P- and S-waves are the phases picked for local and regional events. As a part of the PNSN’s ANSS Quake Monitoring System (AQMS), the network analysts use Jiggle, a graphical user interface in Java to pick arrivals, locate events, and recalculate magnitudes⁹⁴. The analysts will manually annotate the arrival time and estimate the uncertainties of their picks. The phase arrivals are only picked on a single component per station, with P-waves usually picked on vertical channels (Z component) and S-waves on horizontal channels (E/N or 1/2 components). When it is clear, the polarity (first motion is up–positive–, or down –negative–) of the P-phase is labeled by the analyst as well. Both acceleration and velocity channels are used for phase picking, although velocity channels are the most commonly used. The PNSN operates sites with both velocity channels (broad-band or short-period high-gain seismometers) and acceleration channels (low-gain accelerometers used for “strong motion” seismology). Velocity channels are preferred when both instrument types exist since they usually have a higher signal-to-noise

ratio than the strong-motion channel.

Additional earthquake characteristics may be obtained from the phase polarity and amplitudes, such as focal mechanisms and magnitudes. All event parameters are saved in PNSN’s AQMS database, and reasonably well-located earthquakes and explosions are reported to the ANSS Comprehensive Earthquake Catalog (ComCat²⁷⁴) via USGS Product Distribution Layer (PDL), the software-server infrastructure that all the ANSS regional networks use to distribute earthquake products. It is important to note that the combination of automated tools, which get updated through time, and manual intervention renders the event parameters not statistically stationary over time.

This study splits the PNW catalog into several datasets: one that has PNSN analyst-verified event attributes that were sent to the USGS, which we refer to as the “ComCat event” dataset, one that we refer to as the “exotic event” dataset and that has remained internal in the PNSN AQMS database, and one that focuses on the 2022 Northern California earthquake sequence. These datasets are packaged in different files because they have different window lengths and data attributes. We collect and organize the data from these. We show in Figure 2.1 the annual event counts for the two sets of events, ComCat and exotic, that are selected for the curated dataset. The temporal patterns ought not to be interpreted as changes in seismicity rate since there are systematic biases in the detection and labeling of the events through time, whether they are human (analyst) or instrumental (increased instrumental coverage).

ComCat Events

We query the ANSS ComCat and download 65,384 events with magnitudes greater than 0 from 1 January 2002 to 31 December 2022, which we refer to as “ComCat events”. We only select the events from ComCat sent by the PNSN, whose event ID has a “uw” prefix. The event metadata, including phase picks, are downloaded using libcomcat⁹⁸ and stored in the QuakeML format (v1.2²³⁵). The source type of these events are either earthquakes or explosions. The download contains 997,213 associated phase picks. Among these picks, 944,220 were made on velocity channels and only 52,982 (5.3%) on strong-motion channels. For single-channel stations where only the vertical channel (Z) exists (e.g., EHZ), S-waves were also picked only if the onsets were clear. The temporal evolution of the ComCat events reflects a combination of increased coverage and sensitivity of the seismometers. In 2009, a large

number of the cataloged events came from an intense swarm of earthquakes at Wooded Island in eastern Washington⁸⁶. As listed in Table 2.1, the number of events represented in our final curated dataset is less than what we originally downloaded due to data selection criteria described in Section 2.2.3.

Exotic Events

We also collect data from 5,657 events cataloged by the PNSN since 2002 that are neither labeled as earthquakes nor explosions. The exotic events are not incorporated in the ANSS ComCat and are only available through the PNSN’s ANSS Earthquake Monitoring System (AQMS) database. In this dataset, we include events that were labeled as “surface event”, “thunder”, “sonic boom”, and unfortunately a “plane crash” (a confirmed event near Whidbey Island, Washington, 3 March 2013). We refer to these events as “exotic events” herein. Figure 2.2 shows the number of events in each category for our final dataset.

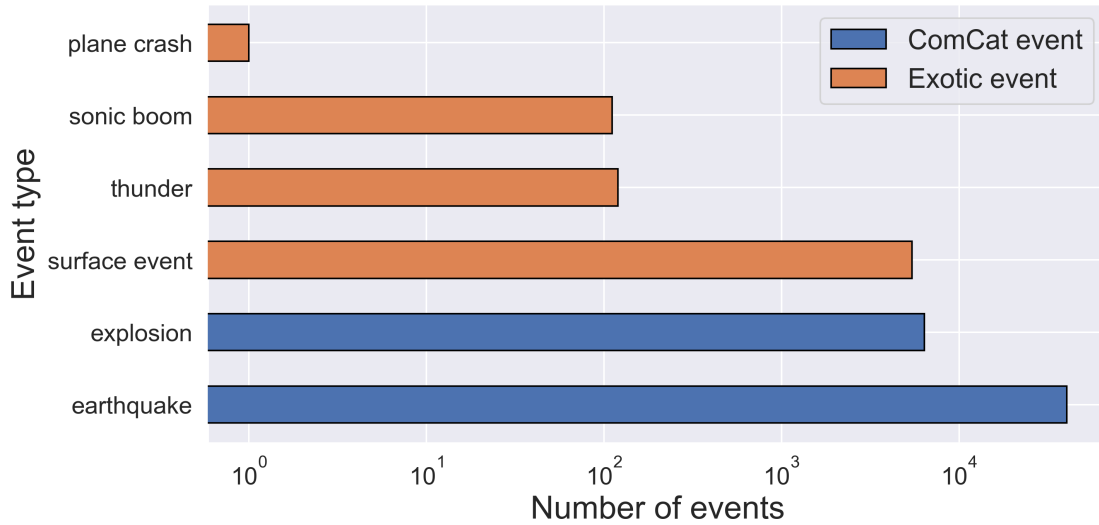


Figure 2.2: Number of events arranged by event type of the curated ComCat and Exotic event datasets.

The temporal evolution of the exotic event catalog depends on manual intervention by the analysts. Because non-tectonic earthquakes are not the priority of the PNSN, analysts only pick when time permits. Most of the labeled exotic events, such as surface events, are detected on well-instrumented volcanoes (see Figure B.2). The lower event count in the period 2005-2008 coincides with volcanic unrest at Mt. St. Helens, when the network was also desensitized during this period to the events around Mt.

St. Helens due to the intense rate of volcano-tectonic seismicity. It is quite possible that other surface events outside of the volcanoes are missing, due to having fewer stations elsewhere.

Most of the exotic events are small in magnitude and seismic amplitude and thus local to a few stations. Due to a lack of additional observation of the events (e.g., a ground truth imagery as done in the ESEC catalog), source characteristics such as the source origin time, location, and magnitude are not provided for these events.

2022 Northern California Ferndale Earthquake Sequence

We also include events associated with the 20 December 2022 M6.4 Ferndale (northern California) Earthquake. This sequence provided us with a rare opportunity to add labels for moderate-to-large earthquake sizes. These events are outside of the PNSN’s authoritative boundary and, thus are not routinely processed by the network. We select 20 events of $M \geq 3$ reported by the California Integrated Seismic Network (CISN) from that sequence and manually pick 609 P-wave arrivals. Table B.2 lists events included in the dataset.

2.2.2 Station Metadata

The station metadata describes the technical information necessary for seismic data processing and tracks the history of any metadata changes. The IRIS DMC stores station metadata as dataless SEED files, but they can be downloaded in the StationXML format from IRIS International Federation of Digital Seismograph Networks Web Service (FDSN-WS, <http://service.iris.edu/fdsnws>). The up-to-date station metadata we use is downloaded using ObsPy¹²⁵. These stations are either long-term installations maintained by a seismic network (e.g., UW²⁷³) or long-time experiments that lasts several years (e.g., US Transportable Array, FDSN code TA^{III}).

2.2.3 Event Waveforms

All digitized data from the PNSN are requested and downloaded through the IRIS FDSN-WS (<http://service.iris.edu/fdsnws>). In total, we download ~ 70 TB of continuous data in miniSEED from 1 January 2002 to 31 December 2022, which takes 2 months to complete. We first curate waveforms

from high-gain velocity seismometers and specific channels from short-period (EH?) and broad-band (either BH? or HH?) seismometers. We do not use the SL? and SH? channels since they are simply derived from EH? channel after low-pass filtering or down-sampling. We also include waveforms from strong-motion EN? stations separately since there are also picks made on these channels by the analysts. We do not correct for instrumental response and do not integrate the acceleration to velocity. All waveforms are resampled to 100 Hz from their original sampling rates, which may be 40 (most BH? channels) or 100 (most EH? and HH? channels). The resampling step is necessary for deep neural networks with fixed input sizes. We keep the data as is, even if it is clipped.

For each ComCat event, we only select the stations where both P- and S-wave are picked. We prepare 150-second data for ComCat events: the window starts 50 seconds before and ends 100 seconds after the source origin time (200 seconds after the origin time for the Northern California earthquake sequence). The same length of traces before this time window is curated as the noise waveforms. The reason for including so much noise window ahead of the origin time is to allow user flexibility when trimming and shifting the data in future investigations. In the ComCat events, less than 1% of the S-wave picks arrive later than 60 seconds after the origin time. Thus, most S-wave arrivals are included in the time window. Then, we apply a linear detrending. We also resample all waveforms to 100 Hz, which upsamples the broad-band BH? channels. Due to the small inaccuracy ($\sim 0.00008\%$) of the digitizer clock of the analog EHZ stations, the sampling rate at these stations shifts away from strictly 100 Hz. We correct this by resampling to 100 Hz. Gappy traces are discarded. Missing channels, for example, the vertical-component-only instruments (e.g., channel EHZ) are filled with zeroes to keep the consistency of a three-component stream (further detailed below). Picks are only done with data from a single instrument per site, even if a site may have several sensors. Therefore, each “stream” is independent of the other.

The PNSN operates seismic stations that are particularly remote. The transfer of data through telemetry sometimes leads to artifacts in the time series. Furthermore, the transition from triggered to continuous data was progressive. Sometimes, both triggered waveforms, which are detrended, and continuous data, which are unprocessed, are sent together: the triggered data overwrites the continuous data, creating a step in the data. These show in both short-period (EH?) and broad-band (BH? and

HH?) stations. For example, the time series may contain offsets that could be corrected in the future in the seismic archive at the IRIS DMC (see Figures B.3 and B.4).

The waveforms extracted for an exotic event are not aligned with the source origin time, which is mostly unknown. Instead, we align the waveforms by the phase picks provided by the analysts. The waveforms start 70 seconds before P-wave picks or 80 seconds before S-wave picks, whichever is available. Most exotic events have no picked S-waves, but if both P- and S-wave picks exist, the P-wave is prioritized to align the time window. The time window is 180 seconds long for all types of exotic events, given the occasional long duration and elongation (e.g., cigar-shaped waveforms¹⁵⁸) of the surface events. We follow the same data-curating process and formats as we process the ComCat events.

We also extract noise-only waveforms. These waveforms are extracted just ahead of the event waveforms. We selected high-gain velocity channels (EH?, HH?, and BH?) using a random selection. To further test if there are hidden events in the noise waveforms, we run the machine learning model (see Section 2.2.4) to test whether events could be detected and only found very few occasions where events may have been present.

We organize the three-component waveforms into NumPy arrays and define a *stream* as a three-component array^{93,125}. To improve accessibility in the machine-learning ecosystem, we follow the SeisBench data format convention. The metadata is stored in CSV (comma-separated values) files, while all waveforms are stored in the Hierarchical Data Format version 5 (HDF5) format. The signal-to-noise ratios (SNR) are calculated (detailed below) and saved as attributes in the metadata file.

After applying the selection criteria described above, more than 70% of the ComCat events are kept in the dataset. Figure 2.3 shows the map of the selected events. The datasets cover events within the authoritative boundary of the PNSN, offshore in the Jan de Fuca Ridge, underneath Vancouver Island, and further East in Idaho. We provide an overview of the final number of ComCat waveforms and events in Table 2.1. The summary compiles the data volume across magnitudes from 0 to 6.4. It is possible that most of the events discarded by the selection had no S-wave picks for clipped waveforms. Our selection criteria also excluded more events before 2010, which we attribute to the much fewer S picks available when the data is clipped or when only vertical-component stations are available.

Magnitude range	Number of included events	Percentage of available events	Number of independent streams
0 - 1	19,735	77.1%	70,168
1 - 2	21,717	79.2%	95,406
2 - 3	4,825	42.8%	21,901
3 - 4	398 (15)	37.9%	2,332 (296)
4 - 5	31 (3)	77.5%	205 (138)
5 - 6	1 (1)	100.0%	4 (86)
6 - 7	0 (1)	N/A	0 (85)
0 - 7	46,707	71.4%	190,016

Table 2.1: Number of included ComCat events in each magnitude range. The magnitude used here includes duration (Md), local (Ml), and hand (Mh) magnitude. The number of streams includes both velocity and acceleration channels. Also provided is the number of included events as a percentage of downloaded ComCat events. Numbers in the parentheses show the events and streams from the 20 December 2022 Northern California earthquake sequence and are not included in the total number of events/streams.

2.2.4 Machine Learning Phase Picker and Enhanced Earthquake Picks

We provide an alternative catalog of phase picks from the earthquake event catalog as a use-case of the dataset and a research-grade catalog of new picks of P and S waves using Machine Learning (ML). Automating phase picking using deep neural networks has revived the methodological development for picking seismic waves^{171,180}.

Here, we use the Earthquake Transformer architecture from Mousavi et al.¹⁷⁴ and implement phase-picking benchmark tests on the ComCat events. The SeisBench toolbox provides a set of Earthquake Transformer weights for models pre-trained with different datasets. We select all windowed waveforms from HH?, BH? and EH? channels and detrend the waveform. We compare the picks made by these models trained on STEAD, ETHZ, SCEDC, and INSTANCE datasets with the PNSN analyst picks recorded in the ComCat events. We demonstrate their performance by showing the residuals between ComCat picks and ML-predicted picks for P- and S-waves. The performance metrics are the mean absolute error (MAE), the root-mean-square error (RMS) for the phase picking, and the percentage of detected picks relative to ground truth picks.

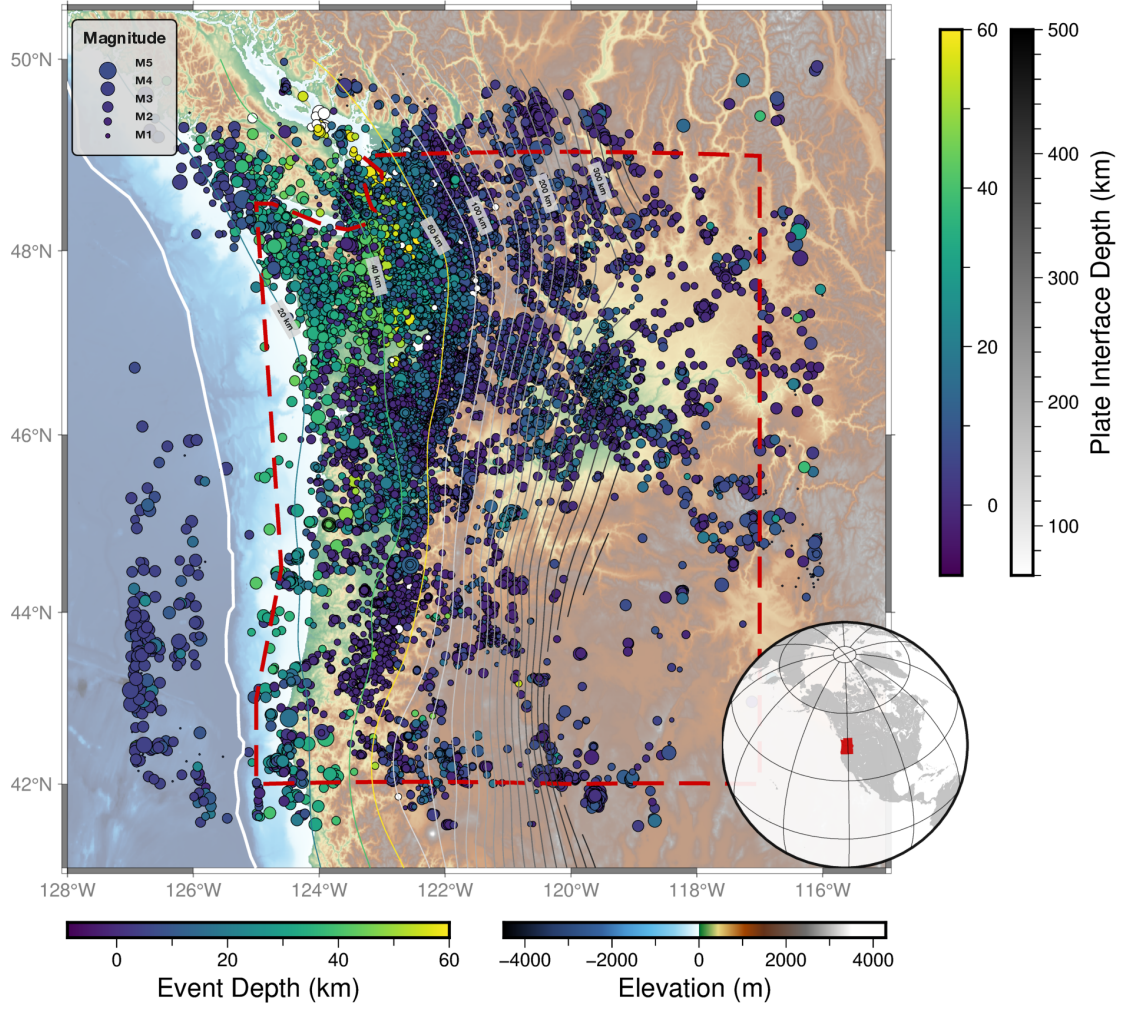


Figure 2.3: Locations of the events included in the ComCat dataset. The red dashed polygon denotes the authoritative region boundaries of PNSN. The solid lines mark the depth contour of the subduction slab with a 20 km interval⁹⁶. Some contours are labeled by the slab depth. The plate boundary between Juan de Fuca and North America Plate (plate depth 0 km) is delineated in the white line. Some events are color-coded white because they are deeper than 60 km. These are intermediate-depth earthquakes.

The input size of the Earthquake Transformer using SeisBench is 3-component, 60 seconds at 100 Hz. The probability threshold for picking is 10%. Figure 2.4 shows the distributions of the residuals among models and for both P and S wave picks.

The approaches to benchmark the detection and picking performance are 1) the seismic network-specific expectations for the manual picking uncertainties and 2) the comparison of bias and variance in the residual distributions relative to other studies^{174,180}. We find a general trade-off between detection

accuracy (completeness) and phase-pick quality (low errors). The model trained with the STEAD dataset has the best quality in phase picks relative to the (ground truth) analyst’s picks, but it misses more than 20% of the detections. In contrast, the model trained with the SCEDC dataset had the best detectability and only missed about 5% of arrivals for both P- and S-waves, but the picking accuracy, especially that of S-waves, is poor. Both models show negative mean residuals for both P- and S-waves, indicating that the ML picks are always earlier than the manual picks. There is also a similar pattern on the model trained with ETHZ and INSTANCE dataset in Figure 2.4. The performance trade-off between detection and picking accuracy makes retraining the phase pickers using the PNW data necessary.

Using our curated dataset of ComCat earthquakes and explosions, we retrain the Earthquake Transformer model. Instead of training from scratch (randomly initialized weights), we start the training from the SeisBench-trained model, which used the STEAD dataset, and continue training for additional 100 epochs on our dataset. We note that about 3% of the STEAD data set contains PNSN data, which may be problematic for data leakage. However, the STEAD data is bandpass-filtered 1-45 Hz, while we do not filter the data. We randomly select 70% of the ComCat dataset for training and use the rest 30% for testing. We use a triangular label with a 10-sample half-width. We use the same loss function that Mousavi et al.¹⁷⁴ used to train the Earthquake Transformer (a weighted sum of loss from P-, S- and detection branches). We use a small learning rate (1×10^{-4}) with the Adam optimizer¹¹⁹ during the training. Compared with the other pre-trained models, the transfer-learning on the PNW dataset improves the detection accuracy, considerably improves the S-wave picks, and performs as well as the STEAD-trained dataset (see Figure 2.4). Although not eliminated, the negative mean residuals are reduced after retraining. We also test all these models on strong-motion (acceleration) channels, for which INSTANCE contains the most acceleration waveforms (28.3%). The PNW transfer-learned model outperforms other pre-trained models on strong-motion channels, as shown in Figure B.5.

The ability to find more and accurate picks by the retrained Earthquake Transformer makes it possible to create a future Machine-Learning-enhanced earthquake catalog. We revisit waveforms from the ComCat events that included either P or S picks. There are 683,133 P- and 244,431 S-wave picks for 62,054 events from these waveforms. We detect 16,201 (2%) and 207,146 (85%) new arrivals out of 686,748 time windows for P- and S-waves using the refined phase picker. As a crude quality control,

we remove the picks where the ratio between the S-travel time and the P-wave travel time exceeds 2.5 or below 1.5. We add these picks with PNSN manual picks as a part of the curated dataset in a separate file. We also use this retrained model to predict the noise waveform and drop those with any prediction greater than 0.1. This step effectively removes unpicked seismic events in the noise waveform.

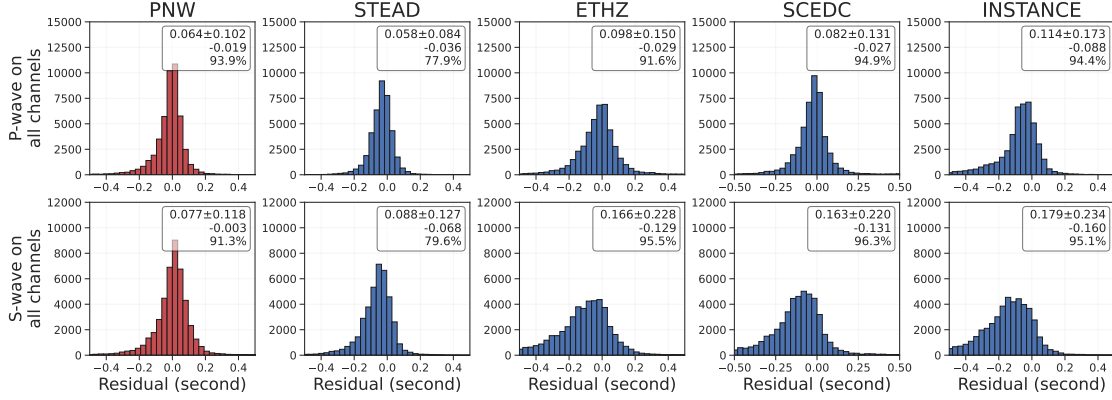


Figure 2.4: Distributions of P- and S-wave picking residuals ($t_{ML} - t_{PNSN}$) from the benchmark testing on velocity seismograms. The number in the upper right corner of each subplot shows the mean absolute error (MAE), the root-mean-square error (RMSE) as the uncertainties to the MAE, the mean value of the residual, and the picking completeness in percentage concerning the ground truth. The PNW-retrained Earthquake Transformer outperforms the other four pre-trained models from SeisBench³⁰⁷ in both picking accuracy and detecting completeness.

2.3 Results

The datasets consist of two files per set, one HDF5 file containing the waveforms and a CSV file with the metadata (attributes).

2.3.1 Waveforms

There are 190,016 and 9,267 three-component streams curated from ComCat and exotic event catalogs, respectively. Figure 2.5 shows the counts of streams arranged by channel type as a yearly estimate. We store all waveforms in HDF5 files using h5py⁵⁰ and index them by the trace name in the metadata. The attribute `trace_start_time` in YYYY-MM-DDTHH:MM:SS.SSSZ format describes the UTC time at which the stream begins.

The data is saved as vertically concatenated NumPy arrays of fixed window length (here 150 s), three

components. It is distributed over several “buckets” that are “groups” under the HDF5 taxonomy. The trace name (a data attribute saved in the metadata data frame), the index of the data in the bucket, and the index of the first dimension.

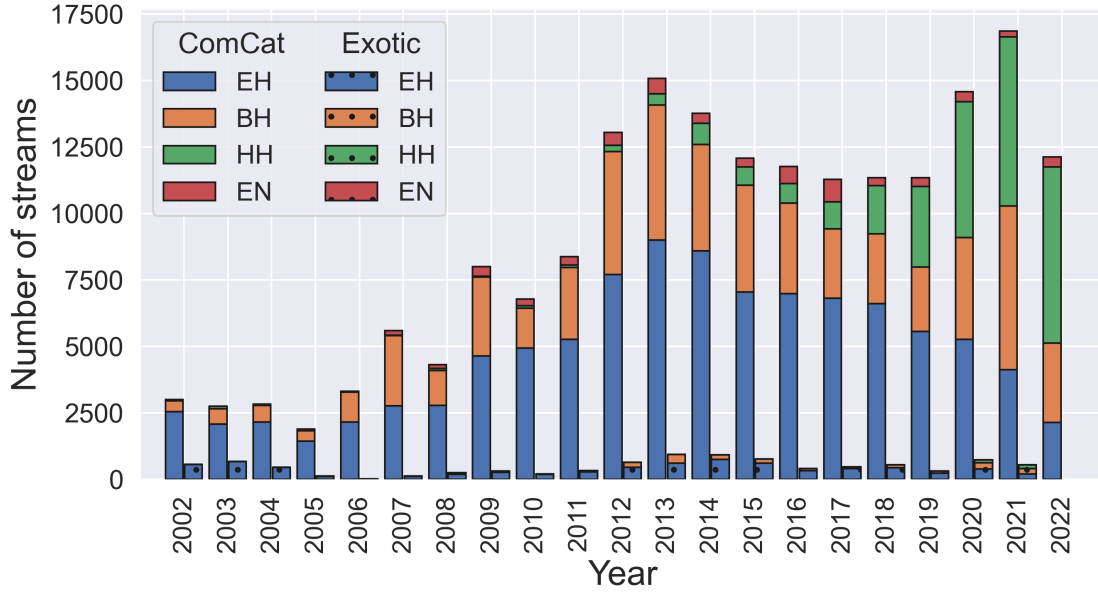


Figure 2.5: Number of streams from each channel type used in the ComCat and exotic event catalogs through time. Short-period (EH?) and broad-band (BH?) sensors were the predominant channels for both ComCat and exotic datasets before 2012, while the recording at higher sampling rates at broadband sensors (HH?) increasingly has become the standard since then. A limited number of streams from strong-motion accelerometer EN? channels is available in the dataset since 2007.

2.3.2 Metadata

The metadata describes the waveform data and its attributes and is essential to our dataset. Each stream corresponds to one record (or a row) in the metadata file. We follow SeisBench conventions again. The unit of each attribute is appended as part of the attribute’s name. For example, `source_latitude_deg` indicates the latitude of the source in degrees. A full description of the attributes is listed in Table 2.2. As many attributes are self-explanatory, we provide more details below.

Attribute	Description	Example
event_id	Event identifier	uw10564613
source_origin_time	Source origin time in UTC	2002-10-03T01:56:49.530000

source_latitude_deg	Source latitude in degree	48.553
source_longitude_deg	Source longitude in degree	-122.52
source_type	-	earthquake
source_type_pnsn_label	PNSN AQMS event type	eq
source_depth_km	Source depth in kilometers	14.907
source_magnitude_preferred	-	2.1
source_magnitude_type_preferred	-	Md
source_magnitude_uncertainty_preferred	-	0.03
source_local/duration/hand_magnitude	Ml, Md, and Mh if available	1.32
source_local/duration_magnitude_uncertainty	Magnitude uncertainty if available	0.15
source_depth_uncertainty_km	Source depth uncertainty in kilometer	1.69
source_horizontal_uncertainty_km	Source horizontal uncertainty in kilometer	0.694
station_network_code	FDSN network code	UW
station_code	FDSN station code	GNW
station_location_code	FDSN location code	01
station_channel_code	FDSN channel code (first two digits)	BH
station_latitude_deg	Station latitude in degree	47.5641
station_longitude_deg	Station longitude in degree	-122.825
station_elevation_m	Station elevation in meter	220.0
trace_name	Bucket and array index	bucket1\$0,3:15001
trace_sampling_rate_hz	All traces resampled to 100 Hz	100
trace_start_time	Trace start time in UTC	2002-10-03T01:55:59.530000Z

trace_P/S_arrival_sample	Closest sample index of arrival	8097
trace_P/S_arrival_uncertainty_s	Picking uncertainty in second	0.02
trace_P/S_onset	P- or S-wave onset	emergent
trace_P_polarity	P-wave arrival polarity	positive
trace_has_offset	Any visible offset in the trace	0 or 1
trace_missing_channel	Number of missing channel of the trace	0, 1, or 2
trace_snr_db	SNR for each component	6.135 3.065 11.766

Table 2.2: Attributes in the metadata file. Some source attributes are not available for exotic events.

Station Network Code

Stations selected in both datasets may come from nine different FDSN network codes. These stations are either installed and maintained by PNSN (e.g., UW and UO) or used by PNSN when doing phase picking and events locating (e.g., PB, CC, IU, CN, HW, TA, US). Maps of the stations shown in the dataset show a similar distribution for both ComCat (Figure B.1) and exotic events (Figure B.2). All stations are in-land stations, and no off-shore stations (e.g., OOI) are used in our dataset. The numbers of streams from each FDSN network and their references are listed in Table 2.3. PNSN stations contribute more than 85% of streams in the ComCat and Exotic event datasets.

Event ID

An event identifier (ID) is given to each event by the PNSN after the processing is finalized and sent to ANSS through USGS Product Distribution Layer (PDL). The ComCat events contributed by the PNSN have IDs of eight-digit numbers with a “uw” prefix, e.g., “uw10568488”. The event IDs are unique in the catalog. The exotic event IDs are internal to the PNSN AQMS database and cannot be accessed through USGS. To distinguish them from ComCat events, we add a “pnsn” prefix to their

event IDs.

Event Type

When processing a seismic event as the seismic data comes in, the event type is manually specified by the network analysts. For example, the PNSN labels “probable explosion” waveforms that have the characteristics of shallow quarry blasts (strong P waves and location near known quarries). Until the 1990s, the PNSN would confirm these explosions by phone confirmation, though this is no longer routinely done. When sending the finalized event from the AQMS database to the ComCat, PNSN maps and merges several types of events into one: “earthquake”, “slow earthquake”, and “long period volcanic earthquake” are mapped into the “earthquake” category; “explosion”, “shot” and “probable explosion” are merged into the “explosion” category. For simplicity and consistency, we use the event types “earthquake” and “explosion” for the ComCat events, but their original event types are also included for reference in the metadata. Table B.1 lists the latest PNSN event-type labels from the PNSN AQMS database.

Source Magnitude and Type

The event size, as represented by the source magnitude, is only available for the ComCat events. All ComCat events included in the dataset have magnitudes less than seven and greater than zero, as shown in Table 2.1. The magnitude completeness of the catalog is estimated using the method of Wiemer and Wyss²⁹⁷ and found to be around 2 for the years 2019-2022 (Figure B.6). The types of magnitudes reported are typical to regional earthquakes that have local seismicity: the local magnitude (Ml) and the duration magnitude (Md).

There are three types of magnitude used in the dataset. The PNSN uses a local magnitude (Ml)¹¹³ that measures the magnitude of a local earthquake using the average maximum amplitudes of two horizontal seismograms converted to have the Wood-Anderson response, preferably taken from broadband seismometers, and corrected for the distance between the source and the receiver. Such magnitude is reported by the National Earthquake Information Center (NEIC) for all earthquakes in the US and Canada. The coda duration magnitude Md is calculated based on the duration of shaking measured on

the vertical component and could be the only available magnitude product for small events or those not well recorded on well-calibrated stations with horizontal components. Over the course of time, processes to calculate the magnitudes vary because of varied processing routines and analyst interventions.

Until 2012, the PNSN only reported duration magnitude to ComCat for most earthquakes using the algorithm from Crosson⁵⁴, except for a few significant events that were manually changed to the local magnitude. The early seismic stations of the PNSN only had vertical components, a small dynamic range, and short-period sensors that would clip even for relatively small magnitude events. It is not possible to obtain a local magnitude from such data. As the network modernized over time, higher dynamic-range three-component sensors were added, and the data quality improved, which allowed PNSN to determine an Ml for more events. From 2002 to 2011, 46,326 events had duration magnitude preferred, while only 483 events (average magnitude 2.45) had local magnitude reported as the preferred magnitude type. From 2012 to 2015, the PNSN calculated and reported both duration and local magnitudes, though the local magnitude was still only calculated for larger events. Since 2015, the PNSN has switched from having duration magnitude to the local magnitude as the preferred and default magnitude. 80% of all events included in the ComCat dataset until 2008 have a duration magnitude preferred, after when there were increasingly more Ml-preferred magnitudes (Figure 2.6). While the duration magnitude is still calculated, it is only the preferred magnitude for about 10% of the events each year. From 2002 to 2022, there were also 111 events with an Mh magnitude in the dataset, extracted from the NEIC and manually added by the network analysts. Note that there is no moment magnitude Mw reported in this dataset because the moment magnitude is obtained from low-frequency seismograms, which are often buried in the seismic noise for small earthquakes. Mw magnitude may be included as Mh.

There are potential challenges in interpreting the magnitudes as ground truth labels. Md and Ml have known systematic biases that arise from the particularly high near-source scattering of shallow earthquakes or quarry blasts^{123,287}. In 2012, the PNSN adopted AQMS, which included a method to measure coda duration that was inconsistent with the previously used method. The PNSN staff did a rough recalibration of their Md relationship to partially account for the systematic difference. However, there is a known inconsistency of the Md magnitudes for the smallest events before 2012 and after 2012. Future efforts must be made to re-calculate the magnitudes more systematically, ideally using consistent

methods, throughout the 2002-2022 period.

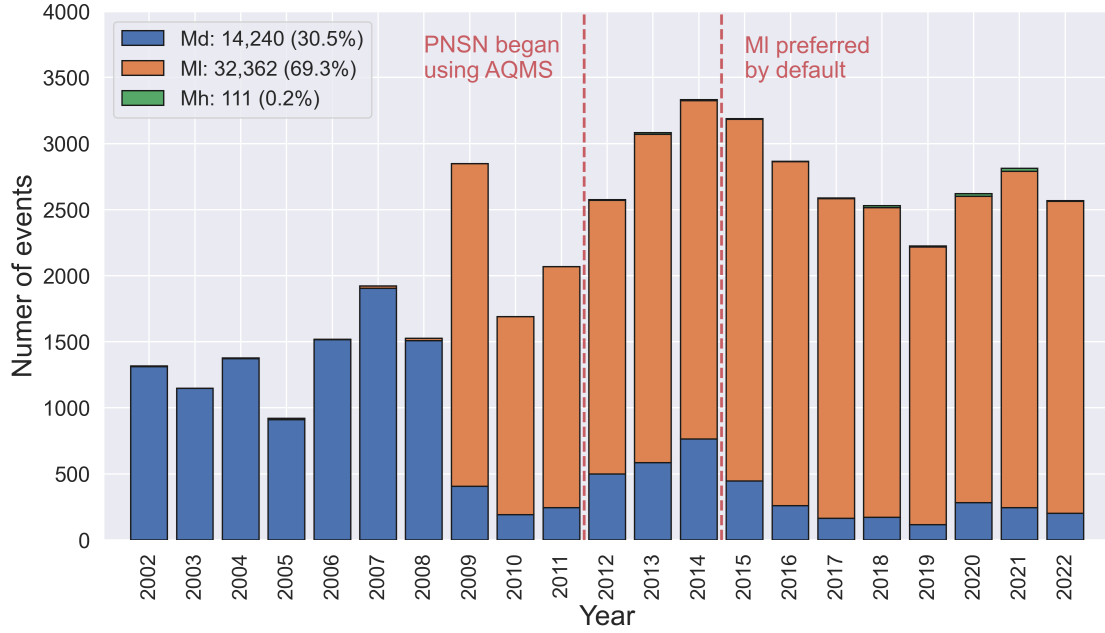


Figure 2.6: Magnitude types of ComCat events as a function of time. Md and Ml denote duration and local magnitudes, respectively. Mh denotes magnitudes manually inserted by the analysts. Before the PNSN began using the ANSS Earthquake Monitoring System (AQMS) in 2012, 483 events had Ml estimates, and 46,326 events had Md estimates in the dataset.

Table 2.1 shows the event counts per magnitude bin for this dataset. The largest event in the dataset comes from Mw 6.4 Northern California, 20 December 2022 by the CISN, but this event was outside the PNSN’s authoritative boundaries. Thus, ComCat preferred an origin contributed by CISN. The largest earthquake in this dataset within PNSN’s authoritative boundaries is Md 4.8 Brinnon, Washington, on 25 April 2003 (event ID uw10583988). Relatively small magnitude uncertainty (0.04), depth uncertainty (0.59 km), and horizontal uncertainty (0.347 km) were reported.

Stream Signal-to-Noise Ratio

The signal-to-noise ratio (SNR) is an important factor in measuring the noise level in the traces. Similar to Michelini et al. ¹⁶⁵, we define the noise window as 8 seconds before the P-wave arrival for the ComCat events. To better capture the energy of emergent S-wave onsets, the signal window is defined as 1 second before to 2 seconds after the S-wave arrival. For the exotic event catalog, since P-wave and S-wave arrivals may not be available, the noise window is defined to begin 12 seconds after the beginning of the traces.

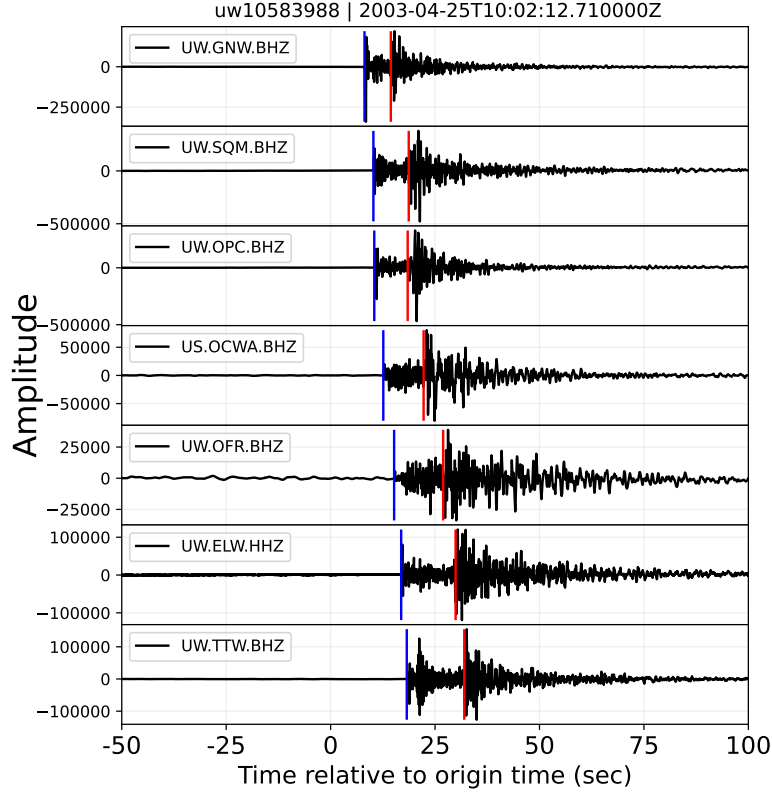


Figure 2.7: Waveform from event uw10583988 (M4.8 Brinnon, Washington, 25 April 2003) included in the dataset. Only the vertical component is shown. The blue and red vertical lines show P- and S-wave arrival picked, respectively. The amplitude units are in counts.

The signal window is the same as exotic events, P- or S-wave, whichever is available. For each component, the SNR is defined as

$$SNR = 20 \log_{10} \frac{|S_{98}|}{|N_{98}|} \quad (1)$$

where $|S_{98}|$ and $|N_{98}|$ are the 98% percentile of the absolute values in the signal and noise window, respectively. When no data is available, e.g., a single-channel station with only the EHZ channel, **NaN** (**not-a-number**) is filled as a placeholder in the missing channels. Figure 2.8 shows the distribution of individual SNRs calculated from the ComCat and exotic event catalogs. The traces with $SNR > 80 \text{ db}$ (indicating an error in the noise window) or $< -20 \text{ db}$ (indicated too low of a signal) are removed from the dataset.

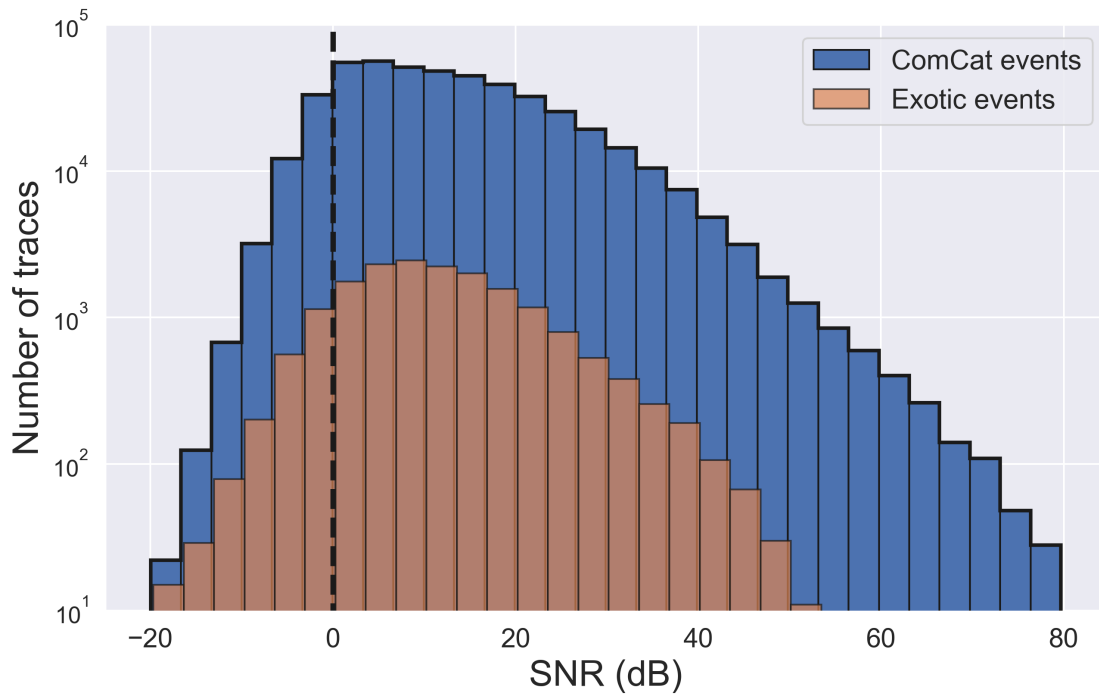


Figure 2.8: Distribution of signal-to-noise ratios (SNR) of the traces from ComCat and exotic events. SNRs are calculated on each component of the three-component streams.

Uncertainties

The metadata includes four types of uncertainties for the ComCat events. The P- and S-waves arrival uncertainties are estimated at the time of picking. Before the PNSN used AQMS, the uncertainty was directly measured and recorded in the phase data, and a weight was calculated. Using Jiggle from AQMS since 2012, the analysts assign weight as an integer ranging from zero to four to each pick by visually measuring the impulsivity of the arrival. A zero weight indicates the highest accuracy of picks, typically for P-wave arrivals, and has 0.03 seconds of uncertainty. A weight of three indicates a low pick accuracy, typically for S-wave arrival with 0.3 seconds of uncertainty. Phase uncertainties are used when locating the events, but those with uncertainty weights of four are typically not used in earthquake locations. Before 2012, PNSN used Spong (an adaption of Fasthypo¹⁰¹) as the location engine. This changed to HYPOINVERSE¹²⁰ after PNSN started using AQMS and Jiggle.

The origin location (depth and horizontal) uncertainties are the error estimated from the location engine. Figure B.8 shows the locations of the events with horizontal uncertainty greater than 20 km.

Note the cluster off-shore Oregon that is outside of the PNSN authoritative boundaries. The PNSN has poor location constraints on these events since there are almost no offshore seismic stations except for the Ocean Observatories Initiative Regional Cable Array (FDSN network code OO²²⁹), which are occasionally picked during PNSN routine data processing. ComCat may not choose these origin products from PNSN as preferred. However, the events with high horizontal uncertainty only make up 0.4% of all ComCat events, and their picks are still accurate enough to be part of the dataset.

We also include the magnitude uncertainties in the metadata. The magnitude is first evaluated on the channel level. For three-component stations, the channel-level local magnitude is calculated only if a P- or S-wave is picked on one of the components to only select clear signals. Since 2012, a few single-component stations (EHZ) also contribute to the local magnitude and have the same weight as three-component stations. The event magnitude is the median of all channel magnitudes that meet the SNR criteria. The event magnitude uncertainty is the median absolute deviation (MAD) of channel magnitudes used for event magnitude calculation. These uncertainties are calculated for all magnitude types except Mh.

P-wave Polarity

When analysts pick the phase arrivals, Jiggle also automatically measures the first motion of the P-wave picks with weights less than one (e.g., best waveforms), leaving the rests as “undecidable”. The analysts can manually override these polarities if they are confident. Less than 42% of P-waves in this dataset have undecidable polarity information. The P-wave polarity ratio between positive and negative as a function of the year is shown in Figure B.9. The sudden switch to a preference to assign or report positive polarities in 2012 highly suggests that the switch to AQMS and Jiggle in 2012 has affected the PNSN analysts’ output. Until this data collection effort, we were unaware of this fact, and the reason for the abrupt change is unclear.

Network FDSN Code	Number of Streams			Reference
	comcat	exotic	noise	
UW*	100,561	5,653	26,716	University of Washington ²⁷³
PB	41,674	461	11,126	Plate Boundary Observatory Borehole Seismic Network

CC	23,988	3,119	6,784	Cascades Volcano Observatory/USGS ³⁵
TA	9,912	4	3,012	IRIS Transportable Array ¹¹¹
CN	6,008	2	1,692	Natural Resources Canada (NRCAN Canada) ¹⁸⁴
US	3,420	0	981	Albuquerque Seismological Laboratory (ASL)/USGS ⁷
UO*	3,593	28	891	University of Oregon ²⁷²
HW	840	0	252	Hanford Washington Seismic Network
IU	20	0	4	Albuquerque Seismological Laboratory (ASL)/USGS ⁶

Table 2.3: Description of network FDSN code and their references. Networks annotated by an asterisk mark (*) are maintained by the PNSN. The number of streams shown for each network is from ComCat events, exotic events, and noise, respectively. PB and HW network does not have a registered FDSN network DOI.

2.4 Conclusion

This work contributes to collecting and curating a seismic dataset for the Pacific Northwest region. The curated dataset is provided with the long-standing work and labeling of the Pacific Northwest Seismic Network analysts and seismologists. We described the temporal and spatial characteristics of the data attributes.

This original contribution focused on preparing the seismic waveforms and PNSN-provided data attributes (phase picks and default source parameters). We picked additional waveforms for the recent 20 December 2022 Northern California earthquake sequence, the largest event recorded recently in proximity to the PNSN authoritative boundaries. We also transfer-learned an established phase picker, the Earthquake Transformer¹⁷⁴, on the best quality of the PNSN picks and provided additional picks for S waves, which we provided in this contribution as an alternate catalog of picks.

There remains tremendous work to improve the quality and consistency of the data attributes. We use version control on the curated dataset through GitHub to allow for future development of the data and metadata. Examples of future developments may be a refinement of the current attributes or the addition of new labels. In particular, the attribute “magnitude” should be carefully interpreted as 60%

of the catalog uses duration magnitude, and 40% of the catalog uses the local magnitude, but both may have biases. Therefore, a follow-up task is to re-calculate these magnitudes using consistent methods. Another avenue for improvement is to re-estimate the polarity of the P and S waves, using the known labels and predicting the “undecided” labels. Furthermore, we have not yet included other types of tectonic events, such as low-frequency earthquakes⁶⁶, but these would improve the diversity of events. Finally, an obvious next step will be event classification work that will take the waveforms and predict the event type.

Code and Data Availability

The facilities of IRIS Data Services, and specifically the IRIS Data Management Center, were used for access to waveforms, related metadata, and derived products used in this study. IRIS Data Services are funded through the Seismological Facilities for the Advancement of Geoscience (SAGE) Award of the National Science Foundation under Cooperative Support Agreement EAR-1851048. The earthquake catalog on ComCat contributed by PNSN was downloaded using libcomcat⁹⁸. The Earthquake Transformer implementation is from the SeisBench toolbox³⁰⁷. All plots are made with Matplotlib¹⁰⁹ and PyGMT²⁷¹. The final datasets and the codes used in this study are available at <https://github.com/niyiyu/PNW-ML>, and we have made the dataset available through SeisBench at <https://github.com/seisbench/seisbench>.

This chapter was published as Ni, Y., Denolle, M. A., Fatland, R., Alterman, N., Lipovsky, B. P., Knuth, F. (2024). An Object Storage for Distributed Acoustic Sensing. Seismological Research Letters, 95(1), 499-511. doi: 10.1785/0220230172

3

Object Storage for Distributed Acoustic Sensing

3.1 Introduction

Distributed Acoustic Sensing (DAS) is an emerging technology for measuring seismic vibrations. DAS is revolutionizing geophysical sensing because it provides unaliased wave field images with resolution as

fine as ~ 0.5 m along optical fibers up to ~ 170 km in length²⁸³. DAS utilizes repeated laser pulses along optical fibers to measure phase changes of light that is scattered from imperfections along the length of the cable. These phase changes are proportional to the rate of extensional strain along the axis of the fiber. Typical sampling rates using DAS are in the range of 0.01 Hz - 100 kHz, with longer cable lengths ultimately limiting the highest possible sampling rate. DAS dramatically expands the capability of dense seismic observation and has been used for a vast range of applications, such as detecting new tectonic faults¹⁴⁰, high-resolution subsurface imaging^{18,314}, cataloging earthquakes¹⁸⁵, tracking marine mammal vocalizations²⁹⁹, observing ocean gravity waves³⁰², and monitoring vehicular traffic and infrastructure state of health^{141,289}. DAS has the potential to transform geophysical sensing in the broadest sense and push the frontiers of geophysical research.

DAS data rates vary widely depending on the observational target. Data rates may be calculated as $r = \alpha f L / \delta x$, where α is the size of one sample on one channel, here assumed to be $\alpha = 4$ Bytes, f is the sampling frequency, L is the total fiber length, and δx is the spatial sampling interval (channel spacing). DAS observations targeted to measure ocean surface gravity waves (SGWs) need only be recorded at 1-10 s period sampling rate to capture unaliased $T = 2-20$ s period SGWs. Unaliased spatial sampling in the shallow water limit requires a spatial sampling $T\sqrt{gH}/2$, where g is gravitational acceleration, and H is the water depth. In 100 m deep water, 2-20 s period SGWs have wavelengths of $\sim 30-300$ m; in 1 km deep water, the same periods have wavelengths 100-1000 m. Assuming maximal DAS range ~ 150 km and that samples are stored as single precision floating point numbers gives data rates from 50 MB/d to 1.7 GB/d. As an example of an intermediate data rate, we now consider DAS observations of tectonic earthquakes. Here, although seismic Q limits observable source frequencies to ~ 10 Hz²⁴⁴, azimuthal variation in seismic observables¹¹⁶ and small-scale variation in near-surface structure²⁵³ may justify spatial sampling at 1-10 m; the total resulting data rates are on the order of 14-28 GB/d, assuming a shorter, 10-20 km long urban optical fiber span. Other applications may result in much higher data rates. DAS observations targeted to observe the vocalizations of marine mammals may attempt to sample ~ 750 Hz signals with 2 m wavelengths⁶⁴ associated with the acoustic wave speed in water. Such observations result in data rates of 19.4 TB/d. Optical fading and other effects may, in practice, necessitate greater spatial sampling intervals e.g., as in²⁹⁹; for that reason, we take this number to be an upper bound estimate.

Managing 10-1000 TBs data volumes is presently the foremost barrier to democratizing DAS data access and research. This poses a challenge for storage and sharing on multiple fronts. First, the data volume is immense and surpasses today’s total volume of seismic data stored at national archives (e.g., at Incorporated Research Institutions for Seismology Data Management Center, IRIS DMC¹⁶). DAS metadata is largely unconventional and yet to be agreed upon by the community. Furthermore, DAS data sharing has been limited because, as of June 2023, no federally funded facility accepts DAS data and because enabling public access to large datasets at an individual principal investigator (PI) level is difficult. A pioneering effort to address these needs led a collective of PIs to open a shared storage system for heterogeneous DAS datasets²⁵⁴. Similar to the Department of Energy’s Geothermal Data Repository (GDR²⁹⁴), PubDAS uses GLOBUS endpoints to transfer entire DAS data files to end users. Beyond data storage, the processing of large volumes of DAS data is limited due to impractical choices in data formats (detailed below), limiting the processing required for seismological research. New DAS data analysis frameworks have been developed, e.g., DASCORE that provides DAS data I/O and basic processing³⁶, and DASSA that enables parallel data processing on modern supercomputers⁶¹.

Cloud computing has revolutionized scientific computing with large datasets. Seismological, and therefore DAS, research typically handles subsets of data using identical workflows and software. These workflows are perfectly suited for horizontal parallelization, the optimal design for cloud computing¹⁵². The rise of cloud computing also fueled the development of data formats optimal for cloud-native storage and parallel throughput. The combination of cloud-native data formats and horizontal scaling of cloud computing architecture is an attractive solution for DAS seismological research.

This study proposes a novel data platform for DAS that incorporates object storage with cloud-optimized data formats: we store DAS data in Zarr¹⁶⁶ or TileDB¹⁹⁸ file formats using the MinIO open-source object storage. MinIO is compatible with Amazon Web Services (AWS) Simple Storage Service (S3) object storage. The combination of these technologies replicates on local Linux servers a cloud-native framework that can be seamlessly ported to AWS. We demonstrate below with several canonical examples that typical DAS workflows can be designed with low-memory, parallelized jobs on a local Linux server and AWS. The demonstration includes examples of public access to the new DAS public archive of the University of Washington (UW) FiberLab.

3.2 Background

A typical data flow in a DAS experiment begins at acquisition. DAS interrogator units usually have built-in storage capable of storing several TBs of data. Some installations may involve real-time data streaming, as was the case for the Whidbey and SeaDAS-N experiments at the FiberLab^{143,144}. Given the typical configurations with the number of channels and sampling rate, internal storage is insufficient for months-long continuous recordings that can accumulate up to hundreds of TBs. For this reason, it is common to use a separate Network Attached Storage (NAS) connected to the DAS interrogator via ethernet in order to sync the data as the unit is recording. The NAS can also serve as the primary data repository for further processing, or data may be subsequently transferred to a separate archival server or workstation. This last step necessitates transferring full copies of the relevant HDF5 files, which could amount to hundreds of TBs and even petabytes (PBs) worth of data. This is the model, for example, of PubDAS and the GDR²⁵⁴. Still, the expectation is that data transfer will proceed with single-thread requests and be limited by the internet speed.

3.2.1 Cloud Storage

Cloud storage is a general term referring to services that allow users to store data somewhere other than their local computers but with the maintenance and upkeep of such storage systems handled by the “cloud provider” (e.g., AWS) rather than the end-user or IT personnel affiliated with the end-user. In this arrangement, the cloud provider is responsible for storing, managing, and maintaining the infrastructure and network, ensuring that data is safely accessible with virtually unlimited capacity. However, all of this is provided at a cost, and consequently, cloud providers will offer many storage models with variations in flexibility to stretch the end-user’s dollar further. One such type is “object storage”, which treats each data file as an object identified by globally unique IDs. As there are no hierarchical structures across objects, this type of storage appears simpler than a traditional networked file system. It can be scaled up much larger for an order-of-magnitude reduction in cost. It is provided by most cloud platforms, usually under individual brand names (e.g., AWS “S3”, Microsoft Azure’s “Blob Storage”, and Google Cloud Platform’s “Object Storage”). Several do-it-yourself server products allow users to deploy local object storage systems. One example of these local storage systems is MinIO. This high-performance

open-source object storage implementation provides an “object storage” abstraction across various modes of deployment. The single-node single-drive (SNSD) is designed for a single machine, single drive, and datasets typically up to 20 TBs on a workstation. The single-node multiple-drive (SNMD) is designed for a single machine with multiple drives and datasets typically up to 500 TB on a rack-mounted server. Lastly, the distributed multiple-node multiple-drive (MNMD) is designed for a multiple-node server and multiple drives with datasets typically at and above 1 PB on multiple units of rack-mounted servers. MinIO is AWS S3-compatible in that it uses an identical API, which supports access control through the credential keys and secrets granted to users with proper permissions. This compatibility facilitates researchers moving between commercial cloud storage systems and locally hosted options with minimal impact on their code or workflow.

3.2.2 Challenges in Storing and Formatting Large Seismic Data

Traditionally, seismic data repositories have mostly been hosted by data centers managed partly by research institutions. One example is the IRIS DMC and its data archive in Seattle, Washington, USA. The archive has grown over time and accumulated ~ 877 TB data as of 1 Jan 2023. The IRIS DMC has shipped 7.2 PB data cumulatively between 1 January 1990 and 1 January 2023 and more than 1 PB data in the single year of 2022. Other seismic archives face similar challenges, especially driven by user-specific research and trends toward tackling larger and larger datasets^{16,209}. Supporting this level of data growth has already greatly challenged the DMC and has prevented the archive from hosting more voluminous DAS datasets.

Another example of a large seismic archive on the cloud is the AWS Open Data of the Southern California Earthquake Data Center (SCEDC; s3://scedc-pds), which has provided its entire archive of continuous time series and curated dataset, including an earthquake sequence dataset recorded on DAS stored in DAS-native format^{95,321}. The SCEDC has not changed the formats the seismic instruments provide, which suit cloud storage with small (~ 10 MBs) single-day, single-channel files. Another example of a large seismic archive is that of PoroTomo⁷⁴, which was collected in 2016 and is hosted in DAS-native SEG-Y and HDF5 formats as an AWS Open Data (s3://nrel-pds-porotomo/). The DMC is transitioning to cloud storage on AWS in 2023 and will adopt a new cloud-optimized data format for future incoming

data. Thus, there has been a trend of moving data to the cloud where storage is available, albeit oftentimes at significant financial cost.

In seismology, variants of HDF5 schema (e.g., ASDF¹²⁶, HDF5eis²⁹⁶) have been introduced to accommodate multi-dimensional time series with no dataset size restrictions. However, the monolithic nature of all HDF5 variants introduces data access latency, whether on local drives or the commercial cloud. Since the hierarchical data structure must be flattened to be stored on disks, HDF5 readers must first read several byte blocks to decode the structure and find the address of the actual data in the file. Therefore, there are trade-offs in chunking the data: the smaller the chunk, the smaller the memory requirement, but the longer it takes to read the data schema map⁴⁹. Latency is made worse when compression is enabled. Another bottleneck of HDF5 is that the entire file has to be downloaded locally to be read. The downloading effectively duplicates data on local drives and is dramatically limited by the data volume (e.g., file size), the network connectivity or speed, and the storage availability on the local computing server. The computing overhead of copying large volumes of data is wasted if users only need a subset out of the whole data, which defeats the purpose of designing a single large file with small datasets within. Similar formats used in the geoscience community suffer the same issues on the cloud, such as NetCDF²¹⁵ and GeoTIFF²¹⁷.

Efforts have been made to optimize reading HDF5 on the cloud. One example is made by the Kerchunk project (see Data and Resources) to scan the HDF5 structure (byte ranges and compression information) and saves this information as index files in order to locate datasets directly without changing the original HDF5 file. Additionally, the National Aeronautics and Space Administration (NASA) supported sliderule project has developed a cloud-optimized read-only HDF5 library, H5-corro. H5-corro is currently used to read data from the ICESat-2 laser altimeter, stored directly as HDF5 files, into AWS S3²⁴³. Although both of these projects are under ongoing and rapid development, neither approach generalizes well for diverse H5 structures and data schema types¹⁸⁹.

The ability to efficiently read small space-time subsets of data is essential for many scientific studies using DAS. This issue occurs with DAS because, for example, the complexity of the cable geometry may challenge traditional seismic data processing. Rather than using the entire cable, DAS analysts may therefore select linear segments for surface wave dispersion analysis⁷², velocity monitoring²²¹, and

beam-forming¹⁸⁵. Furthermore, various factors affect the noise level of the DAS data, including the coupling between the cable and the ground and proximity to anthropogenic or natural noise sources. Finally, a cable may be long enough to span regions with different wave physics, e.g., seafloor cable that extends onshore, offshore, and deep water regions²⁶⁶. These effects naturally split the full cable into multiple segments that can be separately analyzed, which further necessitates a data-sharing system that favors such a reading pattern.

3.3 Object Storage and Format for DAS

This study proposes to develop and test an object storage framework for DAS: we deploy S3-compatible object storage locally and on AWS and optimize data chunking for the performance of two cloud-optimized data formats, Zarr and TileDB (see Figure 3.1).

3.3.1 Object Storage for DAS

Object storage has seen increasing use as a framework for large geophysical archives. National archives of large datasets, such as data products from NASA, have migrated to cloud object storage to improve data access delivering over 100 PB on the cloud. The EarthScope Consortium Data Service (formerly IRIS DMC and UNAVCO) is migrating to AWS S3 Cloud Storage and will deliver almost 2 PB of data upon their merger. Paired with object storage is the necessity for a cloud-optimized data format. Choosing a hierarchical data format such as HDF5 requires high-memory virtual machines and for users to load or download the entire HDF5 in memory locally. Although HDF5 was invented for better data sharing, it can dramatically slow down DAS data sharing and scientific investigation in modern distributed computing environments.

Here, we design a proof-of-concept implementation of a DAS data platform using cloud-optimized data structures and an object storage abstraction. We set up a server (either local or remote) with the MinIO object storage and convert raw DAS data into cloud-optimized formats. We find that this data platform outperforms a traditional, dedicated storage server (with a file system) for three reasons. First, we avoid having a local copy on the compute server and allow data chunks to be loaded directly into memory from the remote storage. Second, both cloud-optimized formats are highly compatible with the

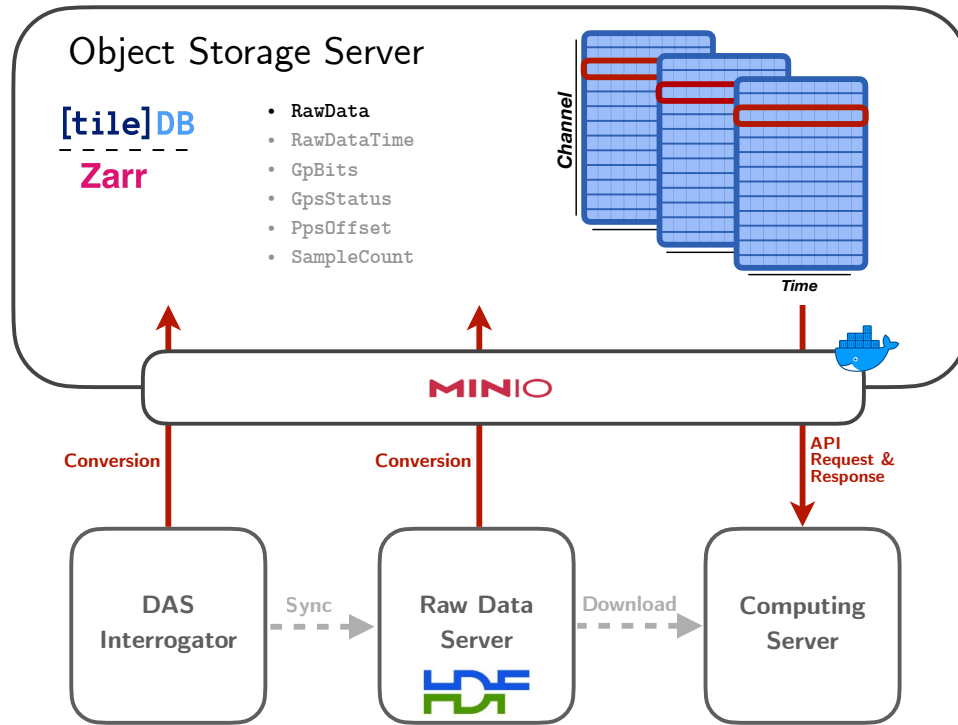


Figure 3.1: Workflow for DASstore: Distributed Acoustic Sensing (DAS) data is converted into Zarr or TileDB format in the MinIO object storage server. The computing server sends a data request to the MinIO object storage and loads the data directly into the memory. A conventional workflow would be downloading the HDF5 file from the raw data server and loading data to memory on the server where computing is performed (gray dashed arrows). The data attributes converted from the raw HDF5 files are stored in the new format, which in the Ocean Observatories Initiative (OOI) examples are: `RawData-2D` DAS data, `RawDataTime-1D` time axis, with four customized datasets: `GpBits`, `GpsStatus`, `PpsOffset`, and `SampleCount`. These datasets are specific to the OptaSense interrogator of the OOI data. The color version of this figure is available only in the electronic edition.

widely used Python data science and scientific computing environment, thereby enabling a user-friendly computing workflow with a shallow learning curve (e.g., NumPy⁹³, Dask²²⁰, and Xarray¹⁰⁸). Third, the S3-compatibility of MinIO facilitates an effortless migration to the commercial cloud, for example, for heavier computation.

3.3.2 Data Schema

Organizing data in a file is a first-order design choice for read/write performance. The chunking pattern is particularly determinant for formats like Zarr to perform well on object storage. A file may contain many data arrays representing different variables, which are split into chunks that form the minimum units to be read and processed in parallel. The size of a chunk is the minimum requirement for the memory of the computing unit. A small chunk will necessitate low memory, and a large chunk will necessitate a large memory. Thus, the number of processing units and memory allocation to each should be defined appropriately for the chunk size and file design.

Seismic data, in general, is multi-dimensional: it has a spatial dimension that may refer to the channel location and a time dimension that refers to the time series that the instrument record. The schema of data containers for seismology is designed to organize the data according to either dimension (e.g., ASDF organizes the data as a dataset per channel). DAS data is, by nature of being constrained along the arc of a fiber, a two-dimensional (2D) array with a time dimension and a channel dimension (i.e., arclength along the fiber). The HDF5 stored by the DAS interrogator has a simple data schema, where the `RawData` object is the 2D DAS data, and the `RawDataTime` object is the 1D time axis.

```
Acquisition
|--- Raw[0]
|   |--- RawData (47500, 12000), int32
|   |--- RawDataTime (12000, ), int64
|
|--- Custom
|   |--- GpBits (12000, ), uint8
|   |--- GpsStatus (12000, ), uint8
|   |--- PpsOffset (12000, ), uint32
|   |--- SampleCount (12000, ), int64
```

Three strategies exist for chunking a 2D array with spatial and temporal dimensions. The first strategy aims to optimize long-time series data investigations (sometimes referred to by the community as “large-T”, red arrow in Figure 3.2): the optimal way would be to have chunks that are elongated on the time axis and narrow along the spatial axis. A second strategy would be to focus on spatial investigations (sometimes referred to by the community as “large-N”, green arrow in Figure 3.2): the optimal way

would be to have chunks that are short on the time axis but long on the spatial axis. A third strategy would be to compromise between the first two.

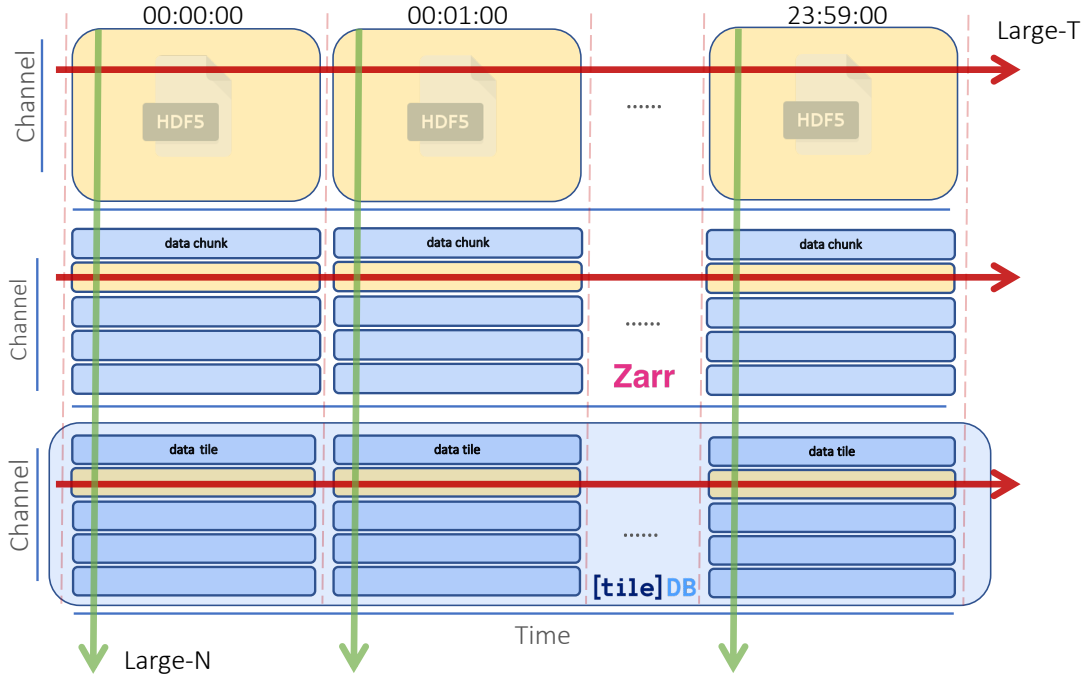


Figure 3.2: Chunking data in the three formats. The raw data is stored from the interrogators as a single file for each minute of recording. The Zarr schema separates each HDF5 dataset into a number of chunks; each chunk is a single object. The TileDB schema collects the entire data into a single array; the chunks are “tiles” and are concatenated in time. The red arrow represents the case for a “large-T” analysis with a subarray of the DAS cable. The green arrow represented a “large-N” analysis with a smaller time window. The color version of this figure is available only in the electronic edition.

3.3.3 Formatting the Metadata

There are three types of metadata in this context. The first is associated with each object in the object storage, which is used to identify objects and their properties. The storage server globally manages the object metadata and is part of the MinIO built-in design. The second is the format-specified metadata that is inherently part of the format, e.g., the `ARRAY_DIMENSIONS` field in Zarr metadata to be compatible with Xarray. The third is the research-defined metadata that describes the DAS experiment. Here, we focus on the convention proposed by the DAS Research Coordination Network (RCN), a National Science Foundation-supported community effort to define common metadata for DAS.

As of December 2022, the DAS-RCN defined five levels of metadata describing an experiment under this convention: *Overview*, *Cable*, *Fiber*, *Interrogator*, and *Acquisition*. These metadata are saved as key-value-pair attributes with the raw data. In addition, channel-specific information (*channel number*, *location*, and *uncertainty*) is saved in a separate comma-separated values (CSV) file.

3.4 Performance Test on DASstore

We design performance tests based on seismological use cases to optimize the DAS data chunking schema for the best performance of the object storage. We consider data 1) converting data from HDF5 to Zarr or TileDB and 2) implementing a feature extraction workflow. Three local servers are used for the benchmark tests: one raw data server storing HDF5 DAS data, one object storage server deployed in a single Hard Disk Drive (not distributed), and one computing server as the client to request the data and perform computing. Table 3.1 lists the hardware specification of these servers relevant to the optimization criteria, including an AWS EC2 instance for the test on the cloud.

Server	Network	CPU	RAM	Main Storage
Raw Data Server	1 Gbps	96	754 GB	16 TB HDD
Object Storage Server	10 Gbps	16	92 GB	16 TB HDD
Compute Server	10 Gbps	112	1007 GB	1 TB NVMe
Elastic Compute Cloud (m5.xlarge)	Up to 10 Gbps	4	15 GB	8 GB SSD

Table 3.1: Servers used in the tests: role, network, CPU cores, RAM, and primary storage. These specifications emulate a researcher-level DAS setup and include an example AWS EC2 instance for cloud benchmarking.

We conducted these tests using the Ocean Observatories Initiative (OOI) DAS experiment that was conducted in the first week of November 2021 on two ocean bottom cables (north cable and south cable) of the OOI Regional Cable Array²⁹⁸. We focus on the records of a 95-km long segment of the south cable, which goes from the land station westward until an optical repeater at a water depth of ~1500 meters. The data was recorded by an OptaSense interrogator unit with a gauge length of 51.05 m, a channel spacing of 2.04 m, 47,500 channels, and a sampling rate of 200 Hz. The data is stored in HDF5 files every minute (~1.2 GB each), and 7.1 TB was collected. Almost 50,000 channels are rather large for typical DAS experiments²⁵⁴, and reading a single HDF5 file only gives 1 minute of data and

already requires large memory. To reconstitute longer time series requires selecting subarrays (groups of channels) within the file, which is computationally expensive using HDF5. This makes OOI DAS data ideal for testing our platform.

We design two performance tests to emulate potential use cases. First, a writing test evaluates the costs of converting the DAS HDF5 data into Zarr or TileDB and whether it can be done in near real-time. Second, we conduct a reading test that evaluates the computational costs of a minimal seismological workflow that performs simple feature extraction (peak amplitude) after pre-processing (detrending, demeaning, filtering). This second test aims to represent the use case of selecting a group of channels (e.g., a subarray of the DAS cable) and extracting features. More specifically, we extract ten-minute subarray data with 5000 channels and apply the typical processing steps of detrending and tapering to each channel, followed by a second-order $[0.01, 1]$ Hz Butterworth band-pass filter and output the maximum amplitude of each channel. We perform these tests on our local Linux servers (see Table 3.1), emulating the case of a DAS experiment and archives being run by an individual investigator’s computing system.

3.4.1 Baseline

We choose a baseline test that emulates how researchers traditionally access remote HDF5 data. We first download the OOI DAS data from the storage server to the computing server. The total downloading time depends mostly on the ethernet bandwidth between the two servers. Using a single process that saturates the full bandwidth, it takes 110 seconds to download all files (11.7 GB, 106.3 MB/s on average). The feature extraction workflow takes an additional 12.9 seconds on average with a single process. We also chunk the data into HDF5 files with varying chunk sizes (byte shuffle filter and LZ4 compressor enabled). Then we download and run the feature extraction workflow separately on the compute node. This baseline test demonstrates that data download is a bottleneck and particularly inefficient when selecting several channels out of the array.

3.4.2 Writing Test: Data Conversion Costs

Zarr	TileDB
------	--------

Chunk/Tile Size	Writing Time (sec)	Number of Objects	Writing Time (sec)	Number of Objects
12000×1	1349.0	475004	196.8	33
12000×2	894.1	237504	193.2	33
12000×3	833.1	158344	196.5	33
12000×4	695.4	118754	197.3	33
12000×5	551.9	95180	202.4	33
12000×10	460.8	47504	-	-
12000×50	279.4	9504	-	-
12000×100	326.3	4754	-	-
12000×500	248.8	954	-	-
12000×1000	242.4	660	-	-

Table 3.2: OOI DAS data conversion from HDF5 into Zarr and TileDB formats. HDF5 files are converted directly to the object storage from the raw data server. The green numbers highlight the writing time shorter than the data duration, indicating the configurations that would allow real-time conversion.

To test the I/O performance of the platform, we first test data conversion from the raw data server to the object storage. We sequentially convert 10 HDF5 files on the raw data server to the object storage in Zarr and TileDB formats. We use a single process to ensure that neither network nor disk I/O is saturated. The test demonstrates the feasibility of collecting and converting DAS data in real-time. The test also represents the canonical workflow for users interested in earthquake waveforms: each HDF5 file is only 1 minute of data, typically shorter than the seismic waveforms of moderate-to-large-size tectonic earthquakes. To measure the expense of converting and hosting DAS data in these formats, we track the conversion time, array size, and the number of objects after conversion. We do not test on AWS S3 servers to emulate the case of researcher-level data acquisition without edge computing capabilities.

We first test using different chunk-size configurations. We define a single Zarr array and concatenate DAS data along the time axis. To have a more straightforward data schema and have the Zarr compatible with more existing tools (e.g., Xarray), we group all datasets that are described in Figure 3.1 (10 minutes of data and all channels) and flatten the hierarchical structure of the HDF5 file. Zarr saves each chunk as an individual object (each 1-min long). For a fixed data size, choosing a chunk size determines the number of objects after conversion. The test begins with a small chunk (12000×1, equivalent to 1 minute of data at 200 Hz), and the conversion takes ~22 minutes with ~0.48 million objects (each a single channel

for 1 minute of recording or 25 KBs). As we increase the chunk size to 1000 channels per chunk, each object size increases, the number of objects in the Zarr array decreases, and the writing time decreases. Table 3.2 demonstrates that the data conversion would become as short as the duration of the record (e.g., near-real time) for a small chunk size of 5 channels per object.

Selecting an optimal chunk size is critical for Zarr to perform well. A larger chunk size could bring more network overhead and slow down reading. On the other hand, small chunks have better reading performance, but many small objects could slow down the object storage as data grows. This latter latency may be due to our choice of hardware and may not exist on AWS S3. It is recommended to have at least a 1 MB chunk size, corresponding to grouping 50 channels in our case. The size of the Zarr array after conversion is almost equivalent to the original HDF5 files using a byte shuffle filter and LZ4 compressor. Thus the conversion does not create more storage needs for the storage.

Next, we test the TileDB format using different tile size configurations. We concatenate DAS data along the time axis for a 10-min window and store it in a single TileDB array. Similar to the chunks in the Zarr format, a tile is the atomic unit of reading and writing for TileDB. This array has 10 tiles of 1-min data for each channel group. For the OOI DAS, we define the time dimension as an unsigned 32-bit integer (uint32), which allows $2^{32} - 1$ time index in the array (equivalent to ~ 248 days for a 200 Hz sampling rate). We use the same byte shuffle filter and LZ4 compressor as the test for Zarr when writing the data, which generates slightly bigger objects after conversion. As a result, TileDB generates only 33 objects for all tile size configurations: 10 objects correspond to 10-time data (called fragments, including data and data schema) and 2 additional objects per writing operation that store the changes and commits. In our case, we do not have data that gets updated over time and do not need to leverage the data versioning capabilities of TileDB. Unlike the Zarr format, the conversion keeps relatively steady at about 20 seconds as the tile size increases.

3.4.3 Reading Test: Seismological Use-Case of Feature extraction

We request a 5000-channel subarray (adjacent arrays in space and the data) on the computing server from the storage server to conduct the feature extraction workflow using the same 10-minute OOI DAS data in Zarr and TileDB format. We detrend, taper, and [0.01, 1] Hz Butterworth band-pass filter each

channel data. Here, we use Open-MPI to distribute the channel indexes to each process and parallelize the workflow using a different number of processes. The results are shown in Figure 3.3. Each run reads the same amount of data (5000 channels for 10 minutes) and is distributed over a variable number of processes.

The performance tests of Zarr indicate a sweet spot in terms of chunk size and the number of MPI processes for optimal read time. A small chunk (under 50 channels) requires large query times but decreases as a function of MPI processes. This may be attributed to the accumulated object storage latency, as a small chunk size corresponds to more S3 requests to get the same amount of data. On the contrary, a large chunk read is faster, but the network overhead and compute times increase with the number of MPI processes. Grouping more channels in a single object effectively increases the data granularity, potentially creating more network overhead, especially when a smaller subset of the data is requested (e.g., as processes increase). A similar pattern is visible on the AWS test (see Testing the Migration to Commercial Cloud).

TileDB is fully parallelized internally. The format indexes the array and supports byte-range requests, and there is minimal extra network overhead. The performance with respect to tile sizes up to 5 channels was almost identical throughout the test.

3.4.4 Testing the Migration to Commercial Cloud

We test our platform on AWS. We upload the OOI DAS data to AWS S3 and query data on an m5.xlarge EC2 instance (general purpose, 4 vCPU, 15 GB RAM) from the S3 bucket. We use 12000×5 tile size for TileDB and 12000×50 chunk size for Zarr, which we found optimal between read and write tests. Figure 3.4 shows the reading test implemented on AWS. The results are similar to those implemented through MinIO and match well with a strong scaling fitting (dotted line in Figure 3.4a), assuming the network transmission time stays constant.

We identify three advantages of using the commercial cloud service to host the DAS data. First, there is usually no cost for the data transmitted into the cloud (no ingress cost), and storing data on the cloud for short periods of time is relatively cheap. For example, it costs \$ 25 per month per TB on AWS S3; for a 1-day experiment, the storage costs would be less than \$1 per day. Second, cloud storage services do not

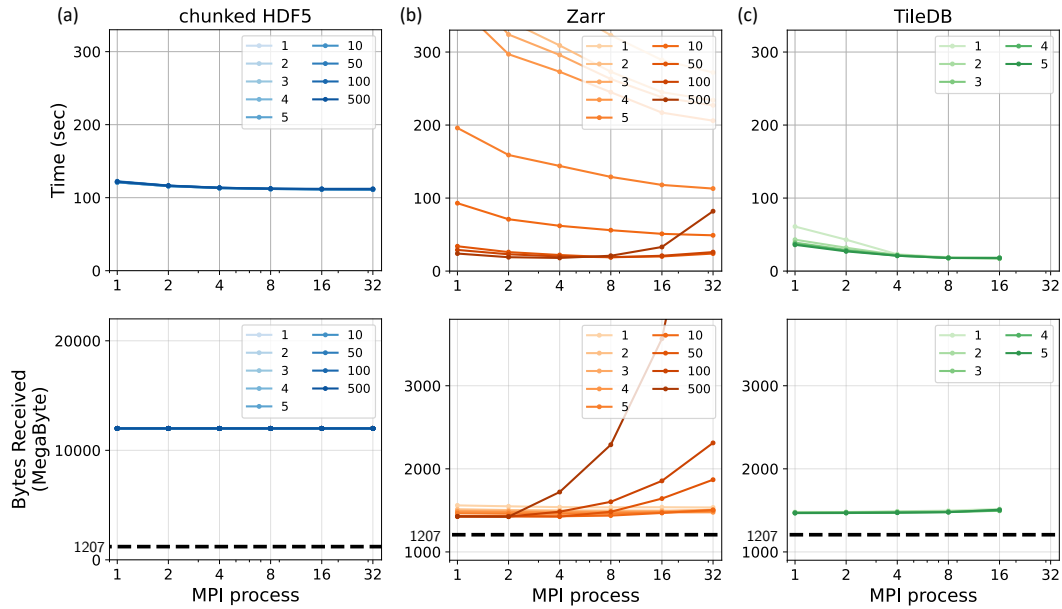


Figure 3.3: Results of the reading test for (a) chunked HDF5, (b) Zarr and (c) TileDB format. For chunked HDF5, the compute server downloads the entire HDF5 file, reads a subarray of 5000 channels and then performs the feature extraction on each channel. The compute server requests a subarray of 5000 channels from the object storage and performs the same feature extraction workflow. The upper panels show the time to finish the test, and the lower panels show the total bytes received by the compute server. The line color indicates the chunk/tile size for Zarr/TileDB arrays. All lines are overlapping with each other for the chunked HDF5 test. The black dashed line in the lower section shows the theoretical data size after compression. The color version of this figure is available only in the electronic edition.

impose a space limit, which is different from local storage, where the quotas are set by the hardware or the cluster centers. Therefore, cloud storage is an ideal repository for write-once-read-many scientific data. Third, commercial cloud storage has better back-end support and can handle a higher data request rate than the local object storage deployment (e.g., MinIO introduced in the previous section).

For most seismological applications, almost no network bottleneck exists between the computing instance and the object storage within the same AWS region, even on the lowest-tier virtual machine. If the computing is performed in the same region of the data, the data transmission is free. This is equivalent to having a massive file storage server with fast data query rate.

3.5 Application: SeaDAS-N Cross-correlation

We next carry out a standard ambient noise cross-correlation analysis using DAS data. We use the continuous records of an urban DAS experiment, SeaDAS-N, collected by the UW FiberLab between April

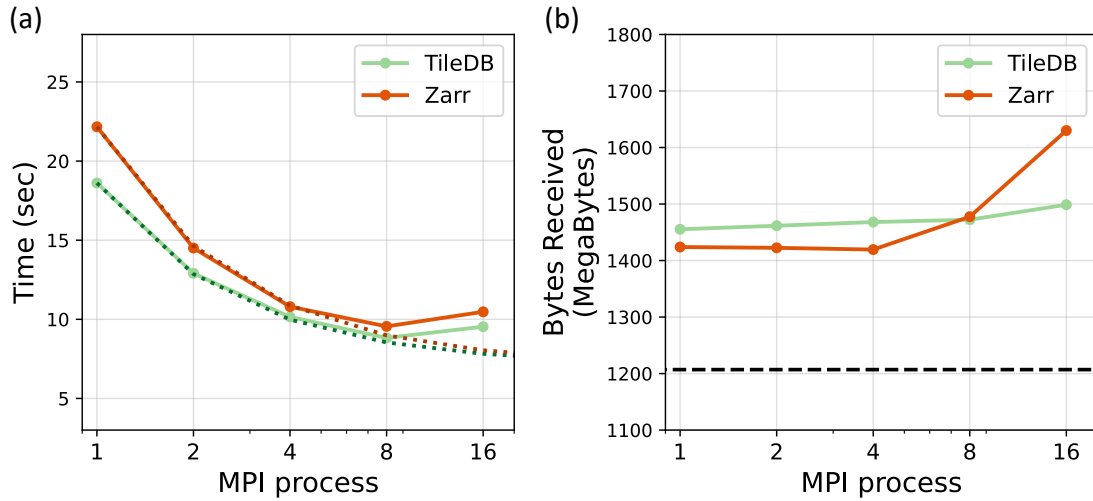


Figure 3.4: Results of the scaling reading test on Amazon Web Service (AWS). Panel (a) shows the reading time averaged over 5 runs. The dotted line indicates a strong scaling fitting, assuming the network transmission time stays constant. Panel (b) shows the total bytes received from the object storage averaged over 5 runs. The line color indicates the data format (Zarr and TileDB). The black dashed line in the right panel (b) shows the theoretical data size after compression. The color version of this figure is available only in the electronic edition.

2022 and March 2023. The dark fiber used is owned by the University and runs from the Atmospheric and Geophysics (ATG) building in the UW Seattle campus to the UW Bothell campus. The fiber runs mostly underground, but the cable segments are above ground, which can be observed in the raw DAS amplitude and confirmed by independent distributed sensing measurements (temperature and strain sensing). The data was collected using a Sintela Onyx interrogator v1.0 with 2,089 channels (4.78-meter channel spacing, 9.56-meter gauge length) at a 100 Hz sampling rate. Figure 3.5 shows one-hour SeaDAS-N data as an example.

We select data from December 2022 and perform a typical workflow for ambient noise seismology applied to DAS data. This type of analysis is common for seismometers to construct surface waves and perform tomography^{41,133,218,231,240,313,326} or allow for monitoring changes in the subsurface^{42,60,221,237}. For DAS data, most cross-correlation analyses have used a subset of the data^{266,280}, and only a few examples exist today of full cable cross-correlations²²¹.

We test the seismological use case of ambient noise seismology with our data platform on AWS. One month of SeaDAS-N data stored on local drives is converted to Zarr and uploaded to an AWS S3 bucket

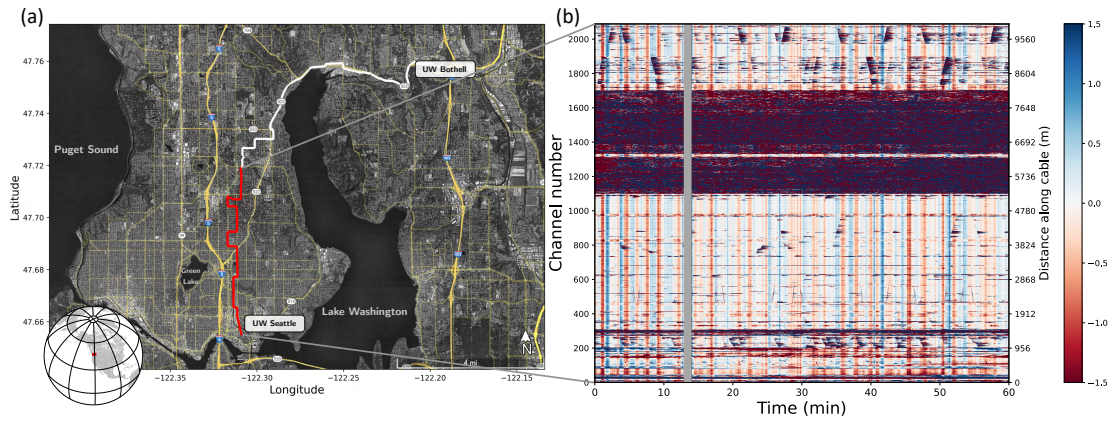


Figure 3.5: (a) Map showing the location of the SeaDAS-N cable (solid line) connecting Seattle and Bothell campus of the University of Washington, United States. Only the red segment was interrogated in December 2022 for data collection. (b) One-hour SeaDAS-N raw data of all 2089 channels (in radians unit, a data volume of ~ 2.7 GB). The image starts 2022-12-15 at 13:30:00 UTC. The gray region indicates a one-minute data gap. The clipped data (channel ~ 1100 -1700) is from the segment where the cable is above ground. The color version of this figure is available only in the electronic edition.

in the us-west-2 region (Oregon). One minute of 100 channels is chunked into a single object on S3. It takes 5 hours to upload 1.7 TB SeaDAS-N data using 4 processes. A modified version of NoisePy¹¹⁴ is containerized and submitted to the AWS Batch service for parallelization. Using all 2089 channels (~ 9.98 km), the correlation of 1-month data is split into 248 small tasks. We distributed the tasks using AWS auto-scaling: a service that automatically manages the type and number of EC2 instances based on the job requirements. Each task contains the following workflow: 1) requests 3 hours of SeaDAS-N data from S3, 2) processes the ambient field with one-bit temporal normalization and spectral whitening, 3) calculates the cross-correlation, and 4) linearly stacks the cross-correlations for each hour of data. In contrast to using HDF5 as the local intermediate data product as designed in NoisePy, each task writes the 3 hourly-stacked cross-correlations directly to another S3 bucket in TileDB format. Figure 3.6 shows a daily stacking of the correlation function using channel 500 as the virtual source.

All tasks run on AWS memory-optimized (R-family) EC2s in the us-west-2 region (co-located with the data). The typical cost (r5.large; 2 vCPU; 15 GB RAM) is \$0.0348 per hour on average, using preemptible SPOT instances (unused virtual machines sold at a much lower rate). On average, it takes 24 hours per task to compute 393 million cross-correlations, with ~ 98 billion correlation operations performed in total. As more tasks are completed, the auto-scaling group automatically shuts down idling

instances to optimize cost. Since the data I/O is fully implemented through our data platform, no extra local storage (“block storage”) is required for each computing instance. In addition, there is no cost during data uploading and transmission within the cloud, but it costs on average \$1.3 and \$2.8 per day storing 1.7 TB DAS data and 3.7 TB correlation function on the cloud. Researchers may download the data product (egress cost applied) or conduct further analysis on the cloud. Overall, we spent $\sim \$430$ to cross-correlate and download the stacked data. While SPOT instances are preemptible, in our experience the small instances (r5.large) are rarely recalled. When they are recalled, batch automatically restarts new instances to relaunch the job that was interrupted. Furthermore, minimizing each job run on batch decreases the risk to lose intermediate results.

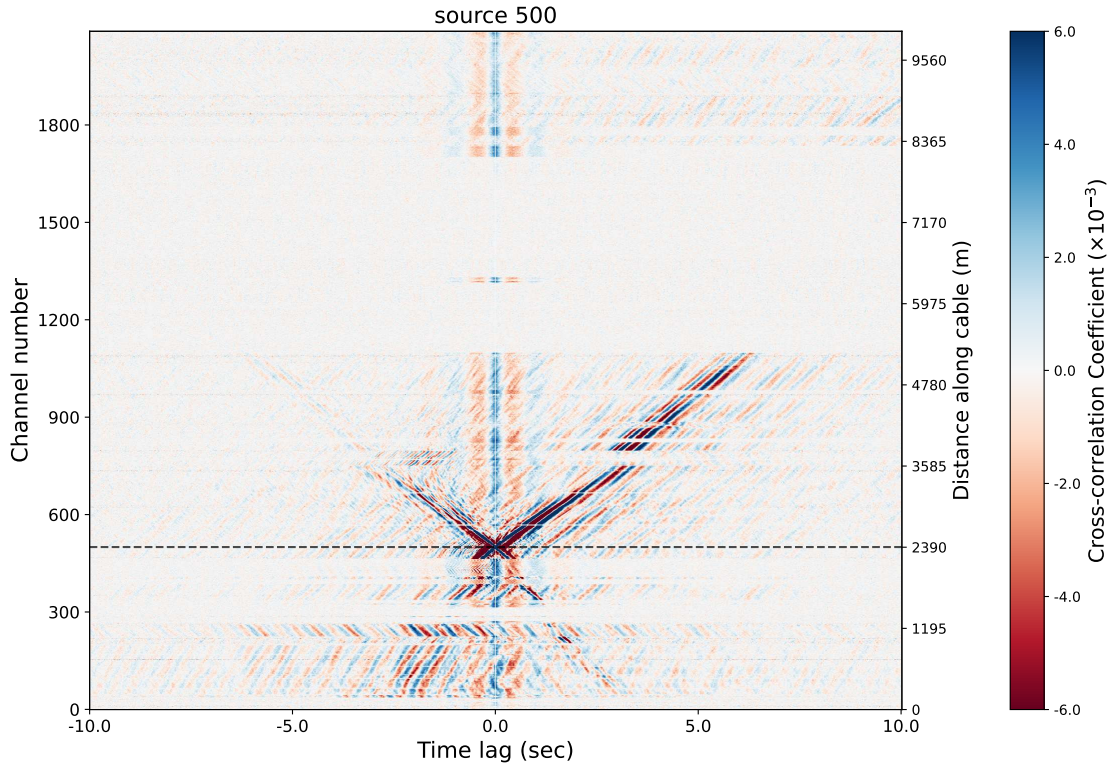


Figure 3.6: An example of ambient noise cross-correlations functions of SeaDAS-N using channel 500 as the virtual source. The channel number is labeled on the left y-axis, and the distance along the cable is labeled on the right y-axis. The black dashed line indicates the location of the virtual source. The color version of this figure is available only in the electronic edition.

The cross-correlation between channel 500 and all other channels of the DAS cable segment (see Figure 3.5) is shown in Figure 3.6. We filter the data between 1 and 20 Hz. We find a typical cross-

correlation image: waves propagate from the virtual source at about 400-500 m/s. The most likely wave type in this context is surface waves. Given the frequency band, it is not unusual to see slow-moving surface waves in shallow sediments, particularly in this area²⁵⁵. DAS correlations also often exhibit coherent, zero-lag signals that are instrumental and can be removed with an F-K filter²⁶⁶. Parts of the cable are above ground (e.g., between channels 1100 and 1700), and therefore there is no correlation between channel 500 and these channels. Additional analysis of these data may involve a more careful analysis of the wave-type (Rayleigh versus Love waves)⁷², the generation of dispersion curves, and shear-wave inversion for a velocity model.

3.6 Conclusions

This study presents the first cloud-native workflow for Distributed Acoustic Sensing (DAS) research. We developed a cloud-optimized data platform using object storage and cloud-optimized data formats applicable to DAS seismological research. We attempted to represent a range of seismological use cases, focusing on writing (I-input) tests for the archives and reading (O-output) tests for the users. Our benchmark tests demonstrate that one-time costs for data conversion from HDF5 (a common DAS data format) to Zarr or TileDB are manageable in near-real time.

Data chunking is a first-order controlling factor in I/O performance. The I/O performance for Zarr indicates a sweet spot of 50 to 100 channels per chunk to be manageable for object storage for a typical 200 Hz sampling rate DAS experiment. Performance scaling in I/O is also favorable for Zarr and TileDB. TileDB natively parallelizes I/O. We distribute I/O using MPI for TileDB, and we demonstrate its good scaling performance up to 16 concurrent processes, after which latency becomes cumbersome, and our single disk object storage may not respond well under such concurrency. In whichever case, the performance has been dramatically improved. Researchers may find the optimal format and chunk/tile size that favors their data through a similar I/O benchmark and back-end specification.

This study focuses on performance using cloud services. We anticipate that users will seek to lower costs by minimizing compute time, memory requirements, number of vCPU in the thread. However, long-term storage of 100TBs data sets on the cloud is still cost prohibitive, there we have demonstrated a way for individual researchers to archive DAS data. We demonstrated an example using ambient noise

cross-correlations. We developed the workflow using an existing Python package (NoisePy) that we modified by moving intermediate data products from local storage (i.e., the most common workflow) to cloud storage, which greatly improved the performance.

The integration of DASstore with other DAS community tools in Python can be straightforward: DASstore is only a package for data query and data is returned as NumPy arrays, a generic type for Python processing. We will use GitHub as the conduit for collaboration to initiate issues or discussions, and to support new cloud-native projects.

Code and Data Availability

The scripts of benchmark workflows are available through DASstore's GitHub repository (<https://github.com/niyiyu/DASstore>, last accessed August 2023). Chengxin Jiang adapted NoisePy to NoisePy4DAS, which we modified for cloud environments. The scripts to run the cross-correlation on AWS and instructions on how to run AWS Batch using auto-scaling are available at <https://github.com/niyiyu/NoisePy4DAS-SeaDAS> (last accessed August 2023). The OOI DAS data is available at <http://piweb.ooirsn.uw.edu> (last accessed August 2023). The SeaDAS-N data from December 2022 is freely accessible¹⁴³ through our Python API using the endpoint <https://dasway.ess.washington.edu> (last accessed August 2023), for which a tutorial can be found on the project repository page. The Kerchunk project is hosted at <https://github.com/fsspec/kerchunk> (last accessed May 2023). DAS Research Coordination Network is hosted at https://www.iris.edu/hq/initiatives/das_rcn (last accessed May 2023)

*This chapter was published as Ni, Y., Denolle, M. A., Shi, Q.,
Lipovsky, B., Pan, S., Kutz, J. N. (2024). Wavefield
Reconstruction of Distributed Acoustic Sensing: Lossy
Compression, Wavefield Separation, and Edge Computing.
Journal of Geophysical Research: Machine Learning &
Computation 1(3), e2024JH00024. doi: 10.1029/2024JH000247*

4

Wavefield Reconstruction of Distributed Acoustic Sensing: Lossy Compression, Wavefield Separation, and Edge Computing

4.1 Introduction

Distributed Acoustic Sensing (DAS) is a technology in seismology that uses Rayleigh scattering in optical fibers to measure the spatially resolved strain rate along a fiber³²⁷. The technology has been used for

earthquake early warning systems³¹⁸, event detection⁵, and structure monitoring³¹³ while exhibiting the same level of advancement for physical oceanography observations³⁰², marine mammal monitoring²⁹⁹, and tracking sea ice²⁰². Offshore sensing is particularly challenging, and seismic networks, if they exist, are limited to sparse and expensive ocean-bottom seismometers connected by cables (e.g., Ocean Observatories Initiative Regional Cabled Array). DAS has shown great potential in expanding the observation system by utilizing the seafloor fiber-optic cable previously used for telecommunication^{139,245,252,280,301}.

The large number of channels and the high sampling rate of DAS allow it to collect significantly larger data volumes than traditional methods associated with seismological studies. For a single DAS experiment with ~ 8000 channels and a sampling rate of 25 Hz, the interrogator unit generates ~ 3 GB data per hour, or ~ 72 GB/day. This rate is much higher than that of a modern regional seismic network. For instance, on 22 October 2023, the Pacific Northwest Seismic Network (PNSN, FDSN network code UW, UO) collected ~ 28 GB/day with its 1640 channels sampling at 100 - 200 Hz. Such data volumes pose a challenge to seismic networks and researchers due to 1) data transmission costs through telemetry or the internet, 2) the storage and dissemination of seismic data centers, and 3) the computational cost of data processing and analysis. Despite the development of innovative storage and archival systems and data analysis tools (e.g., Chambers et al.³⁶, Ni et al.¹⁹⁰, Spica et al.²⁵⁴), the challenge for real-time and computationally tractable data streaming remains. As a result, most studies on processed DAS data are performed after the data acquisition is complete, while very few are tested and deployed in real-time (e.g., Romanowicz et al.²²³, who implement channel decimation).

Lossless or lossy data compression methods can be used to reduce the size of DAS data. Lossless compression algorithms reversibly encode the original data in a compact space and are usually universal and commonly generalizable over a wide range of data modalities and formats. Such methods have been widely used in compressing images, e.g., Portable Network Graph (known as the PNG format), and scientific data formats, e.g., HDF5 (Hierarchical Data Format version 5), used by the interrogator units to store DAS data. Dong et al.⁶² proposed a state-of-the-art lossless DAS-specific compression algorithm based on a two-stage pre-processing and encoding step, reducing the storage requirement by 40%. In addition, Dong et al.⁶² is built as a plugin that could be incorporated into the HDF5 writer for real-time and post-hoc data-acquiring scenarios. In contrast, lossy compression methods allow a certain degree

of data distortion, and the information entropy does not limit their compression ratio. The discrete cosine^{251,257}, wavelet^{157,281}, and seislet^{39,78} transformation have been popular methods for seismic array data compression and sparsity representation. Very few of these methods have been applied to DAS data yet. Recently, Issah and Martin¹¹² compared the wavelet decomposition, the ZFP floating-point compression, and singular value decomposition (SVD) and analyzed their impact on the earthquake detection use case. They conclude that different algorithms are preferable depending on the compression ratio and their impact on preserving small seismic events.

A common strategy to compress DAS data is to drop a portion of the data (e.g., Romanowicz et al.²²³) and reconstruct the correlated original data using an interpolation algorithm or simulation governed by the underlying physics. Wavefield reconstruction has long been used in reflection seismology to mitigate the missing, or low-quality channel problem during the data acquisition. Recent studies are usually motivated by the concept of compressive sensing, aiming at generating regular and dense input for the downstream data processing^{79,80}. Similarly in earthquake seismology, previous studies apply the data (wavefield) reconstruction algorithms on long-period teleseismic body and surface waves^{40,176}, regional earthquake wavefields^{153,183}, and the exploration scenario²¹³ from point-based sensors (seismometers). These methods, when applied to large- N seismic arrays, have benefited earthquake ground-motion observation, holding the potential for earthquake early warning. The same concept is closely connected to denoising when a clean and smoothed wavefield is reconstructed, therefore suppressing or separating the noise from the original data. They can also be unified under the same framework if the missing data comes as a result of masking. Previous studies usually seek to find a low-rank representation of the original data with dimension reduction methods including 1) wavelet and curvelet transform as mentioned above, 2) dictionary learning^{38,67,73,154,267} and 3) Machine Learning^{100,247,276,311}.

Finally, the edge computing paradigm brings computing resources and storage closer to where the data is collected or gathered. Benefiting from the rapid development of computer engineering, this framework bypasses the need for data transmission and makes the data directly accessible on the edge server end. Data are processed on-site, and only applicable data products are transmitted. The framework exhibits three major advantages: 1) the demand and financial cost for a high-bandwidth internet have been dramatically alleviated, 2) applications are run in near real-time, and 3) the storage may also be

expanded by attaching the network-attached storage (NAS). However, for DAS, edge servers could still be practically expensive to deploy, especially when dedicated Graphic Processing Units (GPUs) are required for DAS-relevant Machine Learning training and inference, e.g., denoising, phase picking, and data compression. Some sites only have limited power, but running the DAS interrogator already requires enough power (e.g., 100-800 W), and therefore, power may already meet the demand of an edge computer. These limitations motivate the development of edge algorithms to be more computationally efficient for deployment.

In this work, we present various algorithms for offshore DAS data reconstruction and compression. Primarily motivated by the need to save storage space, we only use the term wavefield reconstruction in this study to describe the process of restoring the full channel observation of DAS data from spatially decimated inputs rather than reconstructing a unified continuous wavefield representation. We introduce the Cook Inlet DAS experiment and the data we use for analysis in Section 4.2. We discuss the model designs and training in Section 4.3. Then, we discuss the results of model training and validate the reconstruction on ocean waves in Section 4.4. We also experiment with alternative methods for the DAS data compression purpose in Section 4.5. In Section 4.6, we demonstrate seismological use cases for working with the residuals between the raw data and the reconstructed data.

4.2 Data

We use the data from the Cook Inlet DAS experiment conducted by the FiberLab at the University of Washington (UW). Two seafloor telecommunication cables owned by the GCI Communication Corporation are used that originate from the landing station in Homer, Alaska: the Terrestrial for Every Rural Region (TERRA) runs westward to the Iliamna Bay, near Augustine Volcano, and the southern span of the Kodiak Kenai Fiber Link (KKFL-S) that connects Kenai peninsula and the Kodiak archipelago. Beginning June 2023, both cables have been connected to a Sintela Onyx v1.0 interrogator unit in the GCI landing station for continuous data acquisition. The team has built a heterogeneous edge unit with a Network-Attached-System (NAS) for year-long continuous data storage (~ 160 TB), and two CPU and GPU edge computing servers for data processing. The interrogator unit saves the raw data at a 25 Hz sampling rate and further decimates it into 2.5 Hz; both raw and decimated data are

saved. There are 8531 channels using a 9.47 m channel spacing with a gauge length of 17.55 m between June and September and a gauge length of 23.93 m between September and December.

This experiment does not stream the raw continuous data back. Instead, we transmit back two-minute-long 25 Hz raw data of regional earthquake records. The events are chosen from the Advanced National Seismic System (ANSS) Comprehensive Catalog hosted by the United States Geological Survey (hereafter referred to as the USGS catalog), produced by the Alaska Earthquake Center (AEC), and selected based on their magnitude and distance to ensure a minimum peak amplitude of seismic waves. In other words, earthquakes may be distant with a high magnitude or local with a small magnitude and produce similar peak ground motions. We thus select events that meet the following criteria,

$$M \geq 10^{-1+b \log(R)}, \quad (2)$$

where M is the catalog preferred magnitude (mostly local, in some occasions body-wave and moment magnitude), and R is the hypocentral distance in kilometers. The equation estimates the amplitude of the earthquake arrival on the DAS observation while excluding the effect of radiation pattern, intrinsic and geometric attenuation, cable geometry, and coupling effects. Here, b is an amplitude coefficient: increasing b generally restricts the selection to events with higher amplitude. We use $b = 0.55$, a relatively loose criterion to stream more data for archiving purposes. However, we find only events with $b \geq 0.65$ will exhibit detectable impulsive arrival after a necessary bandpass filtering. After this selection, we download two-minute-long waveform data on a daily basis concatenated into one HDF5 file for each event and each cable, while the raw data stays in the NAS for future retrieval. As of 31 December 2023, 502 GB of raw data from 2615 earthquakes have been transmitted back to UW, an equivalent to 87 hours of continuous recording. No teleseismic event is included. Figure 4.1(a) shows a regional map of Cook Inlet with the seafloor cable and earthquakes that are used in this study.

While the original data remains unprocessed, in this study, we normalize the data by removing the mean and scaling the amplitudes by the standard deviation of each channel for the two-minute data segments. This is the first step in producing a principal component analysis (correlation) of data streams. This preprocessing step is reversible, but in the context of this study, it removes the recording discontinuity along the channel axis and further enhances the continuous ocean wave response on the

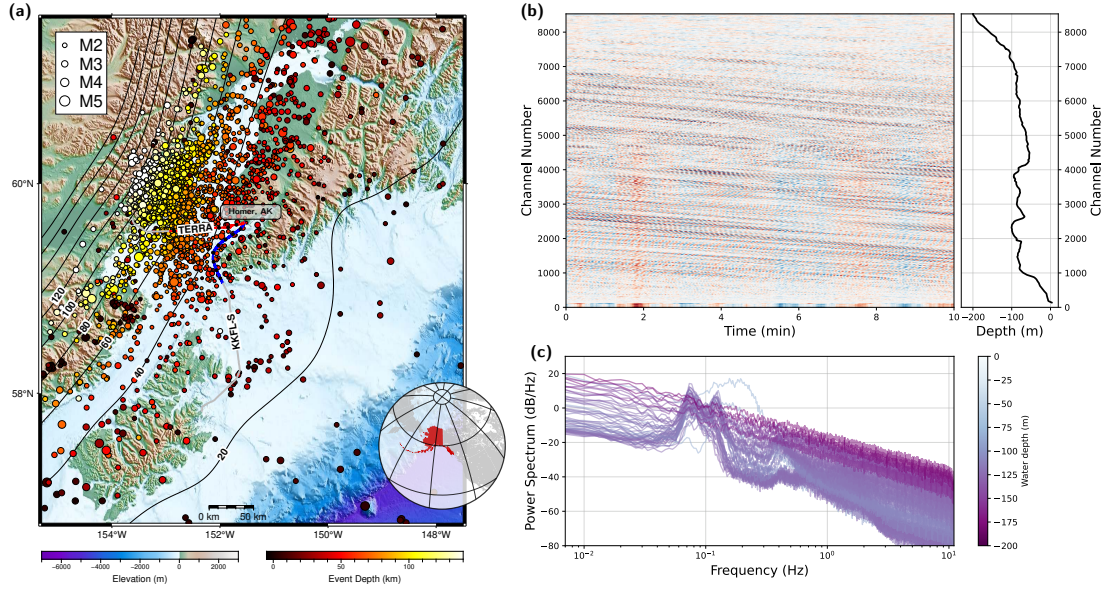


Figure 4.1: (a) Map of the Cook Inlet area in Alaska with two seafloor telecommunication cables shown: the southern span of Kodiak Kenai Fiber Link (KKFL-S) cable and the Terrestrial for Every Rural Region (TERRA) cable. 2615 earthquakes shown are selected from the USGS catalog between 1 July and 31 December 2023. The solid black lines mark the depth contour of the subduction slab with a 20 km interval⁹⁷. (b) A 10-minute recording of the raw data after demeaning and normalization on the KKFL-S cable, with corresponding water depth shown on the right and power spectrum in (c). Channel numbering starts from the landing station in Homer, Alaska. Ocean signals dominate the spectrum at about 10 s. The instrumental noise, a common mode, is clearly visible as a long-period vertical band observable along the whole cable and excessively dominating the data on the land cable segments (channel number smaller than ~ 100).

cable. In the following, all data is normalized as formerly described. Figure 4.1(b) shows a 10-minute normalized recording on the KKFL-S cable. The data appears dominated by landward ocean waves propagating at an approximate velocity of ~ 15 m/s. This is also confirmed in the channel power spectrum (Figure 4.1c): ocean surface gravity waves around the 9-15 s period. Weaker secondary microseisms also can be seen at 2.5 s, a little bit higher frequency than the globally observed microseism¹⁷ probably due to particularly shallow water in Cook Inlet. The data shown in Figure 4.1(b) does not contain earthquakes according to the USGS catalog.

4.3 Methods

4.3.1 Model Architectures

We use the SHallow REcurrent Decoder (SHRED)^{70,303} for a wavefield reconstruction model. SHRED was designed to take a short time series (sensory temporal trajectory) of a field sampled on a sparse network and reconstruct the current state of the field in a dense, 3D spatial array. The addition of the time dimension with the recurrent neural network permits the reconstruction of time-dependent processes. The SHRED architecture is a generalization of Principal Component Analysis (PCA) and separation of variables³⁰³. The adapted SHRED model takes inputs as the strain rate time series measured before and at the current time T_0 from N_{input} channels and reconstructs the spatial wavefield snapshot at time T_0 of N_{output} continuous channel, from which the inputs are decimated. The decimation is performed by extracting only selected channels instead of masking (zero-padding) so that the relative channel location is implicitly preserved. The SHRED model is very simple in its design and is made of two commonly used modules. Specifically, a stacked Long-Short-Term-Memory (LSTM¹⁰⁶) module projects the input time series from a sparse network, here a sub-sampled in space - sparse network - into a lower dimension latent space. A shallow decoder network follows the stacked LSTMs, and the latent space representation sensed from the sparse input is mapped to the full wavefield snapshot⁷¹. We use a multi-layer perceptron (MLP) neural network as the asymmetric decoder module with the ReLU activation functions⁴. The decoder links the hidden state space to the data space (wavefield) and implicitly contains a set of near-orthogonal basis parameterized by its weights. The overall architecture is shown in Figure 4.2 with two stacked LSTM over decimated DAS channels and an MLP decoder with 2 fully connected layers that transform the latent space from an intermediate layer that is half the size of the full output.

4.3.2 Training

In contrast to supervised deep learning methods, which usually require large volumes of human-labeled data for supervised training, SHRED, as a self-supervised model, relies solely on the raw data as the training label, and no task except mapping to the raw data itself is explicitly appointed during the training process. From a practical point of view, we select $N_{output} = 1000$ consecutive channel segments across the cable. Eight seconds of data (200 samples) from $N_{input} = 201$ decimated channels are input, and

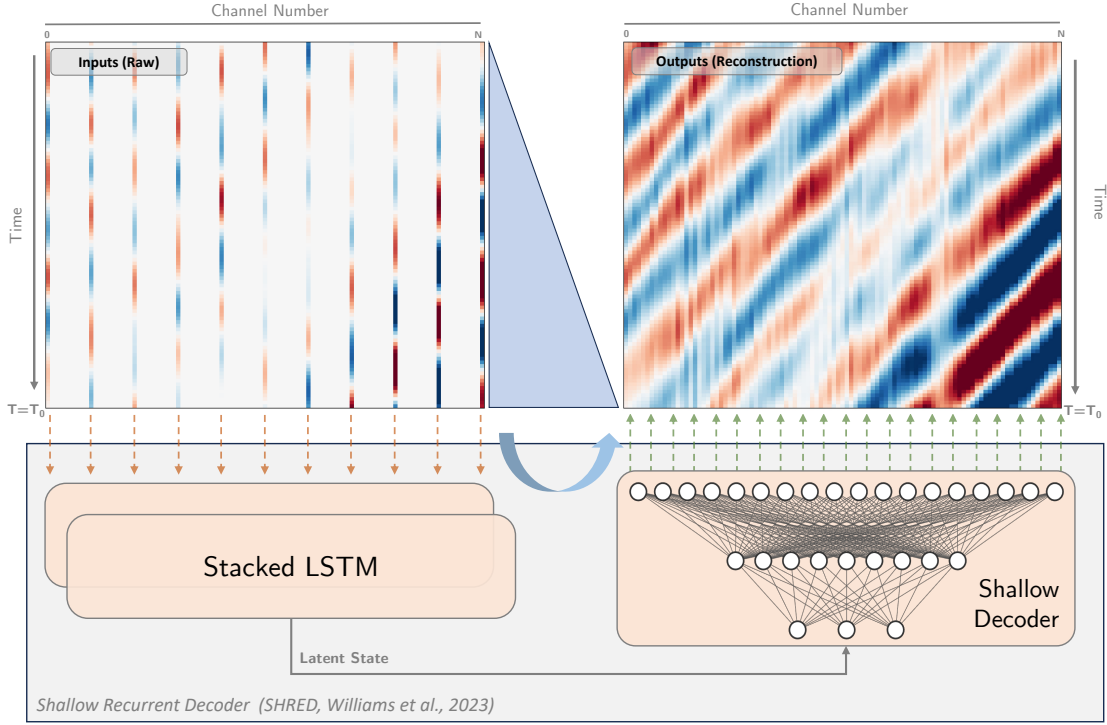


Figure 4.2: Structure of the wavefield reconstruction model, SHallow REcurrent Decoder (SHRED³⁰³). Two LSTMs (long-short-term memory) are stacked together as the input branch. The shallow decoder is a three-layer, densely connected multi-layer perceptron (MLP) branch. The LSTM branch maps the time series from sparse DAS channels into a latent state with a fixed dimension. The latent state is then projected to the reconstructed wavefield at all N channels through the shallow decoder network. Note that time series at and before time T_0 is used for input, while only wavefield at time T_0 is reconstructed.

the SHRED model reconstructs the wavefield snapshot at these 1000 channels. We choose this length of input series so that the dominant ocean wave is able to travel across the nearest decimated channels at ~ 15 m/s. The number of decimated sparse channels is also selected so that the first overtone artifact does not intermingle with the real data in the frequency-wavenumber (F-K) domain (see Section 4.4.3). Moreover, evenly channel decimation is favored over random decimation because SHRED does not have location encoding in the input layer. Without such implicit information on relative sensor location in the dataset, the model has to learn where the sparse observations are extracted from, which significantly increases the training difficulty. Finally, the decoder branch is defined as the latent state with 150 nodes (dimension of the latent space) and one hidden layer with 500 nodes before reaching the output layer with 1000 nodes (DAS channels). The model has ~ 0.97 million trainable parameters in total.

Data augmentation techniques have been commonly implemented to improve the dataset volume, diversity, and complexity from limited data and, thereafter, dramatically improve model performance. Data augmentation methods frequently used in Machine-Learning seismology include time window shifting and squeezing, random noise superimposing, masking, channel muting, and switching^{246,334,339}. Here, we augment the dataset by randomly sampling each file 500 times along both channel and time dimensions: each sample contains eight seconds of data from 1000 continuous channels, totaling about $2615 \times 500 \approx 1.31$ million samples. Decimation is made on these 1000 channels by evenly selecting 201 channels as input, which increases the channel spacing by a factor of five (now 47.35 m). Random windowing along the time axis enables the dataset to feature a diverse seismic characteristic depending on the sample start time relative to the earthquake arrival. Moreover, random channel selection avoids over-fitting the data pattern and quality on specific cable segments, e.g., ocean wave direction and velocity, which tend to be more localized. This step greatly improves the model’s performance and generalizability.

We randomly split the augmented dataset into three subsets for training (70%), testing (20%), and validating (10%). The dataset is well-shuffled so that the validating set represents the testing set well. We use Mean-Square-Error (MSE) as the loss function to measure the misfit between the reconstructed and the raw data. MSE only considers the point-by-point difference, usually dominated by the strong low-frequency ocean wave signal below 0.15 Hz. As a result, the model poorly resolves frequencies higher than 0.15 Hz, generating a clean and smooth ocean wave signal. We use the first-order gradient-based optimizer Adam¹¹⁹ to iteratively update model weights and minimize the loss function. We train the model for 40 epochs with an initial learning rate of 1×10^{-4} and linearly half the learning rate. All model training and testing are implemented with an Nvidia A100 80GB RAM GPU using PyTorch²⁰⁰. A mini-batch training strategy is used, each containing 64 samples from the augmented subset. The loss curve for the training process is shown in Figure C.1. A 10-minute example of raw, reconstructed, and residual data is shown in Figure 4.3 with the power spectra of selected channels, suggesting well-reconstructed data in the long-period band (< 0.15 Hz). The Peak Signal-to-Noise Ratio (PSNR) and MSE are reported and used to assess the reconstruction quality. The metrics are also used for the performance comparison tests, which help us select the model hyper-parameters (Table S1).

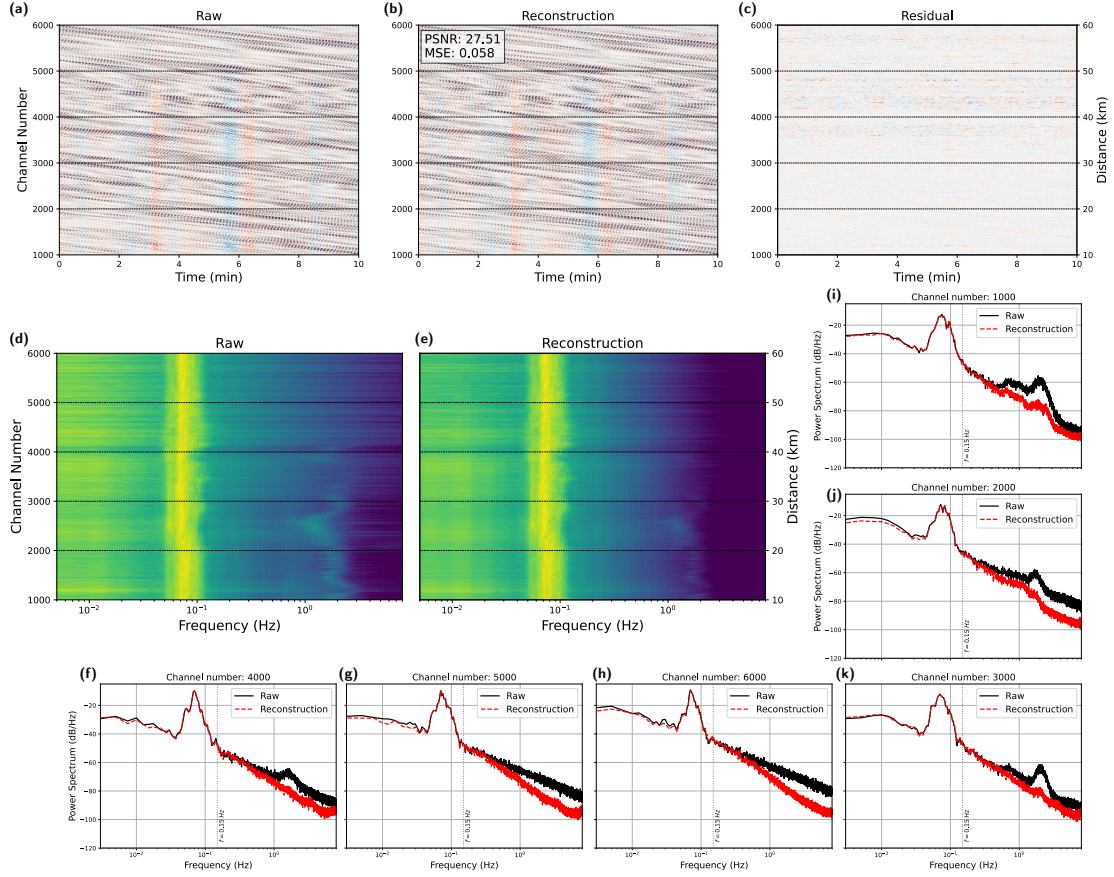


Figure 4.3: (a-c) A 10-minute example of raw, reconstructed, and residual data from the Cook Inlet DAS experiment on the KKFL-S cable sampling at 25 Hz. The same color scale is used for raw and reconstructed data, while the residual is boosted by a factor of 2. (d-e) The power spectrum of the raw data (black solid line) and reconstruction (red solid line) across the cables, with six selected channels shown in (i-k), suggesting a well-reconstruction in the frequency domain below 0.15 Hz.

4.4 Results

4.4.1 Model Performance

The SHRED model performs well on the wavefield data at the dominant frequency of ~ 0.15 Hz. Figure 4.3 reveals that the low frequency, slow propagating ocean waves are correctly reconstructed up to frequencies relevant to seismic waves. It takes ~ 51 seconds to reconstruct this 10-minute-long KKFL-S test recording on an Nvidia A100 GPU with 80 GB RAM. However, the spectral amplitude of the reconstructed waves at higher frequencies is lower than the original waveform. Furthermore, we observe that the maximum frequency of high-quality reconstructions decreases with optical distance. A possible

explanation is that ocean surface gravity waves have a lower amplitude in the distant parts of the cables, and instrumental noise increases due to the laser power attenuation. We show the reconstruction of seismic waves during recorded earthquakes of low and high magnitude: the 22 July 2023 M1.9 earthquake near Anchor Point, Alaska (USGS event ID ak0239btba15, Figure C.2) and the 16 July 2023 Mww7.2 earthquake near Sand Point, Alaska (USGS event ID at00rxvmw1, Figure C.3). The reconstruction model separates ocean waves from earthquake arrivals well, except for the long-period arrival from the Sand Point event, which overlaps with the instrumental noise traveling at the speed of light.

To alleviate these effects, we experimented with model design, training data sets, and loss function, though without success. We tested 1) convolutional layers in the decoder branch, 2) a loss function that penalized the reconstructed Fourier amplitude spectrum to force the training to recognize high-frequency signals, and 3) filtering training data to frequencies above 0.1 Hz and testing the reconstruction. Improvements did occur only when more sensors were included, which reduced the compression ratios and defeated the goal of the effort.

4.4.2 Generalizability

We trained and tested the SHRED model on the KKFL-S cable using training data from the earthquakes and tested it on the same cable. The test data did not overlap with the training and validation data set. The power of this method is that the model generalizes well over time and space within the same cable. This means that only one model needs to be trained for the reconstruction of an entire cable. We also trained the model using data fully from TERRA cable, and the performance of the new model is similar to that of KKFL-S (Figure C.4). Furthermore, the bi-directional vergence of the ocean gravity waves is reconstructed as expected. The model’s generalizability outside the training domain is also of interest. We implemented the wavefield reconstruction using a model trained by the KKFL-S cable and tested on channel 1000-2000 from the TERRA cable. We compared it with a model fully trained using data from TERRA. The measured PSNR and MSE (Figure C.5(b)) indicate that the model trained on the KKFL-S cable exhibits very good generalizability on the TERRA cable.

We also tested the transfer learning ability of the SHRED model. We fine-tuned the KKFL-S model with 20% of TERRA data at a smaller learning rate (5×10^{-5}). During the transfer learning, the

weights in the LSTM module are frozen. In that way, the latent space mapping is preserved and only the weights in the decoder module (MLP) are changed during further training. Figure C.5(c) shows an improved reconstruction after transfer learning. The performance is worse than the model fully trained by TERRA data Figure C.5(d), but this is not surprising.

4.4.3 Ocean Wave Reconstruction

We perform the frequency-wavenumber (F-K) domain analysis by applying the 2D Fourier transformation on the channel 1000-2000 data shown in Figure 4.3(a,b). The F-K transform has strong power when the 2D wavefield image has a high SNR moving wave. Non-dispersive waves, i.e., when all waves at all frequencies travel at the same speed, have strong amplitudes at constant k for all f . Dispersive waves, i.e., when waves travel with different speeds at different frequencies, have strong power in an F-K curve. With an 80 m average water depth h , the dominant power in the F-K domain well matches the ocean surface gravity wave dispersion defined by

$$(2\pi f)^2 = g \cdot k \tanh(kh), \quad (3)$$

where f is frequency, k wave-number, and g gravitational acceleration. The SHRED model well reconstructs the power of bi-directional ocean waves in this dominant frequency and wave-number range (Figure 4.4). We noticed that artifacts exist as an overtone in the larger wavenumber band when the wavelength of interest falls below the closest channel distance after spatial decimation, and the model can not uniquely resolve the wave among its multiple overtones as a form of aliasing. A test where only 101 out of 1000 channels are used for SHRED training exemplifies the effect that inadequately sampled data limits the minimum resolved wavelength. At this decimation rate, the effective channel spacing increases to ~ 95 meters, corresponding to a 0.066 wavenumber. This is where the first overtone artifact starts to show in Figure C.6, indicating the model confused on the right seaward propagating wavelength and frequency. In conclusion, with enough decimated channels, wavefield reconstruction is a possibility for compressing offshore DAS for the oceanographic use case presented above.

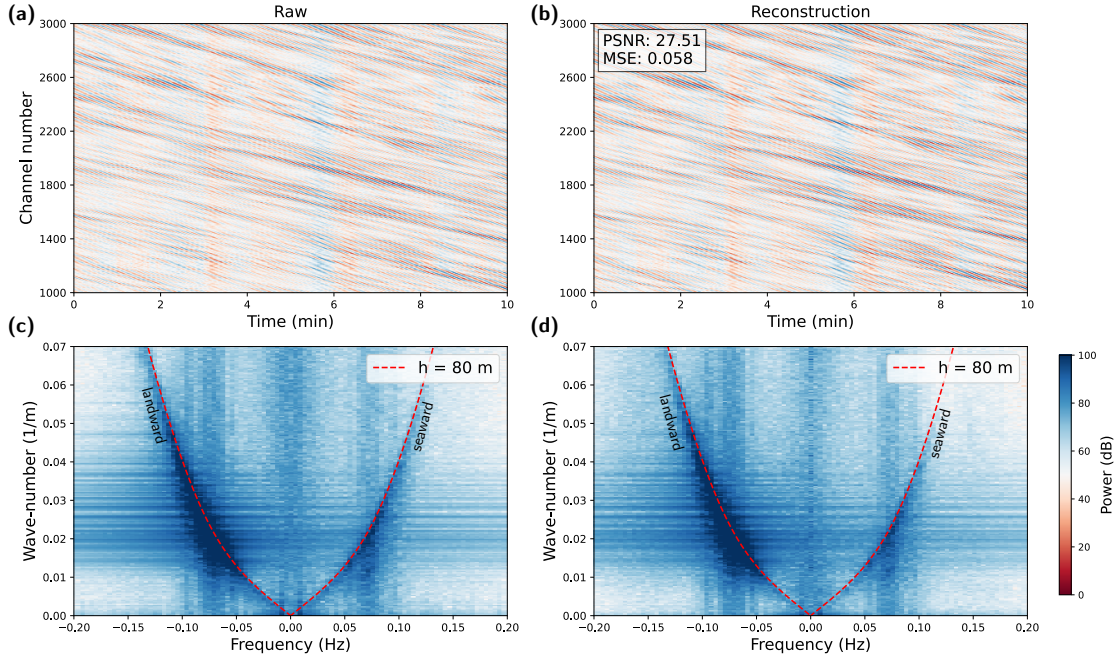


Figure 4.4: (a-b) A 10-minute example of raw and reconstruction data from the Cook Inlet DAS experiment on the KKFL-S cable sampling at 25 Hz. Data from a segment of 2000 channels are shown. The corresponding data in the frequency-wavenumber (F-K) domain is shown in (c) and (d), respectively. The dispersion relation for ocean waves under an intermediate water depth setting ($h = 80$ m) is plotted as red dashed lines.

4.5 Alternative Greedy Data Compression

As a complementary to the DAS data compression, we employ two other novel Machine Learning methods that have recently been used for data compression. Specifically, we adapt the networks from Sitzmann et al. ²⁴⁸ and Tancik et al. ²⁶², hereafter referred to as SIREN and RFFN (Random Fourier Feature Network), respectively. Both SIREN and RFFN are Implicit Neural Representation (INR) methods, widely used for data compression, dimension reduction, wavefield interpolation, and image fusion ^{69,197,262}. When the INR model is implemented on two-dimensional data, the coordinate of each pixel $(x, y) \in \mathbb{R}^2$ is mapped to the intensity ($I \in \mathbb{R}$ for a single-channel image). Due to the invariance of coordinates for a set of images with the same dimension, this mapping is data dependent, and thus, by its nature, the INR models cannot generalize across images: N images correspond to N set of models, and consequently requires N independent training (encoding) process. This lack of generalizability applies to both SIREN and RFFN used in our study. Moreover, data are not split into training and testing

sets: the weights are iteratively updated to fit the data as a whole without the requirement to validate generalizability through a test data set, and thus consistently “overfitting” the data. The loss function is used to be an indicator of the model generalizability, but it is now a measurement of model encoding quality and the amount of data loss in the reconstruction. Here, we apply the INR compression every 10 min for every 1000 channels. We use an RFFN of 3 layers with 160 units and a SIREN of 20 layers with 128 units. While RFFN uses the ReLU activation function, SIREN uses a sine function to connect hidden layers. There are $\sim 193\text{k}$ and $\sim 331\text{k}$ trainable parameters in SIREN and RFFN mapping the channel number and time index to the strain-rate value. These weights, transmissible as an encoded form of the raw data, are only 1.29% and 2.21% of what they can be reconstructed to (600 s at 25 Hz, 1000 channels), respectively.

We use the same data shown in Figure 4.3(a) to train the INR-based networks. The training dataset is selected as 10% samples of the DAS data array, and no data augmentation is applied. We train each model up to 20 epochs, or until the loss function drops below 1×10^{-2} . Similar to how we train SHRED, we use the Adam iterative optimizer and minimize the MSE loss function between the raw and reconstruction data. A 10-minute example of raw, reconstruction, and residual data is shown in Figure 4.5 encoded and then decoded by the SIREN and RFFN. Although the two models show different residual patterns, the MSE suggests an overall well-fitting across the entire data array.

It is worth noticing that a significant amount of time is required to train an INR model, as previously reported in Dupont et al.⁶⁸. Insufficiently encoded data leads to more reconstruction artifacts or data loss. Our test on the DAS data confirmed this, as the fitting time on an advanced GPU is comparably longer than the original data recording time. This implies that it may be impractical to deploy INR-based data compression on an edge machine as a real-time compression setting if the encoding takes as long as, or longer than the recording. However, in a post-hoc scenario where timeliness is not required, and more computation resources are available, the INR-based model exhibits a better compression rate. We report that further tests on these two algorithms were developed and performed on another offshore DAS experiment in Oregon, leading to the same conclusions¹⁸⁶.

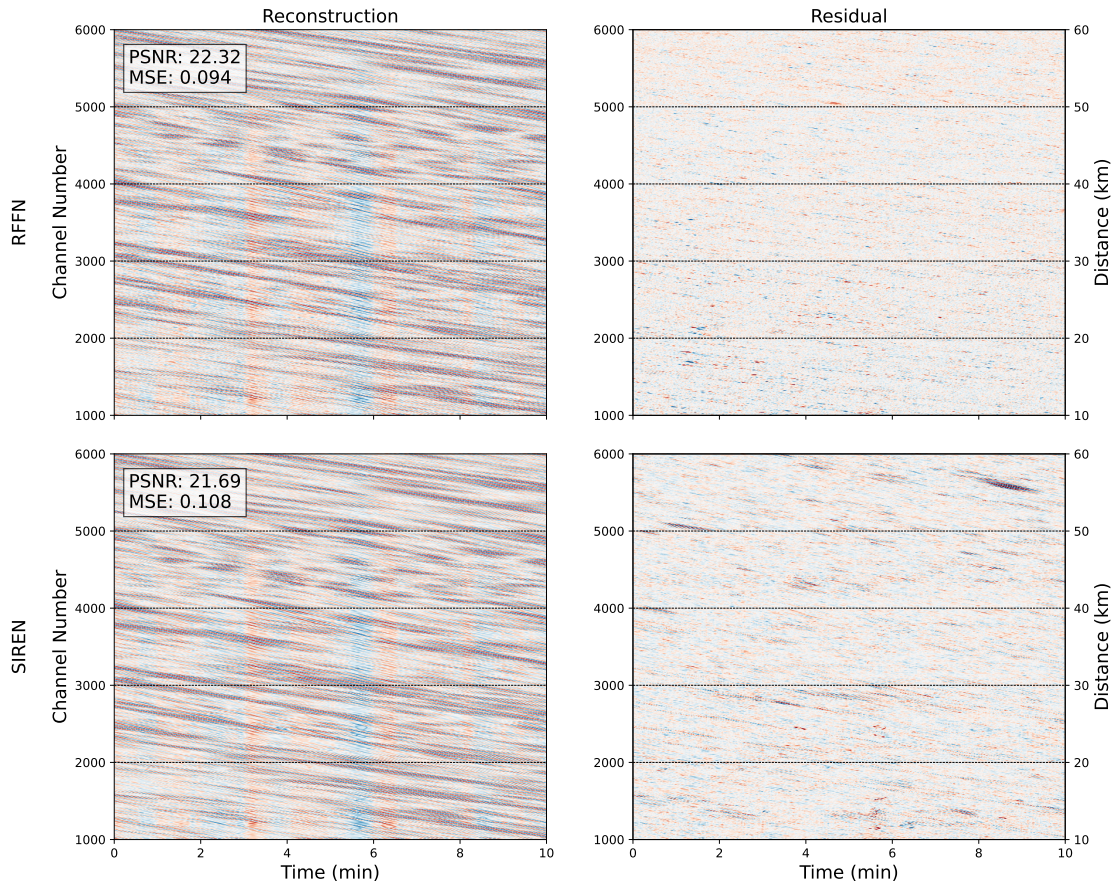


Figure 4.5: Greedy algorithm tests: DAS data compression and reconstruction using Implicit Neural Representation (INR) models: Sinusoidal Representation Network (SIREN) and Random Fourier Feature Network (RFFN) are tested using a 10-minute data from the Cook Inlet DAS experiment (Figure 4.3). Reconstructive waveforms and their residuals are shown in (a,c) and (b,d), respectively.

4.6 Offshore Seismological Use Cases

In this section, we explore offshore seismological use cases on the residual data by subtracting the reconstruction data from the raw data.

4.6.1 Earthquake Detection

Earthquake detection is one of the most important tasks in the earthquake monitoring process. It is the foundation of numerous event-based studies and applications, e.g., catalog building²²⁶ and earthquake early warning¹⁰. For both real-time data streaming and post-hoc data mining settings, a characteristic function is used to detect transient seismic wave arrivals (usually impulsive P- and S-wave onset) and

triggers additional analysis. The sensitivity to weak signals of an earthquake detection algorithm greatly affects the catalog completeness, while its robustness helps prevent false detections due to potential noise and waveform glitches. Similar criteria apply to the early warning algorithm, as a robust and sensitive characteristic function permits longer warning time and fewer false warnings. In traditional station-based seismology, detection algorithms are performed independently on each station, and triggers are clustered into an event through an aggregation process (for example, an association for phase picks). Although array-based methods have been proposed (e.g., Li et al. ¹³⁶), there is a pressing need for a scalable event detection algorithm designed specifically for DAS.

Kurtosis, defined by the fourth standardized moment, is a statistical value measuring the shape of a given distribution, and it has been used as a characteristic function for the state change of a system. Because the seismic phase onset usually generates non-Gaussian distribution on the wavefield measurement, the kurtosis can be used for event detection indicated by a sharp jump when the measuring window includes the onset signal ²². Here, we apply kurtosis measurement to detect earthquake events on DAS data. When combined with characteristic functions like kurtosis and STA/LTA for real-time event detection, the high-pass filter is commonly used to remove strong ocean waves and reveal the transient seismic arrival. This step can be computationally expensive, given the large number of channels in a typical DAS deployment, in addition to the lack of computing resources in the landing station or wilderness. We show that by simply subtracting reconstruction data from the raw recording, the residual wavefield exhibits mostly the transient earthquake arrivals and can be used for event detection straightforwardly without the need to implement a high-pass filter.

Figure 4.6(a-c) shows the 40 s raw, reconstructed, and residual recording of a regional $M_{3.5}$ earthquake on 29 December 2023 (USGS event ID ak023godcr3i). The theoretical P- and S-wave first arrival time on each channel are calculated using the ObsPy implementation of Tau-P ^{27,55} and the IASP-91 velocity model ¹¹⁸. We calculate the kurtosis using a one-second window and average over all channels. The kurtosis on the raw and reconstructed data shows little variability at the time of theoretical P- and S-wave arrival, possibly due to the weak energy arrived at the cable. However, on the residual data where ocean waves have been subtracted, the rapid increase of kurtosis around 22 s and 35 s coincides well with the P- and S-wave arrivals.

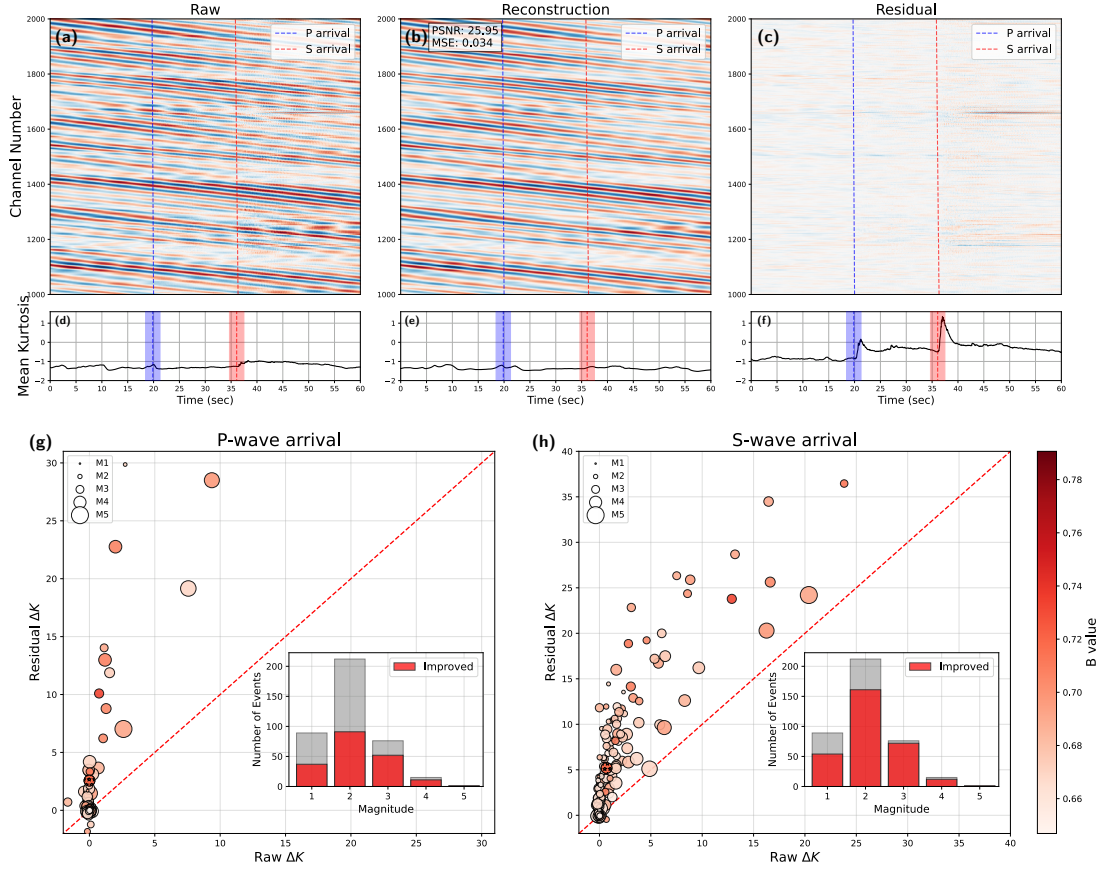


Figure 4.6: (a-c) Raw, reconstructed, and residual data of a 40 s DAS recording from the Cook Inlet experiment. The selected window contains P- and S-wave arrivals from a regional M_{3.5} earthquake on 29 December 2023 (USGS event ID ak023godcr3i). Dashed lines show the theoretical arrival time calculated from Tau-P using the catalog hypocenter location. (d-f) With the noise (gray), P- (blue), and S-wave (red) onset selected, we measure the kurtosis for each data on this shorter time window. (g-h) The improvement of peak kurtosis contrast in the P- and S-wave window before and after removing the reconstruction. As each circle corresponds to a single event sized by its magnitude, 48.8% and 76.3% events show an enhanced arrival on P- and S-wave, respectively. The aforementioned event is filled with stars.

To quantify the improvement in the kurtosis event detection, we evaluate the peak kurtosis in a three seconds pre-arrival (noise), P- and S-wave onset window on each channel (K_{noise} , K_P and K_S , respectively). Windows are shifted based on the theoretical arrival time in order to account for the move-out effect along the cable. For 393 events with $b \geq 0.65$, we calculate the kurtosis contrast ΔK in the P- and S-wave onset window with respect to the noise window ($K_P - K_{noise}$ and $K_S - K_{noise}$). The kurtosis contrast ΔK in the raw and residual data is used as the indicator of improvement after

removing the ocean waves. We selected and found 393 events when using $b \geq 0.65$. As Figure 4.6(g-h) shows, 300 or 76.3% of the events show an improved S-wave kurtosis contrast. The improvement on the P-wave window is less significant: 192 out of 393 events (48.8%) exhibit an improved kurtosis contrast. Moreover, the computation cost of such an operation is significantly smaller than running-window filtering when data are streamed. We conclude that using the residual between raw and reconstructed wavefield can be efficiently used to improve event detection on offshore DAS data, especially on the landing station where real-time pre-filtering is computationally limited.

4.6.2 Phase Picking

Seismic phase picking is the step of annotating the exact arrival time of seismic phase arrival. Earthquake source origin parameters like hypocenter location require precise marking of the onset time of P- and S-wave arrival. In most seismic networks, this step is usually performed by human analysts because automated pickers are usually only deployed for detection. However, picking phase arrivals on DAS is more difficult than that on the traditional broadband or short-period stations, particularly for three reasons: 1) DAS has only one horizontal component and is less sensitive to the P-wave traveling oblique or perpendicular to the cable, 2) the coupling of the cable to the ground may not be favorable, making phase arrivals less impulsive and much more noisy on DAS, 3) the number of channels is significantly larger than what human can handle as a routine work using visual picking methods. Machine Learning methods have been extensively used in seismic phase picking as they are more sensitive, robust, and efficient, especially at low SNR. Many Machine Learning phase pickers have been built to work on various data styles in terms of the instruments, noise level, and event type, and are made standardized by SeisBench³⁰⁷. However, few are designed for DAS data due to the lack of high-quality labeled datasets for model training³⁴¹.

We keep exploring the low-pass filtering effect of the reconstruction and perform an earthquake phase picking test on a one-minute windowed DAS data. The selected window includes arrivals from a regional M2.0 earthquake on 14 July 2023 (USGS event ID ak0238yvmdxr). Filtering is commonly applied in Machine Learning phase pickers to remove microseismic noise, e.g., PhaseNet³³⁷, Earthquake Transformer (EqTransformer¹⁷⁴), and EQCCT²³⁰. In contrast, we pick on the DAS data without

performing any bandpass filtering step to reduce computing time. We implement the ensemble learning seismic phase picker (ELEP³²³) using the EqTransformer as the base model and five pre-trained weights from SeisBench³⁰⁷. As Figure 4.7 shows, the ensemble phase picker can effectively identify and pick the arrival time of P- and S-wave from the residual data at relatively high accuracy, even though the base pickers are neither trained on DAS data nor fine-tuned using transfer learning. We also perform the picking using EQCCT on the same residual data, indicating that EQCCT is giving more picks for both P- and S-wave but at a cost of higher variance (Figure C.8). It should also be noticed that both ELEP and EQCCT produce P- and S-wave arrivals with up to 0.33-second mean bias compared to the travel time from Tau-P, suggesting an underestimated P- and S-wave velocity. This may be attributed to the 1D IASP-91 Earth reference model we use, which does not depict the structure of a subduction zone well, as the case in the Cook Inlet, Alaska. The same picking algorithm is applied to the raw data where no filter is applied (Figure C.9), which exhibits much fewer identified arrivals for both P- and S-waves, less accuracy, and misidentified phases. We show that Machine Learning phase picks on the DAS cable may potentially be a critical off-shore complement to the regional seismic network for earthquake monitoring as well as local tomography study.

4.6.3 Seismic Ambient Noise Interferometry

Seismic noise interferometry techniques can be used to extract dispersive surface waves from raw seismic signals, an important step in passive surface wave tomography (e.g., Nishida et al.¹⁹⁴, Yang and Ritzwoller³¹⁵). On land, such study is usually performed by arrays made of hundreds of nodal seismometers. In such cases, DAS is capable of reaching the same order of aperture while requiring far less workload on data acquisition. On the other hand, it is practically impossible to achieve such a high spatial resolution on the seafloor where deploying Ocean Bottom Seismometers (OBS) is challenging (e.g., Gao and Shen⁸¹, Liu et al.¹⁴⁵, Zhang et al.³³³). By leveraging existing seafloor telecommunication cables, the DAS system provides a unique opportunity for seismic ambient field observation and surface wave tomography²⁵².

While the long-period strong ocean wave has been effectively reconstructed and removed, the residual of the data exhibits coherent seismic ambient signal in the higher frequency band. Here, we use the same

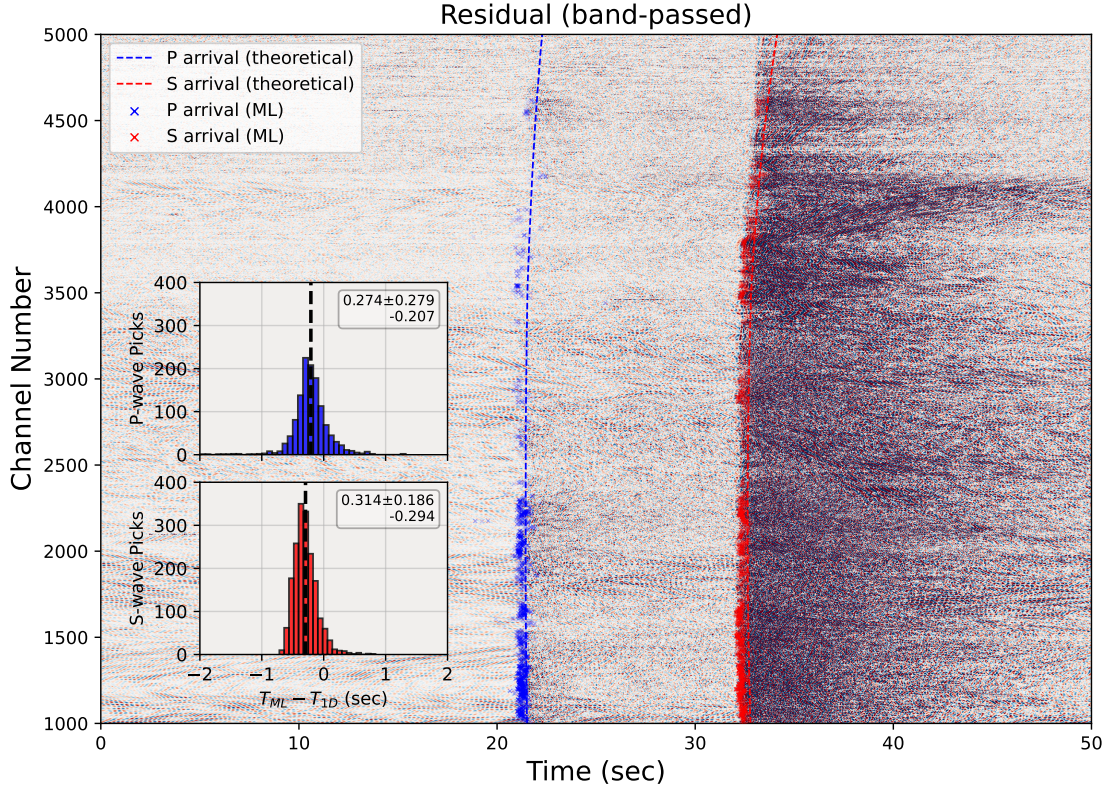


Figure 4.7: A 50 s residual DAS recording after removing the reconstructed wavefields from the raw data. The selected window contains P- and S-wave arrivals from a regional M2.0 earthquake on 14 July 2023 (USGS event ID ak0238yvmdxr). A band-pass filter is applied for illustration purposes only, enhancing the earthquake onset. We use the ensemble learning earthquake phase picker (ELEP³²³) to pick the arrival time on the unfiltered residual data, marked by the blue and red crosses. Dashed lines show the theoretical arrival time calculated from Tau-P using the catalog hypocenter location and IASP-91 reference velocity model. The inset shows the difference between the ELEP picks (T_{ML}) and the theoretical arrival time (T_{1D}) for P- and S-wave onset. The mean absolute difference, the standard deviation as the uncertainties, and the mean value of the difference are reported in the upper right corner.

NoisePy package by Jiang and Denolle¹¹⁴ and Ni et al.¹⁹⁰ to calculate cross-correlation functions for a 100-channel cable segment. Figure 4.8 shows the cross-correlation function of the raw, reconstructed, and residual data stacked over one day using channel 1000 as the virtual source. On the raw and residual data, strong Scholte waves can be extracted in the $[1, 3.5]$ Hz frequency band. Scholte waves are interface waves that travel in the subseafloor as interface waves between the fluid ocean and the solid Earth and are typically seen on dense seafloor surveys²⁶⁹.

At frequencies where we expect no good reconstruction (see Figure 4.3), we do not expect coherent seismic waves to be extracted in the reconstructed wavefield. Indeed, Figure 4.8(c) illustrates the non-

coherent nature of the high-frequency reconstructed wavefield. This suggests that one should not interpret the high-frequency content in the reconstructed wavefield. Nevertheless, the positive outcome of removing the reconstructed wavefield is that the coherence of the seismic wavefield in the residual is enhanced, especially at lower frequencies. In fact, the residuals are effectively “denoised” from the raw data and enhanced the seismic signals. We further comment on the dispersion with a factor of 4 difference between wavespeed at ~ 1 Hz (~ 1200 m/s) and wavespeed at ~ 2 Hz (~ 400 m/s).

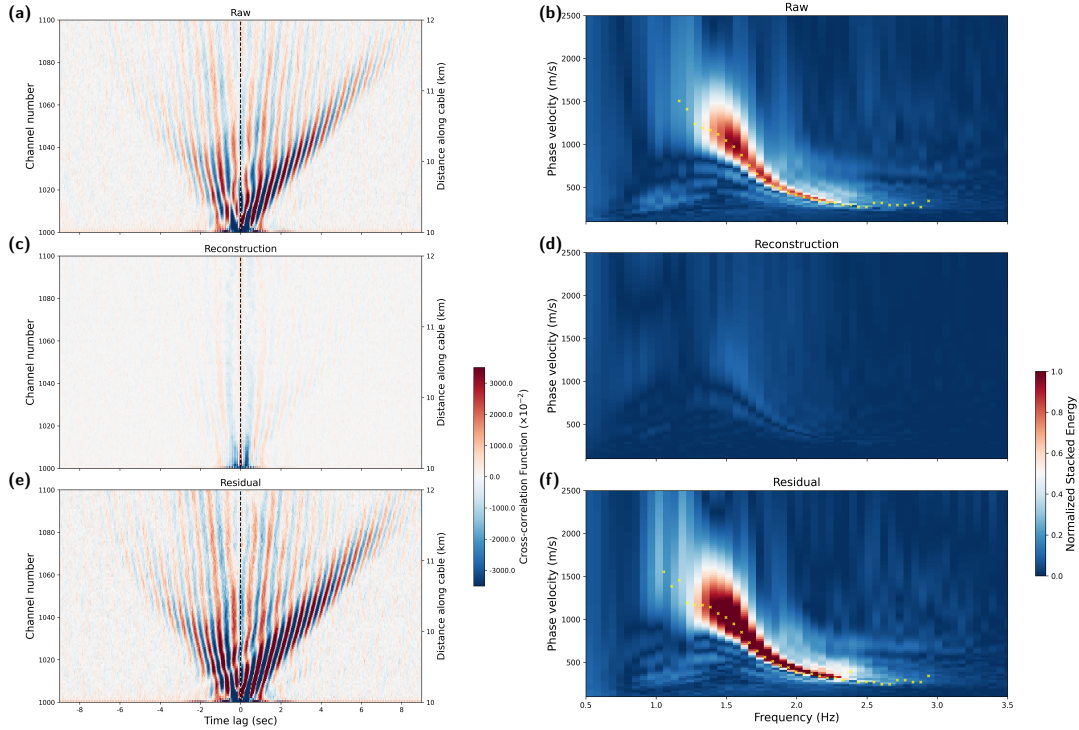


Figure 4.8: Correlated wavefield and dispersion analysis. Cross-channel correlation on (a) raw, (c) reconstructed, and (e) residual data. Correlation functions are calculated over 100 continuous channels and are stacked linearly over one hour, with slant stacked phase velocity dispersion map shown on the right, respectively. All cross-correlation functions and dispersion maps share the same color range, respectively, highlighting little leakage of coherent seismic ambient wavefield in the reconstruction. The dispersion suggests a highly dispersive fundamental-mode surface wave on this cable segment.

4.7 An Edge-Computing Framework

Here, we propose an edge computing workflow for real-time DAS data processing. The paradigm suggests bringing computing resources close to the data and enables time-sensitive low-latency applications. In the cable landing station where the interrogator is deployed, on-site SHRED inference reconstructs the ocean

wave and the subsequent residual data. The aforementioned low-latency applications could be performed for continuous event detection and phase picking. Moreover, the edge machine can concurrently calculate and stack ambient noise cross-correlation functions, with dispersion measurements sent back as a data product. Once needed, the same decimated data can be streamed back and reconstructed at high fidelity in the home institute for various oceanography studies. Our study is a pilot experiment of such a framework, with only 10%-20% of the data streamed back from the field site.

We further envision a reduced need for GPU- and other servers on the edge side. Indeed, it is possible to apply SHRED inference in real-time using a small single-board computer like a Raspberry Pi, thanks to the recurrence of the LSTM model and the simple structure of the decoder network. We experimented with a Raspberry Pi 4B (4GB RAM) that could perform ~ 80 reconstruction operations per second for 1000 channels in a recurrent inference mode with no quantization applied. The kurtosis measurement is also tested, suggesting that we can perform an earthquake detection algorithm on the real-time DAS data using only a low-cost computer.

4.8 Conclusions

This work demonstrates the use of innovative deep-learning methods to compress and reconstruct wavefields recorded by offshore distributed acoustic sensing. We find that SHallow REcurrent Decoders (SHRED) provide an effective framework to compress DAS data, enabling lower streaming rates, and being effective at reconstructing wavefield from ocean wave that composes the broadband DAS data. Two other novel Machine Learning methods are used for complementary data compression: Sinusoidal Representation Network (SIREN) and Random Fourier Feature Network (RFFN). Although both models exhibit no temporal and spatial generalizability and require more time on training, they show great data compression ability and are suitable for a post-hoc data-acquiring scenario.

Moreover, we demonstrated that SHRED does not reconstruct short wavelength (high-frequency) wave components, probably due to the sampling of channels and the lack of spatial encoding. Nevertheless, SHRED allows the separation of the total wavefield into the oceanic contributions from the seismic contributions. We illustrate seismological use cases that could be deployed on an edge computing unit to improve Earth imaging and seismic event detections in an offshore setting.

Code and Data Availability

We use PyTorch to develop the reconstruction and compression models²⁰⁰. The scripts and notebooks for DAS data downloading and model training are available at <https://github.com/niyiyu/DAS-reconstruction>. Earthquake data from the Cook Inlet DAS experiment is publicly available at <https://dasway.ess.washington.edu/gci/>. The USGS catalog is downloaded using ObsPy client²⁷. All figures are made with Matplotlib¹⁰⁹ and PyGMT²⁷¹.

This chapter was published as Ni, Y., Denolle, M. A., Thomas, A. M., Münchmeyer, J., Hamilton, A., Wang, Y., Bachelot, L., Trabant, C., Mencin, D. (2025). A Global-scale Database of Seismic Phases from Cloud-based Picking at Petabyte Scale. Seismica, 4(2). 10. doi: 26443/seismica.v4i2.1738

5

A Global-scale Database of Seismic Phases from Cloud-based Picking at Petabyte Scale

5.1 Introduction

Detecting earthquakes by picking P- and S-wave arrival times is fundamental to seismology. It enables rapid estimation of earthquake source properties, provides early warning for potential ground shak-

ing, and supports seismic hazard assessment. Picking the arrival time is also the first step in building earthquake catalogs. Traditional earthquake detection methods are typically unsupervised and rely on signal characteristics such as impulsivity, using techniques like the short-term average/long-term average (STA/LTA) filter⁹ or kurtosis-based approaches¹⁰⁴. However, these methods are highly sensitive to background seismic noise, limiting their effectiveness to small-magnitude events or recordings from particularly quiet stations.

Recent advances in seismic data processing techniques demonstrate that artificial intelligence (AI) and machine learning (ML) overcome these limitations and have shown high performance in earthquake detection^{173,203,225}, phase picking^{165,174,228,337,340}, and phase association^{161,174,227}. Supervised deep learning approaches to earthquake detection and phase identification require and have benefited from large, labeled datasets (e.g., several hundred thousand examples of P-waves, S-waves, and noise, along with the timing of body wave arrivals) for model training. These datasets are often compiled from analyst-reviewed phase picks cataloged by regional seismic networks and compiled by researchers for AI-readiness^{172,188,316,337,343}. Once trained, deep-learning-based detection and phase picking frameworks have significantly outperformed more conventional approaches in sensitivity and timing accuracy. They can detect arrivals in data with low signal-to-noise ratios and reliably pick arrivals to within less than 0.1 s^{174,337}, without requiring manual intervention for hyperparameters. Additionally, once trained, these models are inference-only, making them extremely fast and scalable. These successes, combined with the abundance of readily available continuous seismic data, facilitate large-scale regional and global data mining efforts.

Data volumes in seismology are expanding rapidly, with researchers now typically tackling datasets of TBs in size that require advanced computing strategies for analysis. Cloud computing is a promising infrastructure that can support this by enhancing the reliability, scalability, and accessibility of data⁸⁴, promoting reproducible and open science. Most importantly for users, cloud computing provides large-scale computing power close to the data archives, eliminating the need to download massive datasets and enabling users to run analyses in-place on the cloud using cloud-native tools (e.g., elastic computing, batch computing, scalable storage, and database services). We refer the readers to Ni et al.¹⁹² for comprehensive descriptions of various AWS resources commonly used for scientific computing,

including the ones used in our experiment. The Southern California Seismic Network (SCEDC) has been hosting a copy of its entire data archive on the Amazon Web Services (AWS) Simple Storage Service (S3) since 2019³²². The Northern California Earthquake Data Center (NCEDC) followed a similar model and architecture in 2024. The EarthScope Consortium is a non-profit organization that supports Earth science research by collecting, managing, and providing access to seismic and geodetic data collected worldwide from all United States (US) NSF-supported seismic experiments, a subset of the US regional seismic networks, and a selection of stations from global seismic networks³²⁵. It serves the scientific community by operating and maintaining networks of instruments, curating data and metadata, and delivering these to end-users. In recent years, the EarthScope Consortium decided to migrate data collection, archiving, and delivery services to the cloud.

Here we present results from one of the first large-scale, cloud-based seismic data-mining efforts. We processed approximately 1.3 PB of continuous seismic data from stations worldwide (see Figure 5.1). Leveraging modern cloud infrastructure, we developed a scalable cloud-native workflow to efficiently manage and analyze this extensive dataset. The sections below describe the waveform formats, cloud architecture, and compute resource utilization. We also describe the structure and contents of the resulting phase-pick database and provide public access through a web service. This work demonstrates the feasibility and advantages of using cloud computing for robust, high-throughput seismic waveform analysis. At the same time, the resulting database can serve as a starting point for the community to study seismicity globally. As in a typical catalog workflow, phase picking is the most runtime-intensive step. We believe that our database, providing global-scale, high-quality phase picks, has the potential to substantially accelerate seismicity studies worldwide.

5.2 Methods

5.2.1 Data

The NCEDC provides public access to continuous seismic waveform data through AWS, utilizing the AWS Open Data Sponsorship program. NCEDC continuous waveform data are hosted in the `ncedc-pds` S3 bucket in the `us-east-2` region (Ohio, United States). Files are organized by network, year, and day of year, and are stored in miniSEED format, with each file representing one day of data from a

single channel. Similarly, the SCEDC offers public access through the `scedc-pds` S3 bucket in the us-west-2 region (Oregon, United States). The EarthScope Consortium provides data access to credentialed users through an S3 Access Point, which gives fine-grained control over permissions and network access. While users interact with the Access Point much like a standard S3 bucket, internally, requests are routed through a Lambda function that handles S3 requests, enabling dynamic access management and custom logic. The combined archives represent 47,354 stations recording data between January 1, 2002, and March 31, 2025, for a total of more than 1.3 PB of continuous seismic data (Figure 5.1).

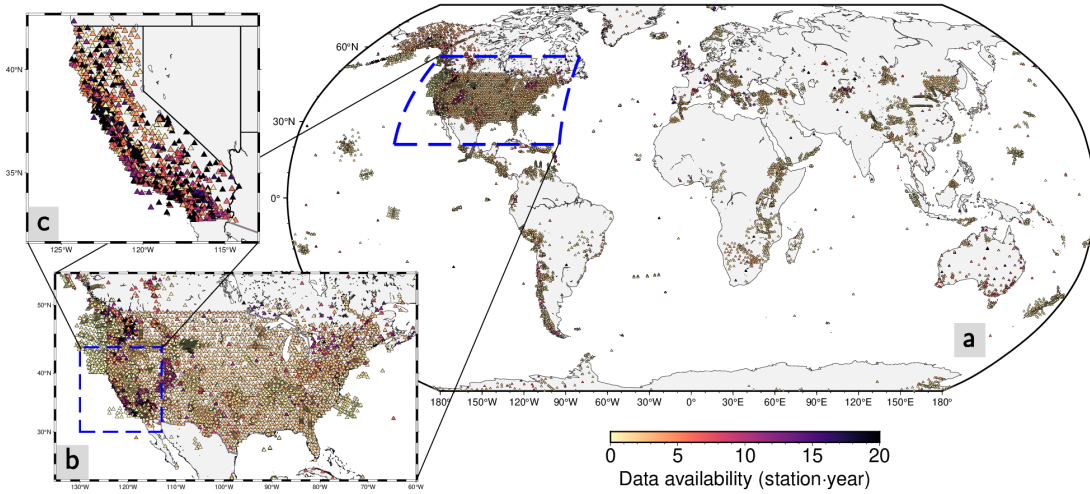


Figure 5.1: Map of stations, displayed as triangles and color coded according to the Data availability in station-years included in EarthScope, NCEDC, and SCEDC archives. Panel (a) shows the distribution of global station data availability. Panel (b) shows a detailed view of stations in the United States, while Panel (c) shows data availability in California.

5.2.2 Workflow

Several earthquake catalog building workflows exist^{214,261,285,330}, some of which have focused on cloud deployments^{124,342}. Here, we develop a scalable cloud-native workflow for seismic data processing designed for large-scale data mining using AWS services, which we illustrate in Figure 5.2.

The workflow begins with a user-specified list of station codes and a defined time range. A unique combination of 40 stations and a 20-day time window is paired as one job, which is submitted to an AWS Batch computing queue. We request 8 vCPU and 16 GB RAM for each job, with interruptible SPOT instances enabled. In contrast to the on-demand counterpart, SPOT instances utilize unused

AWS capacity with a discount (up to 90%) but can be arbitrarily recalled. With appropriate retrieval and checkpoint mechanisms implemented, SPOT instances effectively optimize the cost efficiency of our workflow. The submitted jobs stay pending until requested resources are supplied, automatically elevating queued jobs into the running state until the account quota is reached. With a 12,000 vCPU account quota, the computing queue allows 1,500 jobs to run in parallel. A running job first retrieves instrument response information from the EarthScope International Federation of Digital Seismograph Stations (FDSN) `fdsnws-station` service via the ObsPy library²⁷. A temporary user credential is requested from EarthScope to specifically enable EarthScope S3 access, while no credential is required for SCEDC and NCEDC. Waveforms are loaded directly from the S3 buckets by mapping the network code to the appropriate data center’s bucket, bypassing the need for middleware returning waveforms such as the FDSN `fdsnws-datasetselect` service. For this project, we processed all available seismic data from January 1, 2002, to March 31, 2023, across channels with the following codes: EH?, HH?, BH?, HN?, EP?, DP?, EL?, SL?, SH?, CN?. When stations host multiple location codes or channel types, we pick and record them separately in the workflow. We skip waveforms that are empty, embargoed, or contain over 50 gaps per component.

Once data is acquired, phase arrivals are identified using PhaseNet³³⁷, a deep learning-based algorithm for automatic P- and S-wave detection, trained on the INSTANCE dataset¹⁶⁵, implemented through the SeisBench framework^{180,307}. We extract ground velocity estimates around each peak if horizontal components exist. Each job’s output includes phase picks, job metadata, and checkpoint information during processing. These are stored in an AWS DocumentDB server, which functions as a NoSQL database for tracking job progress and storing results. This cloud-native architecture supports high throughput, fault-tolerant processing of large-scale seismic datasets, leveraging the scalability and modularity of AWS cloud services — separating data storage (S3), compute (Batch), and output management (DocumentDB).

5.2.3 Database

We employ the AWS DocumentDB to store intermediate and final data products. The DocumentDB service is a NoSQL database that manages and stores data in a structured, flexible format. The database

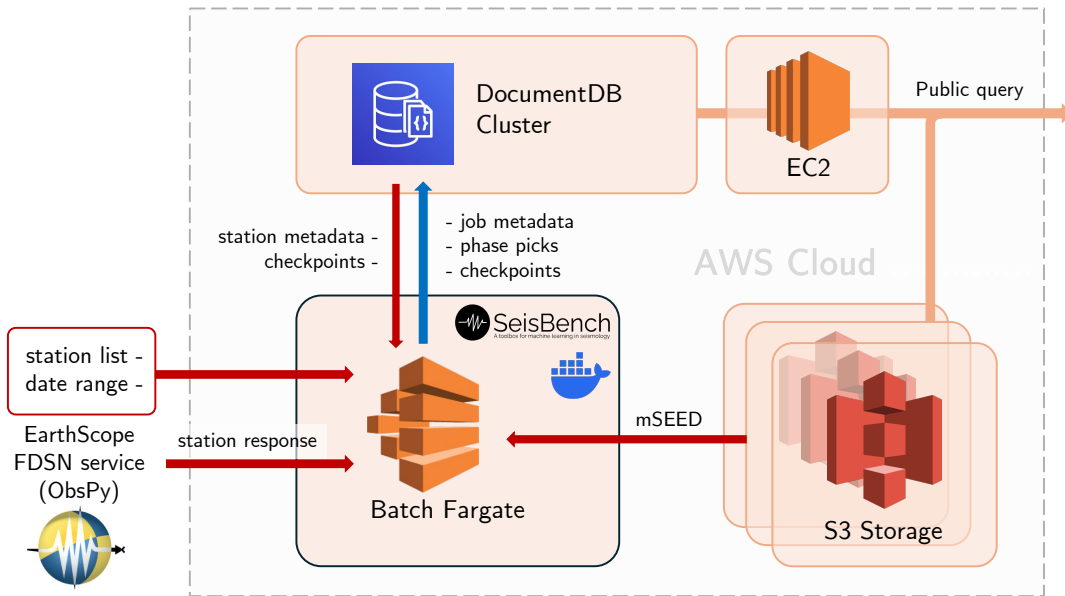


Figure 5.2: The scalable cloud-native workflow for seismic phase picking. Containerized jobs are submitted to AWS Batch, which loads miniSEED seismic waveforms directly from AWS S3 buckets. Phase arrivals were identified with PhaseNet through the SeisBench implementation. A DocumentDB cluster is employed to store job metadata, picks, and checkpoints. Finally, an EC2 instance is used to provide a public database query service.

is organized into several collections, each representing a different category of data relevant to the seismic data processing workflows we’ve implemented here. Each collection contains documents (JSON-like records) with a defined set of fields representing specific information pieces. The collections are also indexed to improve performance and to avoid duplicate picks.

The DocumentDB server organizes station metadata and processing outputs into distinct collections (Table 5.1). This design enables efficient querying and auditing of picking results and the exact software and settings used to generate them. The schema is flexible enough to query independent information, as station information, job configuration, and analysis results are stored independently but connected through unique IDs and timestamps. During this experiment, an I/O-optimized class instance (db.r6g.2xlarge, 8 vCPUs, 64 GB RAM) was employed as the DocumentDB server.

Collection	Field	Description	Type	Example
	trace_id	Station identifier	str	UW.SHW.01
	network_code	Network code	str	UW

stations	station_code	Station code	str	SHW
	location_code	Location code (if any)	str	01
	channels	Available channels	str	SH
	latitude	Latitude of station	float	46.19364
	longitude	Longitude of station	float	-122.23492
	elevation	Elevation in meters	float	1442.0
	start_date	Operational start date	str	1972.275
	end_date	Operational end date	str	3000.001
picks	trace_id	Station identifier	str	CL.SVD.
	channel	Channel code	str	HH
	start_time	Start of pick window	datetime	2012-01-01T02:45:37.400
	peak_time	Peak amplitude time	datetime	2012-01-01T02:45:38.270
	end_time	End of pick window	datetime	2012-01-01T02:45:38.610
	confidence	ML confidence score	float	0.48456
	amplitude	Signal amplitude	float	0.00027764
	phase	Phase type (P/S)	str	P
picks_record	run_id	Associated run ID	ObjectId	ObjectId(...)
	trace_id	Station identifier	str	CL.SVD.
	channel	Channel code	str	HH
	year	Year of record	int	2012
	doy	Day of year	int	1
	n_picks	Number of picks	int	12
	run_id	Associated run ID	ObjectId	ObjectId(...)
sb_runs	run_id	Run ID	ObjectId	ObjectId(...)
	model	ML model used	str	PhaseNet
	weight	Model weight name	str	instance
	p_threshold	P-phase detection threshold	float	0.2
	s_threshold	S-phase detection threshold	float	0.2
	components_load	Input component configure	str	ZNE12
	seisbench_ver	SeisBench version	str	0.8.2
	weight_ver	Weight version	str	2

Table 5.1: DocumentDB Collections and Schema Summary.

5.3 Results

5.3.1 Job Statistics

Mining the EarthScope dataset took less than three days, while the combined NCEDC and SCEDC datasets took less than 16 hours. Given the configurations detailed in Section 5.2.2, this experiment results in a total cost of $\sim \$15,000$. Since this was the first major data mining exercise on the EarthScope archive, job sets were launched manually, progressively, and actively monitored. Each set of jobs was intended to process a year of data recorded on all stations with channels matching those listed above. Figure 5.3a shows the progression of the jobs mining the EarthScope dataset, color-coded by the year the data was recorded. Figure 5.3c and e show the pending and running jobs as a function of time. Time periods where jobs were manually launched manifest as increases in the number of pending and running jobs. Since we used a quota of 1500 running jobs, the pending jobs decreased at an approximately constant rate. There are occasional abrupt decreases in the number of running jobs, for example, the one near 60 hours in Figure 5.3c, that correspond to times when stations are listed as available in the metadata service, but no data is actually loaded. Figure 5.3b, d, and f show the job ID, pending, and running as a function of time. For the NCEDC and SCEDC datasets, all jobs were submitted and queued simultaneously and not parsed by year, as they were with the EarthScope dataset.

5.3.2 Phase Picking

The mining exercise resulted in a total of 4.3 billion picks (2.8 billion P-wave picks, 1.5 billion S-wave picks), with the EarthScope, NCEDC, and SCEDC datasets containing 2.8, 1.1, and 0.4 billion picks, respectively. Figure D.1 shows the total number of picks broken down by the dominant network codes in each dataset. The NC network has the largest number of picks at nearly 1 billion, followed by UW at 0.5 billion, and CI at 0.4 billion. These differences in the number of picks among networks may be factorized into multiple aspects: 1) the amount of data available, 2) the tectonic settings where stations are installed, 3) non-tectonic noise signals that could be incorrectly picked, and 4) noise level that affects the performance of the phase picker.

Figure 5.4 shows examples of the daily number of picks as a function of time for ten stations around

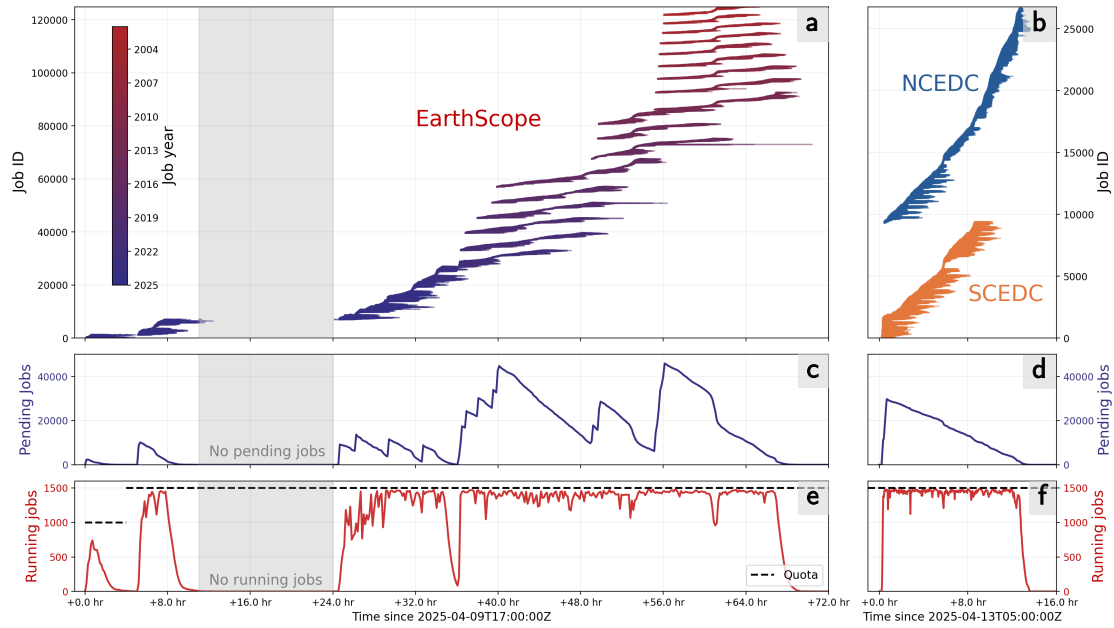


Figure 5.3: Detailed job run history for the EarthScope, NCEDC, and SCEDC dataset. Panel (a) shows the Job ID as a function of time, color-coded by year the data was recorded, for the EarthScope dataset. Panel (b) shows the job progression for the NCEDC and SCEDC datasets. Panels (c) and (d) show the pending jobs as a function of time. Panels (e) and (f) show the running jobs as a function of time. The horizontal dashed line represents the job quota.

the world. Station UW.BVW is a station near Beverly, WA, atop the Saddle Mountains. UW.BVW displays a strong seasonality of detections, with detection rates peaking in the late summer. IU.FURI is a station in southern Addis Ababa, Ethiopia. This station shows a large number of detections beginning in late 2024 and extending into early 2025. These detections likely correspond to a swarm of earthquakes that began in late September and produced 19 M_{5+} earthquakes, the largest of which was a $M_{5.9}$ on February 14th, 2025. IU.MAJO in Matsushiro, Japan, clearly recorded aftershocks from the 2011 M_9 Tohoku-oki, 2014 $M_{6.2}$ Hakuba, and 2024 $M_{7.5}$ Noto earthquakes, as well as many others. AK.MCK in McKinley Park, AK shows strong seasonality in detections and recorded the 2002 $M_{7.9}$ Denali earthquake. HV.KKO in Hawai'i records seismicity from the 2018 $M_{6.9}$ earthquake and ongoing eruptive activity from the Kilauea volcano. NZ.KHZ in Kahutara, New Zealand, recorded the 2013 $M_{6.5}$ Blenheim and the 2016 $M_{7.8}$ Kaikōura earthquakes and aftershocks. IU.QSPA at the South Pole shows a strong seasonality of detections and also records increased detections beginning in late 2024, possibly due to a series of M_{5+} events near the Balleny Islands and on the Pacific-Antarctic Ridge.

C.GOO4 at the Tololo Observatory, Vicuna, Chile, records the 2015 M8.3 Illapel and the M6.7 Coquimbo earthquakes. II.NNA records the 2007 M8.0 Pisco, Peru earthquake. Finally, CI.CGO records the 2009 M4.7 Inglewood, 2009 M5.8 Northern Baja California, the M6.4 and M7.1 2019 Ridgecrest, and the 2020 M5.8 Lone Pine earthquakes.

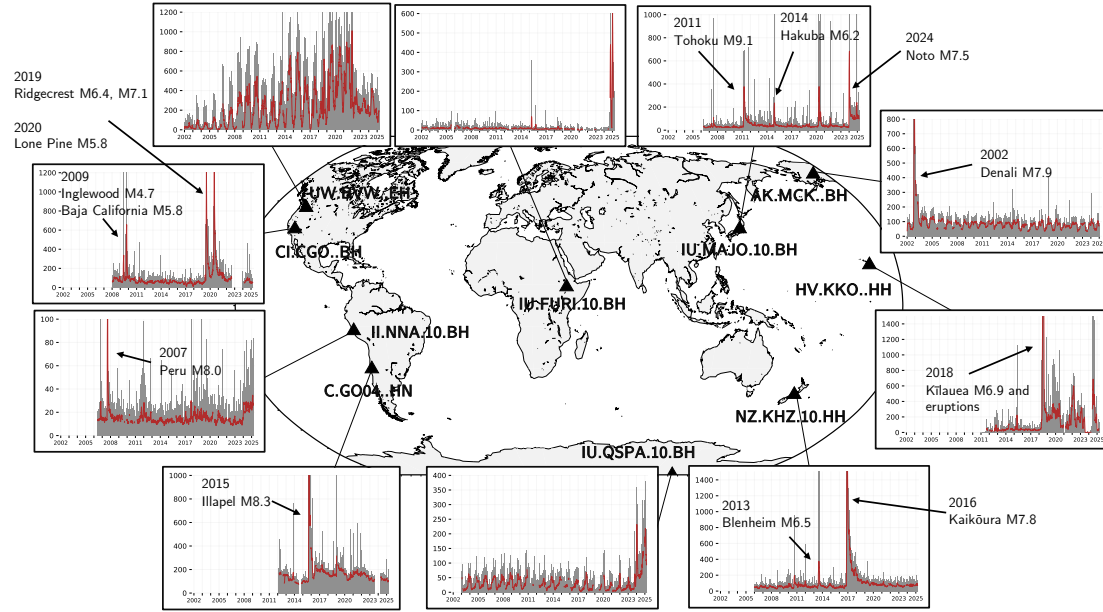


Figure 5.4: Daily picks for selected stations. Stations are indicated by triangles on the central map, annotated with location and channel codes. For each of the ten example stations, the detail plots show a time series of the number of picks per day and a 28-day moving average in red.

Figure 5.5 shows the number of picks and earthquakes after three selected large earthquakes: (a) 2015 M8.3 Illapel earthquake, (b) 2016 M7.8 Kaikōura earthquake, and (c) 2002 M7.9 Denali earthquake. In each case, the daily pick count follows approximately an Omori decay law²⁷⁵. While the Omori decay in the event count can only be observed for a short period in the regional and global reference catalogs used, they are stable over longer durations in the pick count. For the Kaikōura earthquake, the Omori decay in pick counts is stable over more than 1000 days. Notably, the Omori p values, describing the decay rate, differ between picks and events with systematically higher p values for events, indicating a faster decay. For example, for the Illapel earthquake, we estimate $p = 0.87$ for the event counts, a typical value, yet $p = 0.49$ for the pick counts, a surprisingly low value. We suggest this difference originates from the joint effects of event rate and event magnitude on the pick counts. This highlights the value

of analyzing pick count dynamics to study earthquake statistics. Several time series of picks shown in Figure 5.4 exhibit seasonality, demonstrating the potential effects of the noise floor.

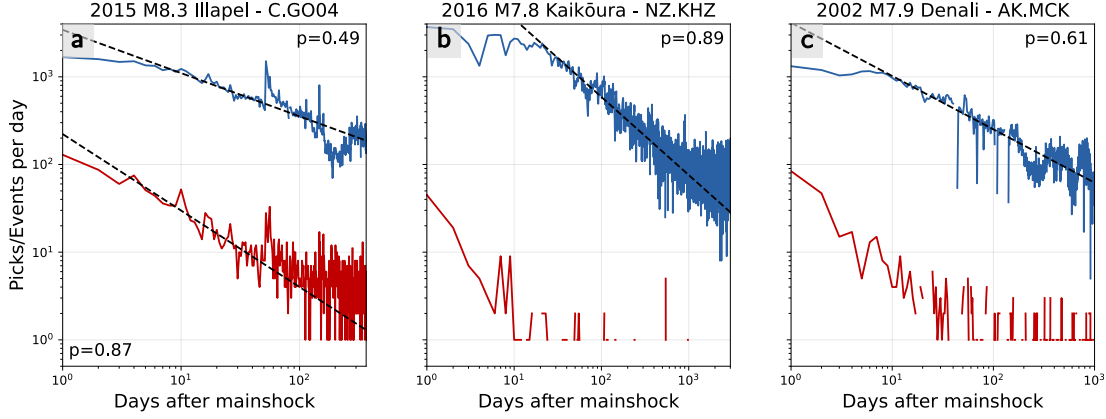


Figure 5.5: Omori-type decays of the number of picks per day (blue) and the number of events per day (red) after the mainshock. We use picks at the reference station provided in the figure title. Event counts for the Illapel earthquake are from the Chilean Seismic Network (CSN) catalog, for the other two examples from the International Seismological Centre (ISC) and United States Geological Survey (USGS) catalog. In each case, we count all events with at most a 1.5-degree difference in latitude and longitude from the reference station.

5.4 Discussion

To estimate the total number of earthquakes we can identify from the 4.3 billion seismic phase picks, we begin by assessing how picks are distributed across different magnitude bins. Based on estimates from McBrearty and Beroza¹⁶¹, we expect an average of approximately 20 P-wave picks per M_1 earthquake, 60 picks per M_2 , and 100 picks per M_3 . Because earthquake frequency decreases with increasing magnitude, for each M_1 event (20 picks), we expect roughly 0.1 M_2 events (6 picks) and 0.01 M_3 events (1 pick). This implies that M_2 s contribute only about 30% as many picks as M_1 s, and M_3 s contribute just 5%. While these estimates are for the dense network in California, the same trends, even though with shifted magnitude values, are observed in deep learning-based catalogs from other regions¹⁸¹. This suggests that the dataset is overwhelmingly composed of picks associated with earthquakes near the regional detectability threshold, e.g., in densely instrumented regions like California M_1 and smaller. Using this assumption, and taking 20 picks per event as a working average, along with an estimated association rate of 25%, based on typical values from literature^{115,161,177,181}, we arrive at a rough estimate of more than

54 million earthquakes that could be associated with the data. We note that many further picks will correspond to actual events that just lack sufficiently many detections to successfully associate them. For context, in the time period of interest, the Southern California Seismic Network (SCSN) catalog contains approximately 450,000 events, while the Advanced National Seismic System (ANSS) Comprehensive Earthquake Catalog (ComCat) includes about 4 million events for the United States and larger global events. Deep learning methods have already demonstrated the ability to increase catalog completeness by up to an order of magnitude in sparsely instrumented regions¹⁹⁹. Given the unprecedented scale of this pick database, we fully expect it to significantly expand the number of earthquakes identified globally, particularly in regions that have historically been under-detected.

We anticipate that the archive of picks will be of use to many researchers in the field of earthquake science. As such, we have made it available for public query using web services and URL builders (see Code and Data Availability). A dedicated EC2 instance receives HTTPS requests and directly returns query results. As an example, one can query the picking results for AK.MCK. station for one month of data using the following Python script. The script uses the pandas package (<https://pandas.pydata.org>) and loads the query results as a data frame. We also offer a binary dump of the entire database for download.

Listing 1: Python query of the picking database

```
import pandas as pd
base_url = "https://dasway.ess.washington.edu/quakescope/service/picks/"
url = f"{base_url}query?tid=AK.MCK.&start_time=2010-01-01&end_time=2010-02-01&limit=1000"
pick = pd.read_csv(url, delimiter="|")
```

Our demonstration is simply a starting point for future and more focused exploration. For instance, we did omit the NCEDC SP? channels that had a name change at some time during the network operation. We also omitted using specialized base models, such as PickBlue, that include hydrophone data and would be more appropriate for picking ocean-bottom seismometer (OBS) data³². There are also opportunities to identify non-traditional seismicity, such as volcanic earthquakes³³⁵ or low-frequency earthquakes associated with slow slip events^{138,182}. The fully open-source and modular design of the workflow ensures reproducibility while allowing flexibility to incorporate different models and pre-

trained weights optimized for different applications. This enables other researchers to easily adopt and extend the cloud-native workflow for their own analyses. Future work might consider using template matching to enhance our detections, as done in California²²⁶.

Moreover, we did not use data from other seismic data providers. For instance, we omitted the non-FDSN seismic networks, the Observatories and Research Facilities for European Seismology (ORFEUS) federated networks, and the AusPass networks. Pulling data from these data centers can be done using their FDSN web services; however, care must be taken not to overload the data service¹⁵¹. Because of the resiliency of the commercial cloud providers to spiked demand in data access, a strategy to improve the stability of the non-cloud-hosted archive is to use cloud storage as a backup in case of service interruption and route users toward the cloud-hosted archive instead of the local seismic network’s data servers.

5.5 Conclusions

In conclusion, our data mining experiment plants the seed for impactful advances in geophysics. Cloud-based and AI-aided picking of P- and S-waves can be used to retrain neural networks and improve the rapid and precise assessment of earthquake hazards, such as earthquake early warning³³². These newly detected potential earthquakes and waveforms may be valuable to build foundational models for seismology that learn fundamental seismic signal patterns from massive waveform libraries^{146,290}, and then be fine-tuned for specific tasks (e.g., picking, polarity, backazimuth, etc.). Running this petabyte-scale workflow on the cloud also provides a testbed for greener computing with tools that can allow researchers to deploy carbon-aware computing jobs²⁹⁵.

Acknowledgment

This work is supported by the Seismic Computational Platform for Empowering Discovery (SCOPED) project under the National Science Foundation (award numbers OAC-2103701 (UW), OAC-2103494 (UT)). YN and MD are also partially supported by the EarthScope Consortium through a Pass-Through Entity (PTE) Federal award no 2310069. The computing resources presented in this paper were obtained using CloudBank¹⁹⁵, which is supported by the National Science Foundation (award number CNS-1925001). Data were accessed from the NSF SAGE data repository operated by EarthScope Consortium

(award number 1724509). JM has been funded by the European Union under the grant agreement n°101104996 (DECODE). The Cascadia Region Earthquake Science Center (CRESCENT), funded by National Science Foundation Cooperative Agreement #2225286, partially supported this project through funding to AMT and LB. AMT was also supported by the National Science Foundation (award number 1848302). We thank Dr. Piero Poli and an anonymous reviewer for their comments and suggestions that improve this manuscript.

Code and Data Availability

Data from SCEDC and NCEDC is publicly available on AWS S3, while data hosted at EarthScope could be accessed through the EarthScope FDSN web service. The DOIs of FDSN seismic networks used in this study are listed at <https://dasway.ess.washington.edu/quakescope/doi.csv> and added as supplementary materials. All data products are hosted at <https://dasway.ess.washington.edu/quakescope>. URL builders are available for database queries. The database snapshots are also dumped as BSON (binary JSON) files. Our code base is hosted on GitHub at <https://github.com/SeisSCOPED/QuakeScope>.

6

Reconstruction of Earthquake Ground Motion Fields with Compressive Learning and Generative Refinement

6.1 Introduction

Accurately predicting earthquake ground motion is critically important in seismology, forming the basis of earthquake early warning (EEW) systems that enable rapid event response^{10,11,107}. The characteristics

of earthquake wavefields are governed by three primary factors: 1) the excitation of seismic waves at the source, 2) their propagation through a 3D elastic medium, and 3) their propagation in nonlinear media in the near-surface environment (e.g., site response). Earthquake sources generate diverse wavefield characteristics depending on their size, geometry, and sense of motions that ultimately govern their rupture dynamics, giving rise to a wavefield enriched in the period of the earthquake duration. The heterogeneity of the Earth model controls 3D wave propagation effects, including geometrical spreading and scattering, which determine wave amplitude decay and prolong shaking through the coda. Finally, near-surface site conditions introduce variability in ground motion intensity^{31,211}. Capturing the combined influence of all these factors for rapid wavefield reconstruction and ground motion prediction remains a challenging problem.

Rapid shaking assessment relies on ground-motion intensity measures, most commonly predicted through ground-motion prediction equations (GMPEs)¹, which regress observations from past events against a set of scalar source and site attributes (e.g., magnitude, source-to-site distance, site conditions, Boore et al.³⁰). GMPE predictions can be rapidly available after an earthquake is detected, but they exhibit two fundamental limitations. First, GMPEs rely on accurate source parameters, especially magnitude (e.g., Wu and Kanamori³⁰⁸), and their initial uncertainties or errors propagate directly into uncertainties of predicted ground motions. Second, because GMPEs parameterize ground motion through scalar features, their predictions are inherently spatially smooth and cannot resolve the localized variability driven by wavepath complexity, scattering, and site amplification, leaving unexplained effects as a “ σ ” error.

Physics-based earthquake wavefield simulation offers an alternative way for ground motion prediction by numerically solving the wave equation. Open-source software (e.g., SPECFEM^{121,204}, SeisSol¹¹⁷, SW4²⁰⁶) has made such simulations accessible, but routine operational usage of such simulations faces two fundamental limitations. First, the computational cost is prohibitively high for routine prediction that necessitates rapid response, particularly for the fine meshes needed to resolve shallow sediments and short-wavelength site effects^{256,264}. Second, the accuracy of any simulation is bounded by the fidelity of the Earth model used, while such input is poorly constrained by limited observations and is subject to inherent uncertainties. Therefore, physics-based wavefield simulations may serve as a scenario-planning

¹We use GMPE rather than the common acronym GMM to avoid confusion with Gaussian Mixture Models used later.

tool as is done by CyberShake⁸⁹, rather than supporting a real-time prediction framework.

Data-driven wavefield reconstruction algorithms offer a third path by inferring the continuous spatial wavefield at current times (i.e, nowcasting) directly from current and past sparse sensor observations. In contrast to GMPEs, these approaches do not require explicit parameterization of source, path, and site effects, enabling them to capture spatially complex wavefield patterns. Moreover, data-driven models require minimal inference cost, enabling near-real-time deployment without repeated numerical solves while remaining adaptive to dynamic sensor configurations. In seismology, studies have reconstructed long-period teleseismic body and surface waves^{40,176} and regional earthquake wavefields²¹². These new methods have enhanced earthquake ground-motion characterization and can be readily applied in EEW applications when framed for forecasting or future prediction (e.g., Ben-Zion et al.²⁵, Clements et al.⁴⁷, Feng et al.⁷⁵, Münchmeyer et al.¹⁷⁹). While these existing applications have focused primarily on waveform fidelity at sensor locations, whether reconstructed wavefields match the ground motion intensity distributions (e.g., the PGA and PGV maps) has not been systematically evaluated.

In this study, we develop and evaluate a multi-stage machine learning framework (EQ-SHRED and EQ-FLOW) that reconstructs earthquake ground motion fields from a sparse virtual sensor network and simultaneously produces spatially resolved PGA and PGV maps together with probabilistic hypocenter estimates as a nowcasting framework. Building on the SHRED architecture³⁰³ and a compressive learning approach, EQ-SHRED pairs a GRU-based recurrent encoder with MLP decoders (Stage 1A) and independent Gaussian Mixture Model source branches (Stage 1B). A conditional flow matching generative model (EQ-FLOW, Stage 2) then refines the low-rank Stage 1A output to recover short-wavelength spatial complexity. We train and evaluate the framework on an augmented version of the HETerogeneous Materials and Elastic Waves with Source variability in 3D dataset (HEMEW^S-3D¹³⁴), which provides ground-truth wavefields at a spatial resolution sufficient to resolve the fine-scale features that intensity measures depend on. Our results demonstrate that the framework accurately recovers the large-scale spatial structure of multi-task ground motion fields and intensity maps from only $\sim 0.3\%$ of the surface grid observations, while the generative refinement stage restores short-wavelength spatial complexity and the source branches localize the hypocenter within seconds of the first arrival. The remainder of this paper is organized as follows: Section 6.2 describes the synthetic dataset and sensor

configuration; Section 6.3 presents the machine learning framework; Section 6.4 reports reconstruction results; and Section 6.5 discusses implications for earthquake early warning and directions for future work.

6.2 Data

Training a supervised machine learning model for wavefield reconstruction requires paired examples of sparse sensor observations and the corresponding dense ground-truth wavefield over a large number of source configurations. Real seismic stations are installed only at fixed station locations, so a spatially complete wavefield is never observed in practice. Through synthetic simulations with a known Earth model and source, the full wavefield can be carried out at any desired spatial and temporal resolution, providing the ground-truth data needed for training a data-driven wavefield reconstruction model.

We generate synthetic ground-motion wavefields in a 3D Cartesian domain with flat topography by augmenting the HEMEW^S-3D dataset¹³⁴. The elastic wave equation is solved using an open-source Spectral Element Method (SEM) implementation²⁶⁸. In the original HEMEW^S-3D dataset, the Earth velocity model is structured as random homogeneous layered media superposed with random perturbations. Each simulation is performed using a unique combination of velocity model, source location, and focal mechanism. To simulate a regional earthquake monitoring scenario with fixed geological conditions and Earth structure, we use the same velocity (V_p and V_s) and density model (ρ) for all simulations (sample index 100047). No attenuation is applied for both P- and S-waves at this stage, which will lead to overly large peak ground motion amplitudes. The V_p/V_s ratio is fixed at 1.7, while the density ρ is computed deterministically from the P-wave velocity¹⁶⁹.

$$\rho = 1.6612V_p - 0.4721V_p^2 + 0.0671V_p^3 - 0.0043V_p^4 + 0.000106V_p^5 \quad (4)$$

We randomly select 4,722 earthquake sources parameterized by the hypocenter location $\mathbf{x}_{src} = (x_s, y_s, z_s)$ and the focal mechanism (strike, dip, and rake angles) from Lehmann et al.¹³⁴. We also use the same earthquake source time function, earthquake moment, and meshing configuration. The same absorbing boundary conditions with perfectly matched layers apply to all edges except the top surface. The source amplitude corresponds to a seismic moment $M_0 = 2.47 \times 10^{16} N \cdot m$, equivalent to a

moment magnitude of $M_w \approx 4.86$. Virtual sensors are arranged on top of the simulation domain ($D_x = D_y = D_z = 9.6$ km) to record three-component (East-West, North-South, and Vertical) velocity field at 50 Hz for up to 6 seconds after the simulation origin time. In contrast to the HEMEW^S-3D dataset, the spatial resolution of the earthquake wavefield is improved by a factor of 3 (100 meters) to capture continuous short-wavelength wavefield features. To mitigate high-frequency numerical instabilities, we apply a 10 Hz low-pass Butterworth filter to the broadband velocity wavefield ($\mathbf{x}_{velocity}$, for simplicity, $\mathbf{x}_{vel}$). Each simulation requires approximately 80 core-hours, for a total computational cost of $\sim 378,000$ core-hours across all 4,722 simulations.

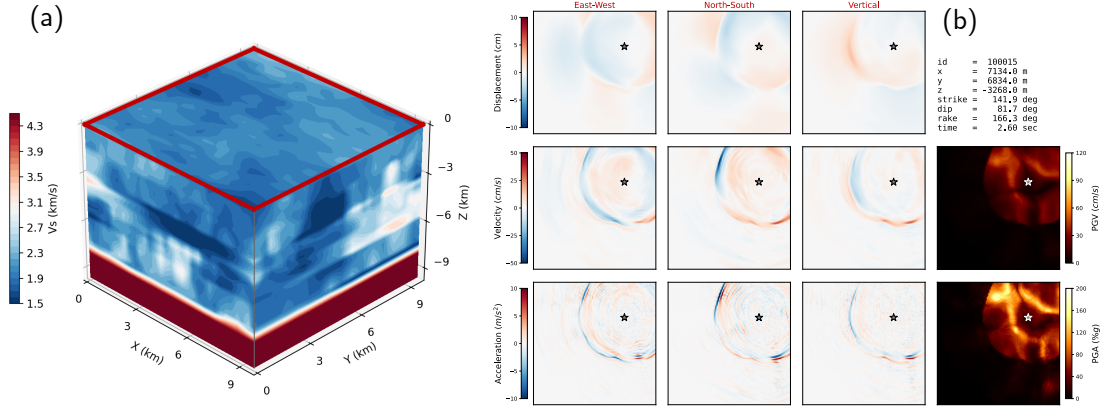


Figure 6.1: (a) Three-dimensional shear wave velocity (V_s) structure of the simulation domain ($D_x = D_y = D_z = 9.6$ km). The red outline marks the surface observation area where virtual sensors record ground motion. (b) Example surface wavefield snapshots from a single simulation (sample index 100015). The nine panels show the three-component (East-West, North-South, and Vertical) displacement (cm), velocity (cm/s), and acceleration (m/s^2) field snapshot, with the projected source location marked by the gray star. The right panels show the corresponding Peak Ground Velocity (PGV, cm/s) and Peak Ground Acceleration (PGA, %g) maps at the same time stamp.

Based on the simulated velocity wavefield, we derive the corresponding wavefield metrics. The three-component displacement field ($\mathbf{x}_{displacement}$, for simplicity, $\mathbf{x}_{dis}$) is calculated by integrating the velocity field over time for each channel component. Similarly, the three-component acceleration field ($\mathbf{x}_{acceleration}$, for simplicity, $\mathbf{x}_{acc}$) is derived by differentiating the velocity field over time for each channel component. The displacement, velocity, and acceleration wavefields for all simulations are

represented as 5D tensors:

$$\mathbf{x}_{dis} \in \mathbb{R}^{N_{sim} \times N_c \times N_x \times N_y \times N_t} \quad (5)$$

$$\mathbf{x}_{vel} \in \mathbb{R}^{N_{sim} \times N_c \times N_x \times N_y \times N_t} \quad (6)$$

$$\mathbf{x}_{acc} \in \mathbb{R}^{N_{sim} \times N_c \times N_x \times N_y \times N_t} \quad (7)$$

where N_{sim} is the number of simulations, $N_c = 3$ is the number of components for velocity, displacement, and acceleration fields, $N_x = N_y = 94$ is the number of spatial grid points of the surface observation, and $N_t = 300$ is the number of time steps. To characterize the maximum intensity of ground shaking, we calculate the horizontal time-dependent Peak Ground Velocity (PGV) and Peak Ground Acceleration (PGA) by taking the geometric mean of both the East-West ($\mathbf{x}_{vel}^{EW}$ and $\mathbf{x}_{acc}^{EW}$) and the North-South ($\mathbf{x}_{vel}^{NS}$ and $\mathbf{x}_{acc}^{NS}$) components of the velocity and acceleration fields, respectively. This is chosen as a rotation-independent scalar measure of ground motion intensity, consistent with standard practice in ground motion characterization²⁹.

$$\mathbf{x}_{pgv}(t) = \max_{t' \leq t} \sqrt{|\mathbf{x}_{vel}^{EW}(t')| \cdot |\mathbf{x}_{vel}^{NS}(t')|}, \quad \mathbf{x}_{pgv} \in \mathbb{R}^{N_{sim} \times N_c \times N_x \times N_y \times N_t} \quad (8)$$

$$\mathbf{x}_{pga}(t) = \max_{t' \leq t} \sqrt{|\mathbf{x}_{acc}^{EW}(t')| \cdot |\mathbf{x}_{acc}^{NS}(t')|}, \quad \mathbf{x}_{pga} \in \mathbb{R}^{N_{sim} \times N_c \times N_x \times N_y \times N_t}, \quad (9)$$

where $N_c = 1$ corresponds to a single-component scalar field.

6.3 Methods

The EQ-SHRED framework consists of two sequential stages (Figure 6.2). In Stage 1, sparse three-component seismic velocity recordings from a virtual station array are processed by a GRU-based recurrent encoder that maps the sensor trajectories to a compact latent representation. Two decoding branches operate on this shared latent space: (1A) the Multi-Layer Perceptron (MLP) decoders reconstruct the full surface wavefield (displacement, velocity, PGV, and PGA) using a compressive learning strategy (Section 6.3.1), and (1B) three independent Gaussian Mixture Models estimate probability distributions of the hypocenter location (Section 6.3.2). In Stage 2, the low-resolution wavefield reconstructions from Stage 1A are fed as conditioning inputs to a flow matching model, which generates

high-resolution refinements of all physical fields (Section 6.3.3). The two stages are trained independently, and the full training and evaluation setup is described in Section 6.3.4.

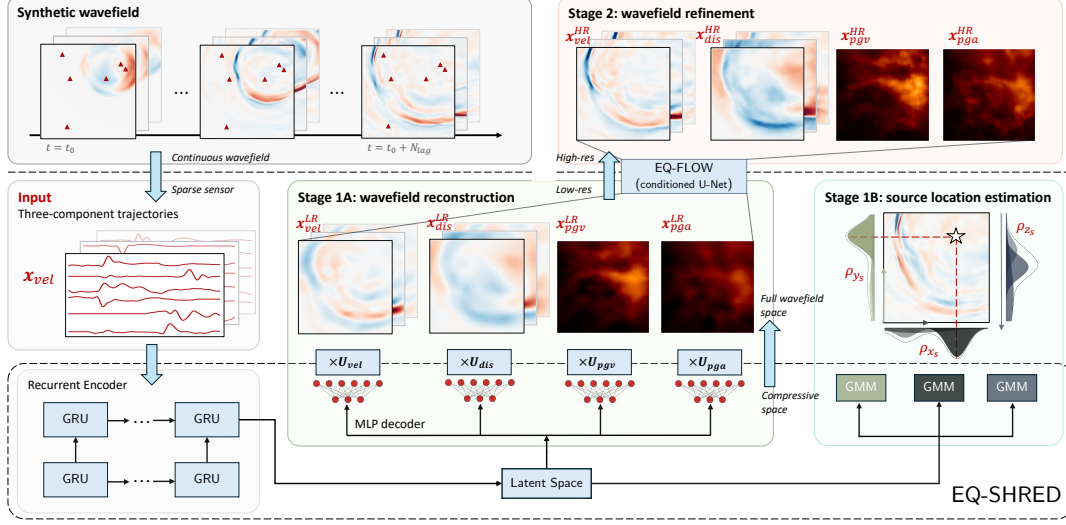


Figure 6.2: Architecture of the multi-stage wavefield reconstruction framework (EQ-SHRED and EQ-FLOW). Three-component velocity trajectories recorded at sparse sensors over a lag window $[t_0, t_0 + N_{lag}]$ are encoded by a stacked GRU-based recurrent encoder into a shared latent space. In Stage 1A (wavefield reconstruction), MLP decoder branches map the latent space to the compressive space, which are then expanded to the full spatial wavefield through their respective left singular mode matrices as the low-resolution reconstructions. In Stage 1B (source location estimation), the Gaussian Mixture Model (GMM) decode the latent space into probabilistic distributions of the hypocenter locations ($\rho_{x_s}(\cdot)$, $\rho_{y_s}(\cdot)$, $\rho_{z_s}(\cdot)$). In Stage 2 (wavefield refinement), a flow matching conditioned U-Net²²⁴ takes the low-resolution wavefield reconstructions as conditioning inputs and generates high-resolution refinements.

6.3.1 Wavefield Reconstruction using Shallow Recurrent Decoder

We extend and adapt the SHallow Recurrent Decoder model (SHRED³⁰³) for for this multi-task wavefield reconstruction. The general form of the SHRED model consists of a recurrent encoder that maps the sparse sensor trajectories into a latent space, and a decoder that reconstructs the full spatial-temporal wavefield. Such a design reflects the underlying key concepts of the separation of variables and the Takens' embedding theorem. Advancements have been made for a wide range of scientific applications, including fluid dynamics³⁰³, plasma physics^{130,216}, and remote sensing³³¹. In seismology, SHRED has been successfully used for seafloor fiber-optic sensing wavefield reconstruction and separation, advancing earthquake monitoring in an edge-computing environment¹⁹¹. While the

original encoder-decoder setting in SHRED is based on the Long-Short-Term Memory (LSTM¹⁰⁶) and the Multi-Layer Perceptron, the general form of a recurrent encoder and a spatial-only decoder allows for the use of the Gated Recurrent Unit (GRU⁴⁴) and the Transformer²⁷⁸ for better temporal encoding³¹⁷. Here, we construct our model (EQ-SHRED hereafter) using a stacked GRU recurrent encoder (two layers, 600 hidden units) and MLP decoders for the multi-task wavefield reconstruction. The ReLU activation function is used for all hidden layers of the wavefield decoders⁴.

We randomly select $N_{sta} = 33$ sensors to form a virtual sparse array of seismic stations, occupying $\frac{N_{sta}}{N_x \times N_y} \approx 0.3\%$ of the total surface grid points. Three-component velocity trajectories $\mathbf{x}_{vel}$ sampled from t_0 to $t_0 + N_{lag}$ serve as the input to the recurrent encoder, where $N_{lag} = 100$ is the number of temporal lag steps, which corresponds to 2 seconds. We choose velocity as the sole physical field for the input, as it is a common type of ground motion measurement for the monitoring of regional small-to-intermediate earthquakes. While the strong-motion accelerometers are increasingly deployed for earthquake monitoring, the acceleration field itself features more high-frequency content and short-wavelength features, and we find that these characteristics made it more challenging to reconstruct from a limited number of sensors.

While the original SHRED is designed to map the latent space directly to the full spatial-temporal space of the wavefield, several studies have shown the effectiveness of a compressive learning strategy^{216,219}, where the Proper Orthogonal Decomposition (POD) is applied to reduce the dimensionality of the wavefield data, and the SHRED reconstruction is performed in the compressive space. While POD generally does not require specific orthogonal basis functions, we perform the singular value decomposition (SVD) to reduce the dimensionality of the wavefield data and improve the interpretability of the POD modes, as in^{212,216,265}. Instead of the standard SVD for full decomposition, we use the randomized SVD algorithm where the decomposition is truncated and a given number of leading modes are returned at high accuracy and efficiency¹⁶⁰. More specifically, for each component of the displacement, velocity, PGV, and PGA fields as a 4D tensor $\mathbf{x} \in \mathbb{R}^{N_{sim} \times N_x \times N_y \times N_t} \xrightarrow{\text{reshape}} \mathbb{R}^{N_x \cdot N_y \times N_{sim} \cdot N_t}$, the spatial and temporal dimensions can be effectively separated through the randomized SVD algorithm. The final

reshaped form of decomposition is given as

$$\mathbf{x} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \quad (10)$$

$$\mathbf{U} \in \mathbb{R}^{N_x \cdot N_y \times r} \xrightarrow{\text{reshape}} \mathbb{R}^{N_x \times N_y \times r}, \quad \mathbf{\Sigma} \in \mathbb{R}^{r \times r}, \quad \mathbf{V}^T \in \mathbb{R}^{r \times N_{sim} \cdot N_t} \xrightarrow{\text{reshape}} \mathbb{R}^{r \times N_{sim} \times N_t}, \quad (11)$$

where $\mathbf{U}$ and $\mathbf{V}$ are the left and right singular vectors, and $\mathbf{\Sigma}$ is the diagonal matrix of singular values. As the left singular vector $\mathbf{U}$ is not time- or simulation-dependent, it effectively forms a POD basis for the spatial modes of the wavefield. Meanwhile, the right singular vector $\mathbf{V}^T$ is not spatial-dependent, and it captures the temporal evolution of the POD basis for each simulation. We multiply the singular values $\mathbf{\Sigma}$ with the right singular vectors $\mathbf{V}^T$ to form the compressive state $\mathbf{C}$:

$$\mathbf{C} = \mathbf{\Sigma}\mathbf{V}^T, \quad \mathbf{C} \in \mathbb{R}^{r \times N_{sim} \times N_t} \quad (12)$$

The dimensionality of the spatial-temporal wavefield $\mathbf{x}$ is reduced by retaining only the leading singular values and their corresponding singular vectors and compressive states ($\mathbf{U}$ and $\mathbf{C}$), considering different singular value decay rates among different fields. Here, we calculate $r = 1000$ modes for velocity, 200 modes for displacement, PGA, and PGV fields. Moreover, we further truncate down to the leading $r' = 100$ modes of each wavefield type and component as the target of the EQ-SHRED wavefield branches. Figure 6.3 shows the singular value decay rates for different wavefields and 10 dominant spatial modes.

As mentioned above, the design of the EQ-SHRED model reflects the underlying key concepts of the separation of variables in spatial-temporal wavefields. Here, the use of the SVD-based POD still honors such a design principle, while only replacing the high-dimensional spatial state ($\hat{\mathbf{x}} \in \mathbb{R}^{N_x \times N_y}$ at $t = t_0 + N_{lag}$) with a low-dimensional intermediate compressive state ($\hat{\mathbf{c}} \in \mathbb{R}^{r'}$ at $t = t_0 + N_{lag}$). Once the EQ-SHRED model predicts the compressive state $\hat{\mathbf{c}}$, the full spatial wavefield $\hat{\mathbf{x}}$ at $t = t_0 + N_{lag}$ can be rapidly expanded by multiplying the left/spatial singular modes of that component and wavefield type.

$$\hat{\mathbf{x}} = \mathbf{U}'\hat{\mathbf{c}}, \quad (13)$$

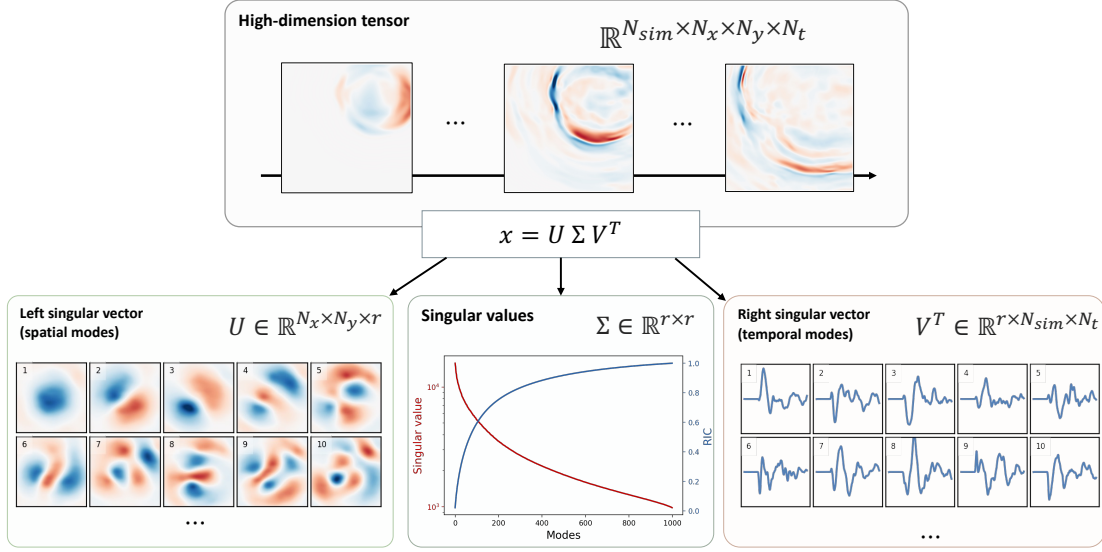


Figure 6.3: Example of the randomized Singular Value Decomposition (SVD) applied to the velocity field (vertical component). The high-dimensional tensor $\mathbf{x}_{vel}^Z \in \mathbb{R}^{N_{sim} \times N_x \times N_y \times N_t}$ is decomposed into the left singular vectors $\mathbf{U} \in \mathbb{R}^{N_x \times N_y \times r}$ (spatial modes), the diagonal matrix of singular values $\mathbf{\Sigma} \in \mathbb{R}^{r \times r}$, and the right singular vectors $\mathbf{V}^T \in \mathbb{R}^{r \times N_{sim} \times N_t}$ (temporal modes). The center panel shows the singular value decay (red) and the relative information content (RIC; blue) as a function of the number of retained modes. The left and right panels show the 10 leading spatial and temporal modes, respectively.

where $\mathbf{U}' \in \mathbb{R}^{N_x \times N_y \times r'}$ is the truncated left singular mode matrix.

There are two key benefits of reconstructing the wavefield within the compressive space. First, the low-rank representation of the earthquake wavefield, as represented by the discrete spatial singular modes, provides a form of physical prior that simplifies what the decoder needs to learn from scratch. The decoder does not need to learn the full mapping from the latent space to the high-dimensional spatially continuous wavefield; instead, it only needs to learn the mapping to the singular values, while the full space can be further reconstructed using the spatial singular modes. Second, simpler model structures with fewer parameters can be used for each decoder branch, allowing more efficient training and inference.

6.3.2 Earthquake Source Location Estimation using Gaussian Mixture Model

Aside from the decoder branches that reconstruct the multi-task ground motion fields, we also implement separate source branches that independently predict the hypocenter location (x_s , y_s , and z_s) directly

from the same latent space. Instead of predicting the source location as a deterministic regression task, we design the source branches to predict the probabilistic distribution of the source parameters, allowing for better time-dependent source parameter estimation and uncertainty quantification. Therefore, each source branch consists of an MLP and a Gaussian Mixture Model (GMM) to predict the probabilistic distribution of the targeted parameters as a weighted sum of N_{gmm} Gaussian distributions. Here, we use $N_{gmm} = 2$. Specifically, the MLP has an output shape of $3 \times N_{gmm}$, corresponding to the weighting factors α_i , the means μ_i , and the standard deviations σ_i of the i^{th} Gaussian distributions. The final estimate of the source location is the weighted sum of individual Gaussian distributions:

$$\rho(x_s) = \sum_{i=1}^{N_{gmm}} \alpha_{x,i} \frac{1}{\sqrt{2\pi}\sigma_{x,i}} \exp\left(-\frac{(x_s - \mu_{x,i})^2}{2\sigma_{x,i}^2}\right) \quad (14)$$

$$\rho(y_s) = \sum_{i=1}^{N_{gmm}} \alpha_{y,i} \frac{1}{\sqrt{2\pi}\sigma_{y,i}} \exp\left(-\frac{(y_s - \mu_{y,i})^2}{2\sigma_{y,i}^2}\right) \quad (15)$$

$$\rho(z_s) = \sum_{i=1}^{N_{gmm}} \alpha_{z,i} \frac{1}{\sqrt{2\pi}\sigma_{z,i}} \exp\left(-\frac{(z_s - \mu_{z,i})^2}{2\sigma_{z,i}^2}\right) \quad (16)$$

6.3.3 Wavefield Generative Refinement using Flow Matching

Although EQ-SHRED effectively reconstructs the multi-task wavefields from sparse sensor observations, the SVD-based compressive learning strategy inherently limits the spatial fidelity of the reconstruction to a number of leading singular modes. The truncated low-rank representation smooths out fine-scale spatial features that are not captured by the retained modes, and therefore lacks high-frequency wavefield patterns associated with short-wavelength heterogeneity when projected back to the full wavefield space. To recover or interpolate these missing spatial details, we introduce a generative refinement stage that takes the EQ-SHRED reconstruction $\mathbf{x}_{LR} \in \mathbb{R}^{N_c \times N_x \times N_y}$ as a low-resolution prior, and produces $\mathbf{x}_{HR} \in \mathbb{R}^{N_c \times N_x \times N_y}$ with refined spatial complexity. We adopt flow matching for this task (EQ-FLOW hereafter) in preference to score-based diffusion models²⁵⁰ or adversarial approaches⁸⁸, as its straight interpolation paths yield a simpler and better-conditioned training objective with fewer function evaluations required at inference¹⁴².

Flow matching¹⁴² is a generative modeling framework that learns a continuous normalizing flow by

directly regressing a vector field $\mathbf{v}_\theta$ that transports samples from a simple source distribution p_0 (e.g., standard Gaussian noise) to the target data distribution p_1 along straight conditional paths. Given a data sample $\mathbf{x}_1 \sim p_1$, a noise sample $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and the scalar flow-matching time variable τ ², the conditional flow path is defined as

$$\mathbf{x}_\tau = (1 - \tau) \mathbf{x}_0 + \tau \mathbf{x}_1, \quad \tau \in [0, 1], \quad (17)$$

with the corresponding target vector field $\mathbf{u}_\tau(\mathbf{x}_\tau | \mathbf{x}_1) = \mathbf{x}_1 - \mathbf{x}_0$. Specifically, we use flow matching for the generative wavefield refinement stage, where $\mathbf{x}_1 = \mathbf{x}_{HR}$, $\mathbf{u}_\tau(\mathbf{x}_\tau | \mathbf{x}_{HR}) = \mathbf{x}_{HR} - \mathbf{x}_0$, and the vector field is additionally conditioned on the low-resolution wavefield $\mathbf{x}_{LR}$, i.e., $\mathbf{v}_\theta(\mathbf{x}_\tau, \tau, \mathbf{x}_{LR})$, so that the generated samples are guided toward high-resolution wavefields consistent with $\mathbf{x}_{LR}$. At inference, a refined high-resolution wavefield $\mathbf{x}_{HR}$ is synthesized by integrating the conditional Ordinary Differential Equation (ODE) $d\mathbf{x}_\tau/d\tau = \mathbf{v}_\theta(\mathbf{x}_\tau, \tau, \mathbf{x}_{LR})$ from $\tau = 0$ to $\tau = 1$ starting from a Gaussian noise sample $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ using a numerical ODE solver (Figure 6.4).

We employ a 2D U-Net²²⁴ as the backbone architecture of EQ-FLOW for $\mathbf{v}_\theta$. The conditioning input to the U-Net is the channel-wise concatenation of the noise sample $\mathbf{x}_0 \in \mathbb{R}^{N_c \times N_x \times N_y}$ and the low-resolution wavefield $\mathbf{x}_{LR} \in \mathbb{R}^{N_c \times N_x \times N_y}$, where N_c is the number of components of the wavefield type ($N_c = 3$ for velocity and displacement; $N_c = 1$ for PGV and PGA). The time variable τ is encoded using sinusoidal embeddings and injected into each residual block⁸², following standard practices in diffusion and flow matching architectures^{105,324}. Note that the model operates on individual wavefield snapshots: for each simulation, all N_t time steps are treated as independent training samples, so the full training/testing set consists of $N_{\text{train}} \times N_t$ snapshot pairs $(\mathbf{x}_{LR}, \mathbf{x}_{HR})$. Since different wavefield types exhibit different spatial patterns and distributions, we train separate U-Net models for each wavefield type. More details on the dataset construction are described in Section 6.3.6.

6.3.4 Model Training and Testing

In this section, we report the training and testing of the proposed models: EQ-SHRED in Section 6.3.5 and EQ-FLOW in Section 6.3.6. All experiments are performed with on one NVIDIA RTXPRO 6000

²We use τ to avoid confusion with the time dimension of earthquake wavefields.

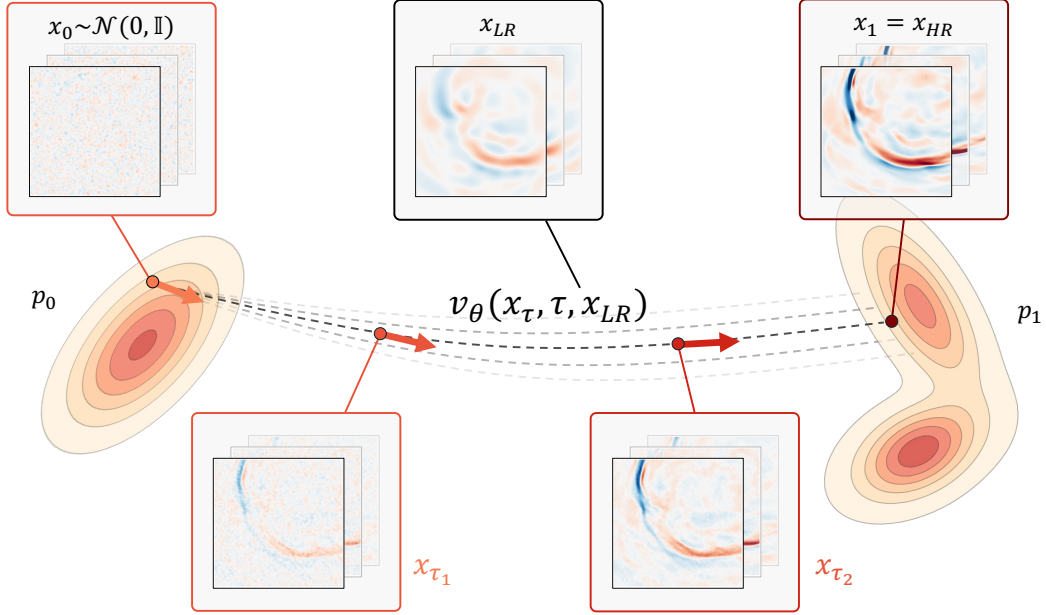


Figure 6.4: Schematic of the EQ-FLOW generative refinement stage based on conditional flow matching. A Gaussian noise sample $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is progressively shifted to the target distribution p_1 of high-resolution wavefields $\mathbf{x}_1 = \mathbf{x}_{HR}$, constrained by the low-resolution wavefield $\mathbf{x}_{LR}$ reconstructed by EQ-SHRED from Stage 1A. Wavefield snapshots at two intermediate time steps ($\mathbf{x}_{\tau_1}, \mathbf{x}_{\tau_2}$) illustrate the gradual emergence of fine-scale wavefield features as the integrating path evolves from noise toward the high-resolution target.

Blackwell 96 GB GPU.

6.3.5 EQ-SHRED

We employ a two-stage sequential training strategy for the EQ-SHRED model (see Figure 6.2) corresponding to Section 6.3.1 and Section 6.3.2, respectively. First, the wavefield reconstruction in Stage 1A allows the optimization of both the recurrent encoder and the wavefield decoder branches. Weights are randomly initialized at the beginning of Stage 1A and updated using the AdamW optimizer with an initial learning rate of 5×10^{-5} and a weight decay of 1×10^{-5} ¹⁴⁸. We perform a 70/30 random splitting of all 4,722 simulations for training and testing, respectively. We perform the model training to minimize the multi-task wavefield branches loss function $\mathcal{L}_{\text{wavefield}}$ as the sum of the velocity, displacement, PGA, and PGV reconstruction loss, $\mathcal{L}_{\text{vel}}$, $\mathcal{L}_{\text{dis}}$, $\mathcal{L}_{\text{pga}}$, and $\mathcal{L}_{\text{pgv}}$, respectively. For each wavefield type, the loss is measured as the mean squared error (MSE) between the predicted $\hat{\mathbf{c}}$ and the ground-truth compressive

states $\mathbf{c}$ summed over all components of each field.

$$\mathcal{L}_{\text{vel}} = \sum_{i=1}^3 \|\mathbf{c}_{\text{vel},i} - \hat{\mathbf{c}}_{\text{vel},i}\|_2^2 \quad (18)$$

$$\mathcal{L}_{\text{dis}} = \sum_{i=1}^3 \|\mathbf{c}_{\text{dis},i} - \hat{\mathbf{c}}_{\text{dis},i}\|_2^2 \quad (19)$$

$$\mathcal{L}_{\text{pga}} = \|\mathbf{c}_{\text{pga}} - \hat{\mathbf{c}}_{\text{pga}}\|_2^2 \quad (20)$$

$$\mathcal{L}_{\text{pgv}} = \|\mathbf{c}_{\text{pgv}} - \hat{\mathbf{c}}_{\text{pgv}}\|_2^2 \quad (21)$$

$$\mathcal{L}_{\text{wavefield}} = \mathcal{L}_{\text{vel}} + \mathcal{L}_{\text{dis}} + \mathcal{L}_{\text{pga}} + \mathcal{L}_{\text{pgv}} \quad (22)$$

Once Stage 1A converges to form a consolidated and structured latent space representation and the corresponding wavefield decoders, we then optimize only the weights of the source branches during Stage 1B, while freezing the weights of the recurrent encoder and the wavefield decoder branches. Using the same AdamW optimizer with an initial learning rate of 1×10^{-5} and a weight decay of 1×10^{-5} ¹⁴⁸, we update weights in the source branches and minimize the source location estimation loss.

$$\mathcal{L}_x = -\log(\epsilon + \rho(x_s)) \quad (23)$$

$$\mathcal{L}_y = -\log(\epsilon + \rho(y_s)) \quad (24)$$

$$\mathcal{L}_z = -\log(\epsilon + \rho(z_s)) \quad (25)$$

$$\mathcal{L}_{\text{source}} = \mathcal{L}_x + \mathcal{L}_y + \mathcal{L}_z \quad (26)$$

where $\epsilon = 1 \times 10^{-6}$ is a small constant to avoid numerical instability, (x_s, y_s, z_s) are the ground-truth source locations. The multi-task wavefield branch losses are still being calculated and monitored, but they are not backpropagated.

We design this two-stage strategy for EQ-SHRED so that the latent space is constructed to map from sparse sensor observations to the multi-task wavefield snapshots. In other words, the latent space is not constrained by static hidden variables, i.e., the earthquake source location, the focal mechanism, and the velocity model. Therefore, the source location estimation is expressed as a decoding task from the structured latent space, rather than directly from the raw sensor trajectories.

In both stages, we use the cosine annealing learning rate scheduler with a total of 150 epochs and a minimum learning rate of 1×10^{-6} ¹⁴⁷. A batch size of 256 simulations is used to leverage the computational efficiency of the GPU hardware. All experiments use a fixed random seed of 1234 to ensure reproducibility. We also employ multiple data augmentation strategies to improve the model generalization: 1) simulations are shuffled in the training dataset; 2) random Gaussian noise with a standard deviation of 0.1 is added to the standardized input sensor trajectories; 3) a random number of sensors (maximum 10) are muted to simulate dynamic sensor network geometry and potential sensor failure; and 4) the trajectory windows are randomly selected from the full 6-second time series to enable temporal generalization. The training and testing losses are shown in Figure E.1. All experiments use a fixed random seed of 1234 to ensure reproducibility. The training and testing of each stage take 8 hours to complete.

6.3.6 EQ-FLOW

The EQ-FLOW model is trained independently from the EQ-SHRED model. The training and the testing datasets are constructed by pairing low-resolution wavefields $\mathbf{x}_{LR}$ with the corresponding ground-truth wavefield snapshots $\mathbf{x}_{HR}$ from the simulation database. For each sample, $\mathbf{x}_{LR}$ is obtained by truncating the ground-truth compressive state down to only $r' = 100$ modes as is done in Stage 1A, and then projecting it back to the full wavefield space using the left singular modes, i.e., $\mathbf{x}_{LR} = \mathbf{U}'\hat{\mathbf{c}}$, where $\hat{\mathbf{c}}$ is the truncated compressive state. To improve the model generalization and robustness, we add random perturbations (up to $\pm 30\%$) to the compressive space $\hat{\mathbf{c}}$ before performing the projection. The $\mathbf{x}_{HR}$ counterpart is the full-resolution simulation snapshot at the same time step without additional processing.

Given the wavefield pairs $(\mathbf{x}_{LR}, \mathbf{x}_{HR})$, the noise sample $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and a flow time $\tau \sim \mathcal{U}[0, 1]$, the interpolated sample $\mathbf{x}_\tau$ is constructed from Equation 17. The U-Net model for each wavefield type is then trained to minimize the conditional flow matching loss

$$\mathcal{L}_{\text{FM}} = \|\mathbf{v}_\theta(\mathbf{x}_\tau, \tau, \mathbf{x}_{LR}) - (\mathbf{x}_{HR} - \mathbf{x}_0)\|_2^2, \quad (27)$$

which regresses the vector field directly against the straight-path target $\mathbf{x}_{HR} - \mathbf{x}_0$, providing more

stable gradients compared to score-based diffusion models^{105,250}. We use the same AdamW optimizer and cosine annealing scheduler as for EQ-SHRED^{147,148} for 300 epochs. At inference, the refined high-resolution wavefield is synthesized by integrating the learned ODE using the explicit Runge-Kutta method of order 5(4) (RK45)⁶³, starting from a Gaussian noise sample $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and conditioned by the EQ-SHRED reconstruction as $\mathbf{x}_{LR}$. The training and testing losses are shown in Figure E.2. For each wavefield type, the training and testing take 4 hours to complete.

6.4 Results

6.4.1 Wavefield Reconstruction

We demonstrate the EQ-SHRED and EQ-FLOW framework on the held-out test set by examining the reconstructed multi-task wavefields for a representative example event (sample index 100044). Figures 6.5–6.7 compare the ground-truth simulation, the Stage 1A (EQ-SHRED) reconstruction (projected back to the full wavefield space), and the Stage 2 (EQ-FLOW) refinement for the three-component displacement, velocity, PGV, and PGA fields. For each figure, the left panel (a) shows the full spatial map of the wavefield at a representative time step; the right two panels show the corresponding compressive state amplitudes by inversely projecting the refined wavefield $\hat{\mathbf{x}}$ back to the compressive space $\hat{\mathbf{c}}$, given that the left singular mode matrix $\mathbf{U}$ is orthonormal. We plot the compressive state separately in panel (b) for the leading $[0, r']$ modes reconstructed by Stage 1A, and in panel (c) for the higher-order modes $[r' + 1, r]$ refined in Stage 2. Moreover, we include panel (d), which shows the vector wavenumber power spectral density (PSD) as a function of wavenumber k (km^{-1}). The PSD is computed from the 2D spatial wavefield by taking the two-dimensional Fast Fourier Transform (2D FFT) and radially averaging the vector wavenumber magnitude $k = \sqrt{k_x^2 + k_y^2}$. This omnidirectional spectrum provides a complementary spectral characterization of the spatial resolution achieved by each reconstruction stage.

$$\hat{\mathbf{c}} = \mathbf{U}^T \hat{\mathbf{x}} \quad (28)$$

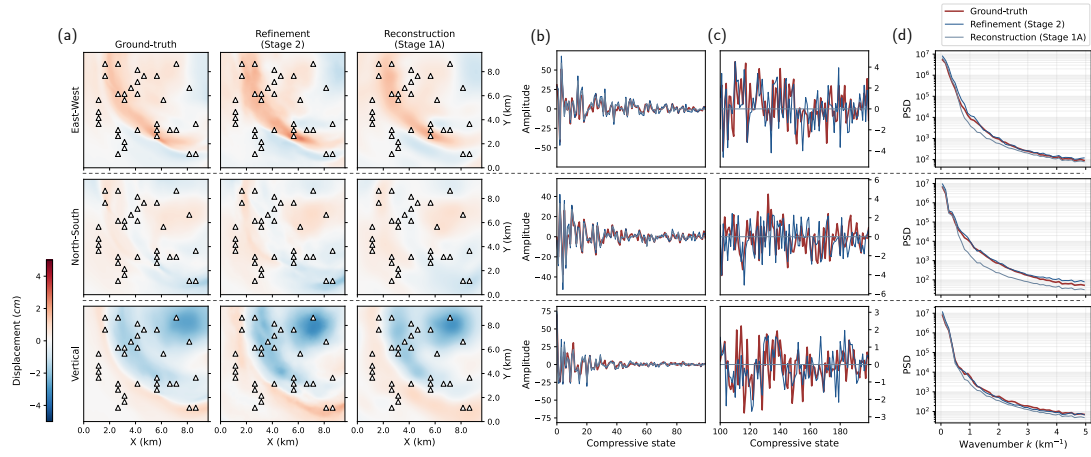


Figure 6.5: Displacement wavefield reconstruction snapshot for a representative test event (sample index 100044). (a) Spatial maps of the displacement field (cm) for three components (East-West, North-South, Vertical) and for ground-truth simulation, Stage 2 refinement, and Stage 1A reconstruction, respectively. White triangles mark the 33 virtual sensor locations that are used as input. (b) Compressive state amplitudes for modes 1–100 per component, comparing ground-truth (red), Stage 1A reconstruction (light blue), and Stage 2 refinement (dark blue). Same comparison for (c) modes 101–200, and (d) vector wavenumber power spectral density (PSD) as a function of wavenumber k (km^{-1}).

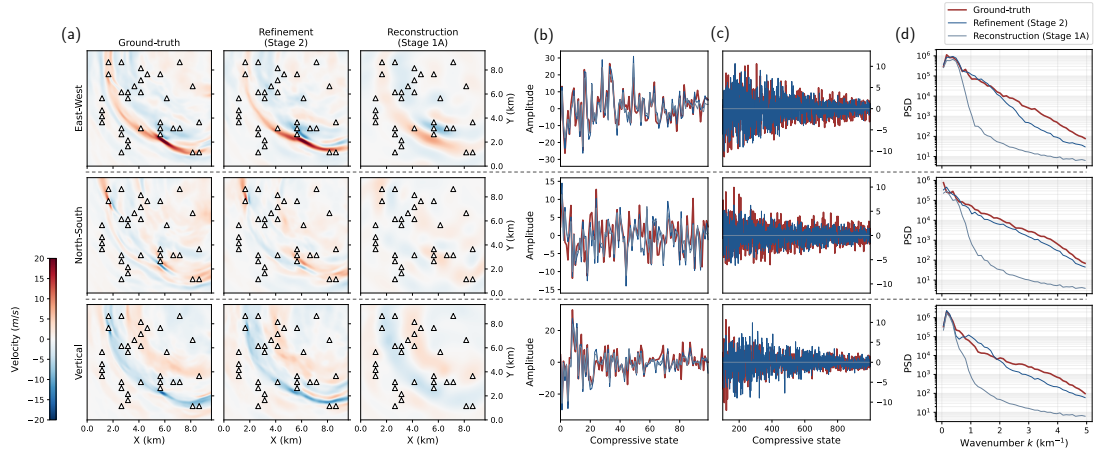


Figure 6.6: Velocity wavefield reconstruction snapshot for the same representative test event as Figure 6.5. (a) Spatial maps of the velocity field (m/s) for three components (East-West, North-South, Vertical) and for ground-truth, Stage 2 refinement, and Stage 1A reconstruction, respectively. White triangles mark the 33 virtual sensor locations that are used as input. (b) Compressive state amplitudes for modes 1–100 per component, comparing ground-truth (red), Stage 1A reconstruction (light blue), and Stage 2 refinement (dark blue). Same comparison for (c) modes 101–1000, and (d) vector wavenumber power spectral density (PSD) as a function of wavenumber k (km^{-1}).

The Stage 1A reconstruction effectively recovers the dominant spatial structure of all wavefield types

using only $\approx 0.3\%$ of all measurements as input. For the displacement field (Figure 6.5), the large-scale radiation pattern and polarity structure of all three components (East-West, North-South, Vertical) are accurately reproduced, and the predicted compressive states closely track the ground-truth for the leading modes. Similarly, for the velocity field (Figure 6.6), the EQ-SHRED reconstruction captures the sharp wavefront geometry. The near-source amplitude patterns in PGV and PGA (Figure 6.7) are also well recovered in Stage 1A. By design, Stage 1A only targets the leading truncated modes and does not predict the higher-order compressive states corresponding to the short-wavelength features. This truncation is reflected in the spatially smooth appearance of the Stage 1A output relative to the ground-truth.

The Stage 2 EQ-FLOW refinement substantially improves the spatial resolution of all four wavefield types. For displacement and velocity (Figures 6.5 and 6.6), the refined snapshots recover short-wavelength features that are not present in the Stage 1A output, and the compressive state representations show that EQ-FLOW partially restores energy in the higher-order modes. This is more evident in the PGA and PGV fields (Figure 6.7), where the ground-truth exhibits localized near-source intensity peaks and finer spatial patterns. These examples demonstrate that our multi-stage framework effectively decouples the recovery of large-scale wavefield structure (Stage 1A, EQ-SHRED) from the generation of fine-scale spatial complexity (Stage 2, EQ-FLOW), with each stage contributing complementary information to the final reconstruction.

It can also be seen from panel (c) that Stage 1A reconstruction produces near-zero amplitudes for those $[r', r]$ high-order modes. This is consistent with the design of the SVD-based compressive learning strategy, where only the leading r' modes are predicted, and only Stage 2 is responsible for recovering the truncated modes. Because EQ-FLOW is a generative algorithm that matches the data distribution in the full wavefield space, the inferred $[r', r]$ modes are not directly conditioned on the input data (velocity trajectories). Consequently, the Stage 2 predictions for these high-order modes exhibit variability and deviate from the ground-truth for individual inferences, even when the overall spatial structure is refined. In addition, the generative nature of EQ-FLOW allows ensemble prediction by performing multiple independent inferences with the same Stage 1A input $\mathbf{x}_{LR}$ and different noise samples $\mathbf{x}_0$. Such ensembles will provide more robust site-level multi-task ground motion predictions

and uncertainty quantification based on the same low-resolution reconstruction constraints.

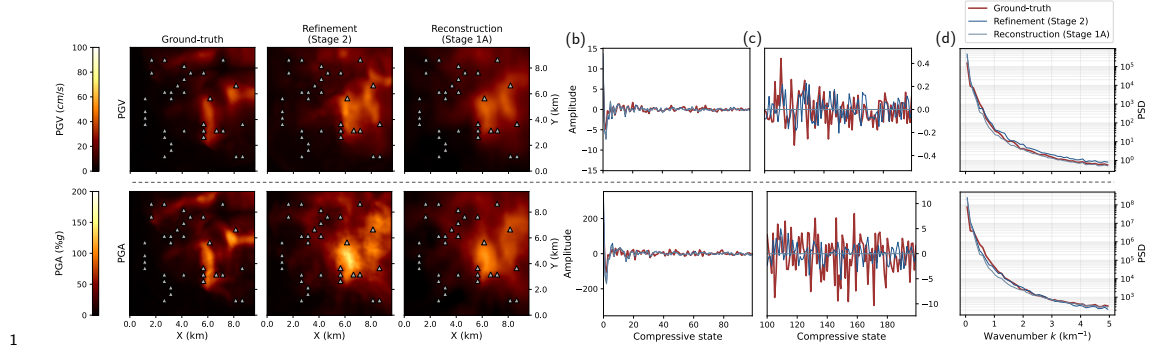


Figure 6.7: Peak Ground Velocity (PGV) and Peak Ground Acceleration (PGA) reconstruction snapshot for the same representative test event as Figure 6.5. (a) Spatial maps of the PGV field (cm/s) and PGA field (%g) showing ground-truth, Stage 2 refinement, and Stage 1A reconstruction, respectively. Gray triangles mark the 33 virtual sensor locations that are used as input. (b) Compressive state amplitudes for modes 1–100, comparing ground-truth (red), Stage 1A reconstruction (light blue), and Stage 2 refinement (dark blue). Same comparison for (c) modes 101–200, and (d) vector wavenumber power spectral density (PSD) as a function of wavenumber k (km^{-1}).

6.4.2 Source Location Estimation

We evaluate the time-dependent hypocenter location estimation performance of the EQ-SHRED source branches. Because the recurrent encoder ingests velocity trajectories from t_0 to $t_0 + N_{lag}$, the predicted probability distributions $\rho(x_s)$, $\rho(y_s)$, and $\rho(z_s)$ at $t = t_0 + N_{lag}$ evolve as the window advances through the full recording. Independent predictions at each time step are made and concatenated to form a time-dependent source location estimation. The same sensor geometry as in Figures 6.5–6.7 is used, with 33 virtual stations distributed across the simulation domain.

Figure 6.8 summarizes the source location estimation results. Panel (a) shows the time-evolving probability density maps for the representative event (sample index 100035). Panel (b) shows the velocity waveforms from the selected sensors, ordered by their hypocenter distance, with the first arrivals detected by the Short-Time Average / Long-Time Average (STA/LTA⁹) method and marked by the red triangles. It can be seen from panel (a) that before any sensor records the first arrival, the predicted distributions for all three coordinates are broad and diffuse since no seismic information is yet available to the array. The predicted distributions progressively sharpen after the first sensor records an onset at the time marked by the solid red vertical line in panels (a) and (b). As more arrivals are captured in the encoding window,

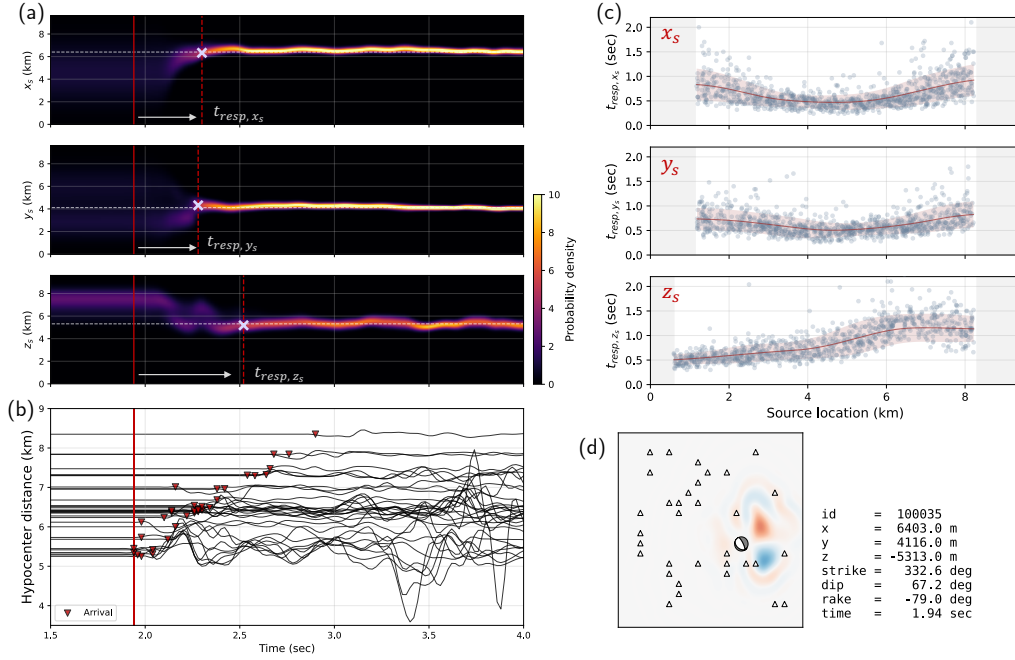


Figure 6.8: Source location estimation results for a representative test event (sample index 100035; see panel (d)) and across the held-out test set. (a) Time-evolving probability density maps for each hypocenter location axis (x_s , y_s , z_s). The solid red vertical line marks the time when the event is first detected by a sensor. The white dashed horizontal line marks the ground-truth source location, and the white cross marks the first model prediction with a minimum probability density of 5, at the time indicated by the red dashed vertical line. (b) Velocity waveforms (vertical component) from the selected sensors, ordered by their hypocenter distance, with the first arrivals marked by the red triangles. (c) Response time as a function of ground-truth source location for all test events. Each dot represents one event, with the red curve showing the running mean value. (d) Wavefield snapshot of this representative event showing the 33 virtual sensor locations (triangles), the source location (gray star), and the corresponding source parameters used for the simulation.

the distributions rapidly concentrate near the ground-truth location (white cross and white dashed horizontal line). We define the response time t_{resp} as the duration from the very first arrival to the time at which the predicted distribution first converges to the ground-truth location with a minimum probability density of 5, indicated by the red dashed vertical line in panel (a). For x_s and y_s , the response time is around 0.3 seconds after the first sensor picks the arrival, while for z_s , the response time is around 0.5 seconds after the first arrival.

We evaluate the response time across all test set events. Panel (c) shows the response time as a function of the ground-truth source location for each coordinate. No events exist in the shaded gray

region to leave buffer space for the absorbing boundary conditions. For x_s and y_s , the response time distribution is relatively symmetric, as events closer to the center of the domain tend to have shorter response times while events near the edges tend to have longer response times. This is consistent with the fact that events near the edge exhibit poor azimuthal coverage from the network, and therefore require longer time to constrain the source location with confidence. For z_s , events located near the surface (small z_s) tend to have shorter response times than deeper events, consistent with the later arrival of S-waves as event depth increases.

6.5 Conclusions

We presented EQ-SHRED and EQ-FLOW, a two-stage machine learning framework for reconstructing multi-task earthquake ground motion fields from sparse sensor observations. EQ-SHRED pairs a GRU-based recurrent encoder with SVD-compressive MLP decoders to recover the dominant large-scale structure of displacement, velocity, PGV, and PGA fields from only $\approx 0.3\%$ of the surface grid, while independent GMM source branches decode the shared latent space into probabilistic hypocenter estimates that converge within 0.3–0.5 seconds of the first seismic arrival. EQ-FLOW then refines the smooth low-rank Stage 1A output via conditional flow matching, restoring short-wavelength spatial complexity and localized near-source intensity peaks in PGV and PGA. In summary, our results demonstrate that this framework can be effectively used to reconstruct multi-task ground motion fields with high spatial resolution and to rapidly estimate earthquake source locations from a sparse sensor network.

It should be noted that the current framework is trained and evaluated on synthetic data from a single fixed Earth model. Future work will focus on 1) generalizing to multiple Earth models and/or sensor configurations through encoding of the velocity model and the sensor locations, 2) extending to broader magnitude ranges, 3) scaling to larger simulation domains with realistic 3D heterogeneity, topography, and attenuation, and 4) leveraging the reconstructed PGA and PGV maps for time-dependent seismic hazard estimation such as liquefaction potential prediction and landslide assessment.

A key challenge, however, is the persistent gap between physics-based simulations and real-world observations. Using purely observational data for self-supervised training is not physically practical, spatially and temporally dense ground-motion recordings do not exist for the scale of regional earthquake

monitoring. Therefore, physical simulations remain necessary to provide ground-truth labels, even though their fidelity is still being investigated. Hybrid physics-ML approaches, such as data assimilation, offer a promising path forward: a generalizable model can be pre-trained on synthetic data and subsequently refined using real observations, provided that the synthetic training distribution is sufficiently diverse.

Code and Data Availability

The original HEMEW^S-3D dataset used to select earthquake source parameters is available from Lehmann et al.¹³⁴. The SEM3D spectral element solver used for waveform simulation is available at <https://github.com/sem3d/SEM>. The EQ-SHRED and EQ-FLOW models are implemented in PyTorch²⁰⁰. The randomized SVD is computed using the SciPy implementation²⁸². The STA/LTA first-arrival detection uses the ObsPy implementation²⁷. All figures are produced with Matplotlib¹⁰⁹. The synthetic ground motion dataset and the codebase will be made publicly available upon submission.

*This chapter was published as Ni, Y., Denolle, M. A.,
Münchmeyer, J., Wang, Y., Feng, K. F., Suarez, C. G. J.,
Thomas, A. M., Trabant, C., Hamilton, A., Mencin, D. (2025).
A Review of Cloud Computing and Storage in Seismology.
Geophysical Journal International, 243(1), ggaf322. doi:
10.1093/gji/ggaf322*

7

A Review of Cloud Computing and Storage in Seismology

7.1 Introduction

Seismology has entered a “petabyte era,” where seismic networks and nodal array experiments routinely generate more data than traditional workstations and institutional clusters can store or analyze. Data

from more than 70,000 seismometers has surpassed 1 PB on the EarthScope Data Archive¹⁶. Novel technologies, such as Distributed Acoustic Sensing (DAS), which collect thousands of sensors per experiment, are also increasing the data storage needs and already surpassing the PBs of data (i.e., datasets exceeding 10^{15} bytes) collected and shared^{254,309,327}. State-of-the-art research networks and transfer tools now enable the movement of a petabyte of scientific data in about a day under optimal conditions (e.g., dedicated inter-institutional research networks with high-throughput protocols such as Globus and dCache), but remain challenging to move through normal internet bandwidth. Moreover, seismic data services standards well-adopted by data centers are not well suited for big-data seismology studies^{16,209}. While seismologists' workflows traditionally involve analyzing data by downloading from archives and working on-premise, this model faces significant difficulties with analysis that requires more than several TBs of data. Researchers are exploring cloud computing to address these bottlenecks, which brings code to data and offers scalable storage and processing.

To turn this data deluge into discovery, seismologists increasingly look to cloud platforms that put compute next to the archive. In this review, we use *cloud computing and storage* to mean shared remote computing and data centers run by global firms like Amazon Web Services (AWS), Google Cloud Platform (GCP), and Microsoft Azure, as well as European research clouds such as SURF and European Grid Infrastructure (EGI) that let geophysicists borrow computing clusters and storage on demand, pay only for the time used. Commercialized cloud computing emerged in the early 2000s, when the industry began providing capabilities such as large-scale object storage and on-demand computing. Geophysicists have traditionally used high-performance computing (HPC) centers to deliver tightly coupled, job-scheduled computing on a shared filesystem, often driven by and designed for large-scale numerical simulations such as wavefield simulations (e.g., SPECSEM^{121,204}) and earthquake dynamics (SeisSol⁹⁹). In contrast to HPC systems, cloud providers deliver elastic resources as metered services that users spin up and pay for only when needed. This encompasses several key design elements of their inner working, including virtual machines (VMs), cloud storage, and advanced services like databases and serverless services, which we introduce in this review. Practically, this means a researcher can open a web portal and launch virtual machines or clusters in several minutes. The cloud's hidden orchestration layer juggles requests by slicing real hardware into many small "virtual" ones, which gives users elastic

capacity but also means users share logically isolated hardware with others. That convenience translates into strengths (instant scale-up, pay-as-you-go, global access) and weaknesses (performance can vary, data may cross the public internet, long-term storage fees add up) that we further discuss. Harnessing cloud infrastructure could fundamentally change how seismologists handle big data, making analyses faster and more collaborative.

Cloud offers tremendous opportunities for easy access to object storage, a centralized, affordable, and widely accessible solution for massive data archives. In contrast to HPC, object storage enables hosting and sharing PBs of public geoscientific data^{3,84,344}. While individual data queries may have modest throughput speeds (10-1000 MB/s), the large parallelization capabilities of cloud systems allow throughput speeds comparable to those of HPC scratch systems (10-100 GB/s). Moreover, cloud providers are storing PBs of publicly available and free-access geoscientific data³. In seismology, hosting seismic data on the cloud is rising, as we illustrate in Figure 7.1. The Southern California Earthquake Data Center (SCEDC) was the first to provide archives of regional seismic networks as open datasets on the commercial cloud³²². It was recently followed by the Northern California Earthquake Data Center (NCEDC) and the EarthScope-operated Seismological Facility for the Advancement of Geoscience repository (formerly Incorporated Research Institutions for Seismology Data Management Center, IRIS DMC). Cloud has also been a promising storage solution for DAS data (e.g., PoroTomo, <https://registry.opendata.aws/nrel-pds-porotomo/>, and Ridgecrest DAS³²²).

A typical entry point for scientists to cloud computing is through a freely accessible JupyterHub with a backend running on cloud platforms. Open-access JupyterHubs with notebooks (e.g., Binder, Google Colab, and EarthScope GeoLab) lower the entry barrier to the cloud by giving users a ready-made Python environment. Users can automate software-to-infrastructure, using tools such as `repos2docker` to containerize software and automatically provide access to cloud-hosted virtual machines. In particular for seismologists, Krischer et al.¹²⁷ has pioneered the use of cloud-hosted Jupyter notebooks, utilizing modest yet free Binder instances provided and donated by various cloud providers (e.g., OVHcloud as of 2025). Alternatively, Google Colab provides free access to modest-sized virtual machines, which utilize GCP resources with pre-defined Python environments. Cloud is also an on-demand platform to host educational materials⁵⁹.

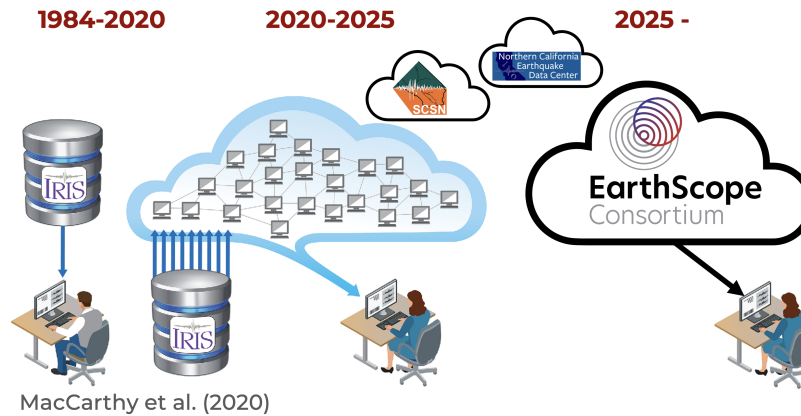


Figure 7.1: Evolution of cloud computing in seismology: before 2020, most computational workflows involved downloading data from the Incorporated Research Institutions for Seismology Data Management Center (IRIS DMC) and other data centers and working locally. Between 2020 and 2025, seismologists have investigated the use of elastic computing by pulling data from existing archives and processing directly on the cloud (e.g., MacCarthy et al. ¹⁵²). At the same time, two regional seismic networks copied their archives of earthquake catalogs and seismic waveforms to Amazon Web Services (AWS): the Southern California Seismic Network (SCSN) and the Northern California Seismic Network (NCSN). Since 2025, the EarthScope Consortium has migrated its petabyte-scale archive to the cloud, enabling researchers to pull and compute directly on the cloud.

Considering these basics, this article reviews how cloud computing has been applied in seismological research. We discuss storage solutions for seismic data (Section 7.2), databases (Section 7.3), the various types of computing resources (Section 7.4), showcase experiments conducted at scale on large archives of broadband seismic data (Section 7.5), and cloud-enabled visualizations (Section 7.6). We also present a series of experiments that have targeted archetypes in big-data seismology: 1) data mining using deep learning models to detect seismic events, and 2) ambient field seismology that requires intensive generation of cross-correlation at scale. Both tasks are characterized by a high data intake and large computational requirements, yet differ in output and processing specifics. In particular, we focus on the specific requirements of each workflow and how they affect the choice of cloud tools. Finally, we present our recent experience in running a cloud seismology workshop (Section 7.7) and discuss the opportunities, cost, and challenges in cloud computing for seismological research (Section 7.8).

7.2 Cloud Storage

7.2.1 Cloud-hosted Data Archives

Transferring and sharing data between institutions is essential for large-scale collaborative research in seismology. Software from command-line tools (e.g., `sftp`, `wget`) to hosted services (e.g., Globus, Allen et al.⁸) is made available to securely and efficiently share data over the internet to facilitate the transfer of moderate to large-sized datasets. However, on-premise storage and bandwidth limitations constrain point-to-point transfer efficiency. At the same time, the host takes responsibility for processing requests and maintaining stable data access, usually at the cost of additional man-power. Cloud storage provides an accessible solution for hosting and sharing scientific data with exceptional scalability and durability, along with improved findability, accessibility, and reusability³.

Cloud object storage offers a significant advantage by enabling the storage of large datasets and scalability for massively parallel queries from both within and outside the cloud. For instance, AWS provides the Simple Storage Service (S3), a scalable object storage service that allows users to store data as objects without practical limits on total data volume. Azure (Blob storage) and GCP also provide similar services. In contrast to the filesystem used in HPC, files are saved as individual objects and organized in a bucket with a flat structure. S3 objects can be identified by a prefix and a key, much like a folder and its filename in a filesystem, which users are more familiar with. While S3 is not a Portable Operating System Interface and differs from the Unix-style semantics most researchers are used to, the departure is advantageous: freeing storage from strict POSIX rules lets cloud object stores scale to billions of files, offer global access from any service, and replicate transparently across regions.

Seismic data centers curate and offer vast amounts of invaluable data through the International Federation of Digital Seismograph Stations (FDSN) web services, which standardized API queries for seismic data and metadata delivery^{95,110,210}. Cloud storage fundamentally transforms data center solutions, improving robustness, durability, and stability to data management within the facility, resilience to storage read spikes, and proximity between storage and compute nodes on the user end²⁴. The SCEDC has been the pioneer in migrating, hosting, and publicizing ~ 150 TB of continuous data onto the cloud^{322,343}, followed by the effort of NCEDC of their ~ 190 TB, and most recently, the EarthScope Consortium seismic data archive, which has surpassed 1 PB. Direct access to cloud-based archives enables the development of cloud-native workflows, which we will discuss in Section 7.5. For instance, the

workflow’s throughput could not be achieved through the FDSN `fdsnws-dataselect` web service. However, we note that there are examples of petabyte-scale storage services with high throughput operated by scientific institutes, for example, for particle physics¹⁶⁷.

Data centers migrating their repositories, or offering copies, in cloud environments must consider how data will be discovered and accessed directly by researchers. Whereas previously, these repositories were only accessible behind services that acted as abstraction layers, the organizations’ data can now be exposed to direct access (e.g., back-end or raw files). This is important to avoid adding processing bottlenecks, limiting the ability to subset data, and slowing down data queries and downloads. At the same time, such direct access increases the burden on the user to write efficient code, downloading only the relevant sections of data, a task previously handled by the access layer. Operational requirements like controlling access to restricted data, log data use, etc., may slow down the data access, and careful design must be considered depending on the application. New services and software are needed to support the efficient discovery and use of large-scale analyses beyond the simplest cases.

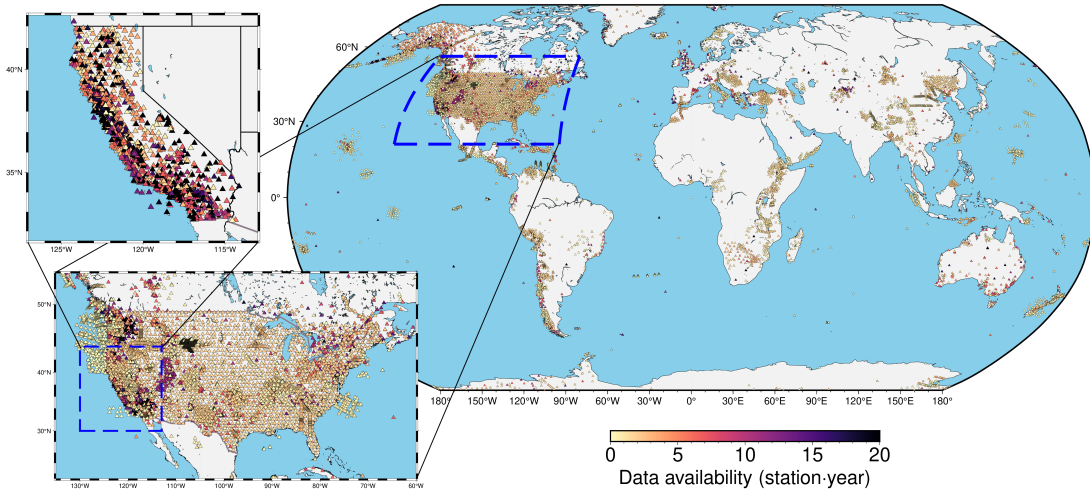


Figure 7.2: Station map of cloud-hosted data: A total of about 1.3 PBs of seismometer miniSEED data is hosted on Amazon Web Services (AWS) cloud storage: the EarthScope Consortium seismic data archive, and the Northern and Southern California Earthquake Data Center. Each triangle indicates a seismic station and is color-coded by its data availability.

7.2.2 Data Formats on Object Storage

Array Data

Historically, data centers have relied on SEG-Y (for active seismology) and Standard for the Exchange of Earthquake Data (SEED, for passive seismology) as primary data formats⁹¹. SEG-Y, designed initially for tape storage in the 1970s, remains widely used due to regulatory requirements in the petroleum exploration industry. Still, it suffers from poor parallel read performance and high I/O latency in modern cloud environments. Meanwhile, SEED (or more accurately, miniSEED) is also actively used for seismic time series archiving and shipping. Specifically, the miniSEED standard in its 2+ version adopted the paradigm that separates waveforms (i.e., miniSEED with minimal metadata) from their metadata counterpart (i.e., dataless SEED with no time series). This only makes this format partially cloud optimized because it has no native support for object-based data partitioning, indexing, or scalable metadata integration. It also requires additional infrastructure to serve efficiently from cloud storage (e.g., an indexing layer or a metadata catalog).

More recently, the HDF5 format (Hierarchical Data Format version 5²⁶³) enables researchers to design their data structures and storage. HDF5 supports a customizable and flexible hierarchical schema, allowing the storage of multidimensional seismic waveforms and metadata (e.g., Krischer et al.¹²⁶, White et al.²⁹⁶). On the other hand, such a self-describing structure also presents obvious limitations when used at scale in distributed cloud computing. The monolithic nature of HDF5 files introduces overhead for metadata handling and parallel access, while losing efficiency on byte-range requests and compressed blocks, i.e., read subsets of large datasets¹⁸⁷. Despite efforts from open-source projects such as `h5coro` (H5 cloud-optimized read-only library, <https://github.com/SlideRuleEarth/h5coro>) and `kerchunk` (<https://github.com/fsspec/kerchunk>) that were made to optimize HDF5 for cloud object storage, they are often read-only solutions and do not fully resolve challenges from generalized file structures.

Recognizing these limitations, geospatial data initiatives such as Pangeo and EarthCube have pioneered the adoption of cloud-optimized array formats like Zarr and TileDB, which avoid hierarchical formats and instead store multi-dimensional arrays in a chunked, compressed, and distributed manner. Pangeo's use of Zarr, for instance, has enabled massive parallel processing of gridded climate and remote

sensing datasets, demonstrating $\sim 10\times$ speedup in read performance compared to HDF5 when accessed in parallel from cloud object stores³. Similarly, TileDB has proven effective for sparse geospatial data, including global navigation satellite system (GNSS) and seismic sensor arrays, allowing efficient subsetting and time series access⁹². These technologies are now being adapted for seismology, where efforts such as those by Ni et al.¹⁸⁷ demonstrate that converting DAS data from HDF5 to Zarr/TileDB results in significant memory and compute time improvements. The proliferation of these new open formats and their variants may be an obstacle to the sustainability of our software, so seismological workflows must stay format-agnostic and ready to pivot between, or simultaneously support, multiple storage layouts as standards evolve.

Point Sensors

Seismological research spans a wide range of temporal and spatial scales, requiring storage solutions that can support both short-duration, multi-station queries and long-duration, station-centric analyses. These workflows stress storage along different query axes: some pull data from thousands of stations but only short time windows (e.g., large-N arrays, ambient-noise cross-correlation), whereas others retrieve long, continuous histories—months to years—for each station in the network (e.g., ambient-noise monitoring, template matching). Small-object storage enables efficient writing and retrieval, often providing superior performance relevant to seismological broadband seismic data (e.g., 10 MB per day for a typical 100 Hz 3-component broadband data).

Cloud-optimized formats designed for petabyte-scale datasets—such as Zarr and TileDB—have emerged from the geospatial data community to support large queries on colocated compute, enabling efficient streaming of large data chunks. These formats are particularly advantageous for array-based datasets, including nodal deployments and DAS records (e.g., Ni et al.¹⁸⁷). However, for many seismological applications, which typically involve small, targeted data requests, established formats like day-long miniSEED files remain highly effective. miniSEED supports efficient remote access to small waveform segments, making it well-suited for near-real-time, station-centric data streaming and event-driven workflows.

The data centers mentioned above all host seismic time series in the miniSEED format on AWS

S3. There are differences in conventions for organizing files in the S3 bucket regarding data granularity. Specifically, SCEDC and NCEDC store one channel per object, whereas the EarthScope seismic data archive groups all channels per station in a single file. Despite hosting more small objects, the former structure exhibits high efficiency when querying data from subset channels or locations since no redundant bytes are read. The latter structure usually requires an external database that indexes files to facilitate data query (e.g., using mseedindex, <https://github.com/EarthScope/mseedindex>), although a fully cloud-native access library has yet to be developed. The various bucket structures indicate that object naming (prefix and key) has not yet reached a standard across these data centers because the new paradigm of direct access renders these previously hidden implementation details part of the user interface.

Delivering seismic station metadata in a cloud-native way remains an open challenge. At present, the FDSN web service for metadata still operates on on-premises servers at each data center, creating a bottleneck that is not resilient to large-scale or burst queries. As an initial step toward addressing this, SCEDC and NCEDC have publicized community-standardized StationXML files – containing the complete history of station metadata and instrumental response – directly in their respective cloud storage. While this approach makes metadata accessible from the cloud, StationXML itself is not optimized for object storage: reading and parsing full XML documents is often slower and less efficient than targeted queries via the FDSN web service. Significant work remains to design and adopt metadata formats and delivery mechanisms that are truly cloud-efficient, enabling scalable, selective, and low-latency access.

7.3 Cloud Databases

Seismologists need databases to manage station metadata and curated datasets such as earthquake catalogs and phase picks. A database is an organized data collection designed for efficient storage, retrieval, and management. The simplest form of a database, which is termed a flat-file database, relies on files stored in hierarchical directories, where data is typically structured in plain text formats (e.g., Comma Separated Value (CSV) and JavaScript Object Notation (JSON)). Although flat-file databases are useful for small-scale or static datasets, their limitations in scalability, query flexibility, and concurrent access become

apparent when managing large volumes of data.

Modern databases are broadly categorized into relational and NoSQL (non-relational Structured Query Language) databases, each addressing distinct needs. Relational databases were developed decades before the advent of cloud computing. They organize data into structured tables of rows and columns governed by the relational model. The relational model handles structured data with a strict schema and transactional consistency, making it ideal for curated seismic metadata (e.g., event catalogs or station metadata) scenarios.

In contrast, NoSQL databases became popular during the big data era to address the challenges of scalability, schema flexibility, and heterogeneous data types, which are common in modern seismological research. NoSQL databases include document stores (e.g., compatible with a tool such as MongoDB), key-value stores, wide-column stores, and graph databases. The MsPASS framework is an example of using MongoDB to manage large-scale seismic data²⁹¹. NoSQL databases thrive in cloud environments due to their flexibility in managing semi-structured data (e.g., processed waveforms or phase picks) and scalability across distributed systems. This aligns seamlessly with the growing reliance of seismological research on high-volume, multi-modal datasets, such as machine learning-ready archives.

Among NoSQL databases, document stores like MongoDB and its AWS implementation, DocumentDB, exemplify the advantages of schema flexibility in modern seismological workflows. These systems store data as JSON-like documents, enabling researchers to consolidate heterogeneous datasets, such as workflow parameters, phase picks, and semi-structured metadata, into a single database without rigid schema constraints. This flexibility is particularly valuable in seismology, where evolving research workflows often generate metadata with new and inconsistent attributes. DocumentDB further simplifies scalability by automating sharding and replication in cloud environments, allowing distributed storage of large-scale datasets while ensuring low-latency access. Modern cloud databases thus can store the massive output of seismic processing (e.g., billions of phase picks or cross-correlation measurements) and enable quick queries. We will explore the application of cloud-hosted databases in seismology in the following cloud workflow examples.

7.4 Cloud Compute

Cloud providers offer diverse infrastructures and services well-suited to seismologists' diverse needs. Orchestrating these various cloud services to support seismological research involves integrating compute, storage, and software tools provided by cloud vendors using service orchestrators such as CLI (command-line interface) tools, Python-based Software Development Kits (SDKs, e.g., `boto3`, `google-cloud-python`), or web-based platforms (e.g., AWS Step Functions, Google Cloud Workflows). Scientists developing workflows on the cloud face challenges when providers update their services and adapt open-source software, but this is possible with the help of research software engineers¹²⁴. To detect and adapt to cloud service changes efficiently, ensuring workflow resilience requires version pinning, modular pipeline design, and automated testing. On the other hand, the diversity of resources in the cloud, in particular, the variability of available machines and services, is typically much higher than in on-premise HPC systems. While numerous large-scale national science infrastructures exist³, they are typically much more homogeneous. For researchers, the cloud infrastructure allows them to tune the requested resources precisely to their demand. At the same time, this increased flexibility also comes with increased effort for identifying the optimal resource allocation.

The most basic unit on the cloud for scientists and many other higher-level cloud services is a virtual machine (VM). VMs are the virtualization of hardware and packetization of operating systems, enabling users to access and share physical infrastructure on demand. The host operating system logically isolates different VMs running on the same machine. VMs can be configured to the user's preference, including the number of vCPUs, Random Access Memory (RAM), local storage, and additional resources like GPUs, all selected from a set of provisioned templates. The flexibility of VMs is fueling a democratization of large-scale computing. This review focuses on the type of parallelization well suited for cloud platforms, one of distributed memory, sometimes referred to as “embarrassingly parallel.”

³e.g., Jean Zay (<http://www.idris.fr/eng/jean-zay/jean-zay-presentation-eng.html>) or JUWELS <https://apps.fz-juelich.de/jsc/hps/juwels/index.html>

7.4.1 Batch Computing

As a service commonly available in cloud systems, such as Azure, AWS, and GCP, Batch computing involves the parallelization of jobs on cloud instances, similar to the job arrays in the SLURM scheduler system³¹⁹. The batch computing service is particularly useful for parallelization when the job array shares the same code base and differs only in passing arguments, whether the jobs are simply command lines or containerized tasks. While the embarrassingly parallelizable job runs independently, the multi-node parallel job allows internode communication through message passing libraries (e.g., MPI⁹⁰). Such a framework enables single jobs spanning multiple computing instances as a cloud-based on-demand cluster for high-performance computation applications^{34,56,344}.

Similar to the scheduler in a modern HPC system, an auto-scaling mechanism dynamically provisions and scales cloud resources based on the volume and requirements of submitted workloads. Such a mechanism adjusts the number of running instances or containers to ensure that resources match the computational requirement without over-provisioning. Auto-scaling can be triggered by pre-defined metrics such as CPU utilization, memory usage, request rates, or job numbers, allowing cloud environments to handle traffic spikes and workload fluctuations efficiently. This capability is essential for maintaining high availability, improving fault tolerance, and optimizing resource utilization and spending, making it a key feature in modern cloud infrastructure.

7.4.2 Serverless

Serverless computing is a cloud-native execution model that abstracts infrastructure management, allowing developers to deploy code without provisioning servers. By automatically scaling resources in response to demand, the serverless architecture enables users to focus on their applications rather than managing operational overhead. This is particularly transformative for seismic early warning systems, where latency-sensitive processing of real-time data (e.g., event detection) requires rapid, event-driven workflows. In a serverless framework, cloud providers like AWS (Lambda), Azure (Functions), and Google Cloud (Cloud Run Functions) dynamically allocate compute resources in response to user requests.

A compelling example of serverless computing in seismology is demonstrated by Mohapatra et al.¹⁶⁸,

who tested a hybrid cloud-local workflow using AWS Lambda and the MsPASS framework. Their study found that downloading raw seismic data (i.e., the 40 million-record USArray dataset) to local HPC clusters created untenable bottlenecks, requiring approximately 462 days for single-worker processing. By shifting preprocessing to serverless functions (e.g., noise reduction, metadata filtering), they minimized data transfer volumes and achieved throughput comparable to local HPC processing. The authors conclude that doing some or all processing on the cloud in this fashion will be essential for any processing involving large volumes of data already stored on the cloud. While hybrid workflows incur cloud costs, they bypass local network limitations, offering a scalable path for modern seismology.

7.4.3 Visualization

Effective visualisation turns today’s petabyte-scale seismic data into insight at a glance. Many groups rely on browser-based notebooks that connect directly to cloud object storage. Browser-based notebooks—commonly hosted in JupyterHub or Google Colab—can mount cloud object storage directly (e.g., using tools such as `s3fs` for AWS or `gcsfs` for GCP), read only the data chunk required for a plot, and render interactive figures in real time. When those exploratory notebooks mature, researchers often containerise them into lightweight dashboards (such as `Dash` or `Streamlit`) that run on serverless platforms.

Because ‘serverless’ platforms start containers only at researchers’ demand (e.g., during an earthquake crisis or a teaching lab), hosting costs remain a few USD per month in quiet periods. When visualising full 3D wavefields or large seismic datasets such as from DAS, teams spin up short-lived GPU instances running remote desktop tools such as `ParaViewWeb`, `PyVista`, or cloud-proprietary software such as `NICE DCV` for AWS, to stream pixels, not raw data, to the user.

Once these visualization environments are running, the challenge becomes delivering data to them efficiently. Using cloud-optimized formats such as `Zarr` and `TileDB`, seismic datasets can be stored in small, independent chunks, allowing rapid retrieval of specific subsets — for example, waveforms from a given station and time window, or a horizontal slice through a 3D wavefield — without downloading entire files. Low-resolution previews, map tiles, or simplified 3D model snapshots can be served instantly to support smooth navigation and exploration. More computationally demanding visual products,

such as spectrograms, cross-sectional wavefield views, or machine-learning-derived feature maps, can be generated asynchronously in the background, with only the resulting lightweight images or metadata returned to the user interface. This workflow enables seismologists to interactively explore data, compare results, and monitor networks in real time through a web browser, while keeping bandwidth and latency low. The products of these visualizations still require robust databases to store and organize derived measurements, as discussed in Section 7.3.

7.5 Cloud-Native Applications in Seismology

Seismological analyses often require large computational bursts followed by long idle periods of analyzing, a usage pattern that is tailor-made for on-demand cloud resources. Early adopters, therefore, re-hosted existing HPC pipelines in the public cloud to rent capacity only when peaks arose, a form of cloud-assisted “lift-and-shift” workflow. Wang et al.²⁸⁸ was among the first to demonstrate the use of public cloud computing for seismic data processing at the TB scale and performed noise cross-correlation using the Aliyun cloud service, specifically with the Batch service and the cloud object storage. MacCarthy et al.¹⁵² used the FDSN web service to request data on the fly while detecting harmonic tonal noise. They successfully scanned 6 TB of USArray data within 4 days on a Kubernetes cluster with 50 EC2 nodes. Witte et al.³⁰⁴ built a serverless seismic imaging application ported from the HPC platform, dynamically scheduling jobs and provisioning computational resources using serverless cloud, demonstrating excellent cost efficiency, scalability, and performance competitive with on-premise HPC clusters. Similarly, Zhu et al.³⁴² designed an integrated earthquake detection workflow with containerized submodules, i.e., data streaming, phase picking, association, and event location. The auto-scaling mechanism was implemented at both the cluster and cloud platform levels, enabling the automatic provisioning of computational resources based on job load. The paradigm of these cloud-based workflows was summarized by MacCarthy et al.¹⁵² as comprising three primary components: the infrastructure, cluster management software, and domain-specific research software. These “lift-and-shift” studies share a three-tier pattern: infrastructure (cloud VMs and containers), cluster management layer (Kubernetes, batch, step functions), and domain software (e.g., noise cross-correlation, full waveform inversion, earthquake detection). Computing efficiencies depends on how tightly these elements are coupled.

However, these workflows assumed that data could be fetched on demand, but pulled data from outside the cloud; such a prerequisite cannot be easily met in the cloud for data-intensive tasks. Transferring the raw data or data products can be time-consuming and expensive^{187,288}. For example, Zhu et al.³⁴² spent $\sim 50\%$ of total job time downloading waveforms through the FDSN web service. Requesting data at scale may also pose challenges for data centers that receive unpredictable heavy traffic and clusters where big data is saved and managed on-site. To overcome these bottlenecks, recent efforts embrace “cloud-native” workflows: 1) harness direct access to the cloud-hosted data to avoid copying data^{150,322}, 2) leverage cloud-managed services, and 3) employ containerized software to ensure portability, consistency, and ease of deployment across platforms. To illustrate the cloud-native workflows for seismology, we present two contrasting research workflows in seismology that have benefited from the cloud systems: 1) cross-correlation for ambient noise seismology and 2) earthquake catalog building workflows. Figure 7.3 illustrates the two alternative workflows with data flows and associated cloud services best used in large-scale jobs.

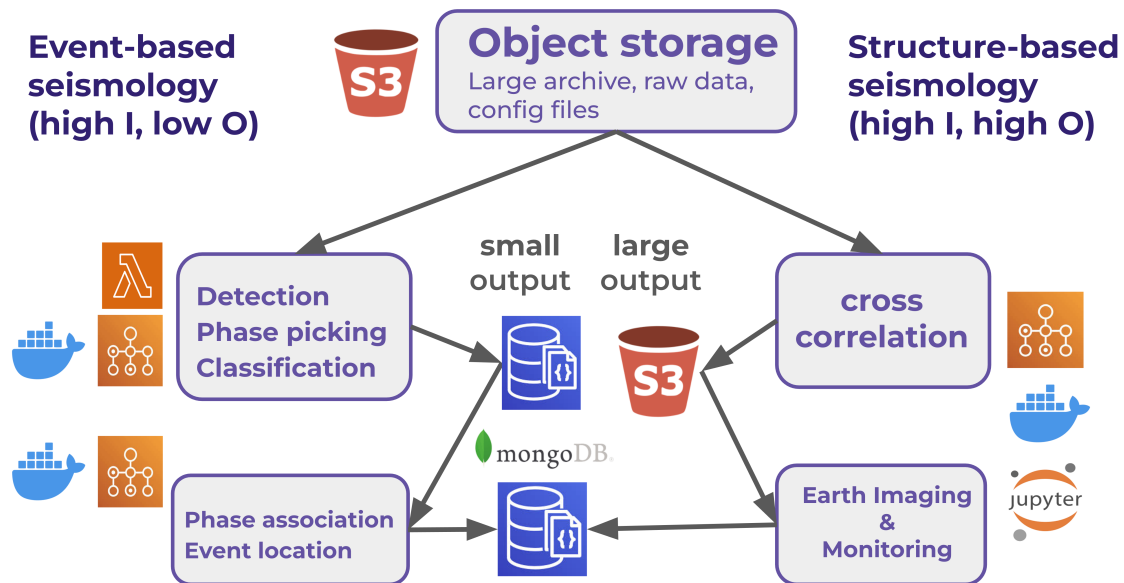


Figure 7.3: Two canonical workflows in seismology: Event-based seismology that reads large volumes of seismic data (high I – high Input) but outputs low volumes of data in databases (low O – low Output), and structure-based seismology that reads large volumes of seismic data (high I) and outputs large volumes of seismic data (high O). The first workflow outlines the basic steps in generating a seismic event catalog. The second workflow describes how to extract seismic properties of the subsurface with ambient field seismology, which generates high data volumes of ambient noise cross-correlations.

7.5.1 Workflow 1: Large Ambient Noise Seismology

Ambient noise seismology is the methodology that utilizes continuous seismograms, typically dominated by a diffuse, ambient seismic field, to extract spatial or temporal variations in seismic wave speeds. This method has been widely used for 1) Earth Imaging reconstruct high-frequency Rayleigh wave and image shear wave structure where seismic stations are located (e.g., Shapiro et al. ²⁴¹) and no longer rely on rare earthquakes, and for 2) Earth monitoring by exploring changes in subsurface structure by subtle phase shifts on the coda of cross-correlations (e.g., Sens-Schönfelder and Wegler ²³⁸).

The method relies on the cross-correlation of short time series between channels, presenting some of the most significant challenges in computational seismology. The cross-correlation typically uses short windows, ranging from minutes to a few hours, recorded at pairs of seismic channels, and stacks these over days to years of data. Thus, the workflow scales quadratically with the number of channels N , a step that favors shared memory processing, and linearly with the number of windows to stack T , a step that favors distributed memory processing. The rise of array seismology with $N > 100$ is a real computational challenge. Because the workflow often entails storing cross-correlation functions, including intermediate steps such as substacking, cross-correlation may involve writing TBs of files. Several efforts have been made for open-source and large-scale computing of ambient noise cross-correlations, some leveraging CPU-based clusters (e.g., Jiang and Denolle ¹¹⁴, Makus and Sens-Schönfelder ¹⁵⁵), others leveraging heterogeneous computing with CPU and GPU (e.g., Clements and Denolle ⁴⁸, Fichtner et al. ⁷⁷, Ventosa et al. ²⁷⁹, Zhou et al. ³³⁶).

Given two canonical seismological approaches for *Earth imaging* and *Earth monitoring*, the workflow to compute cross-correlation functions is multi-step, and their optimal parallelism strategies differ. First, the cross-correlations are performed independently on synchronous time series, which permits distributed memory parallelism, often referred to as “embarrassingly parallel,” and scales only with the overall period of the instrumental record. Second, the cross-correlations are done on *pairs* of seismic channels, and a given channel window of data could be read once and cross-correlated over all other channels with $N(N - 1)/2$ pairs. This strategy often employs multi-threading with shared memory, parallelization across channel pairs, and leveraging GPUs to accelerate the correlation step ^{48,77}. When the data is too large for the memory available, local storage of intermediate products, such as the

Fourier transforms²⁸⁸, or low-rank factorizations¹⁵⁹, and parallelization over groups of station pairs is also possible (e.g., C4 project^{233,234}).

Cloud infrastructure is particularly well-suited for ambient noise seismology, given its significant data throughput (reading and writing) and parallelization capabilities. Several attempts to perform ambient noise cross-correlations on the cloud demonstrated the speed and scalability of adapting cloud infrastructure. Wang et al.²⁸⁸ developed a parallelization scheme to independently calculate groups of channel pairs and perform massive daily cross-correlations, totaling 300M, over 10 hours of processing on nearly 1000 virtual machines using the Aliyun cloud service. Ni et al.¹⁸⁷ performed DAS cross-correlations on AWS using cloud-native workflows and achieved 300M of daily cross-correlations over 64 instances in 24 hours, spending less than 20 USD. Clements et al.⁴⁶ and Schmitt et al.²³³ developed a workflow on AWS that approached “cloud-native” by streaming from S3 to EC2, generating cross-correlation locally, saving the results on disk, and uploading them to S3.

We now present **Cloud-Native NoisePy**, a new version of Jiang and Denolle¹¹⁴ that has been updated with I/O for cloud-based data archives, enhanced object-oriented Python programming, including parallelization, flexibility in computing platforms, and continuous integration. NoisePy leverages cloud object storage for massive I/O parallelization and short-term storage of temporary data, such as daily cross-correlation functions, as illustrated on the right-hand side of Figure 7.3. NoisePy employs two primary parallelization strategies to optimize performance. The first approach utilizes the Batch compute service to execute each daily job of processing and inter-channel cross-correlation concurrently and independently, using the same container with different data. Within each job, NoisePy utilizes native Python multi-threading for parallelization across several steps, including reading data, pre-processing, computing the Fourier transform, cross-correlation, and writing daily results, which are stacked and saved as compressed NumPy `.npz` files back to storage (file structure and format were found optimal when experimenting with AWS S3). After processing all daily data, a final aggregation step combines the results to produce long-term correlation stacks. This architecture efficiently handles large-scale data, generating TBs of cross-correlation outputs in the cloud.

We present the results of an experiment in which we ran NoisePy on one year of SCEDC data from Jan 1, 2022 to Jan 1, 2023. We cross-correlated all of HH? channels that included 288 stations and about

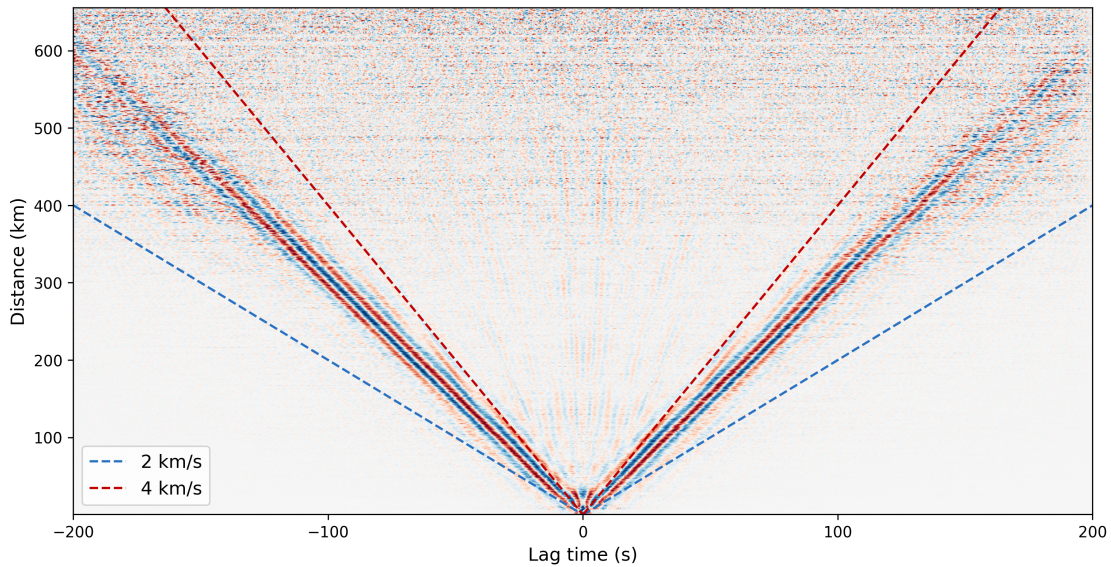


Figure 7.4: Ambient Noise Cross Correlation using 1 year of data from the Southern California Earthquake Data Center: all data are publicly available in the SCEDC AWS cloud storage (`s3://scedc-pds`). The Z-Z component cross-correlation functions are bandpass filtered between 1 and 10 seconds.

43,000 station pairs. Ran on AWS Fargate with up to 64 instances, it took 11 hours of compute time to generate 6.2 million files of daily-stacked cross-correlation, with a volume of 1.6 TB on S3, spending about 250 USD on SPOT pricing. The second step on a similar Fargate cluster took 1.5 hours and generated the final stacks of all inter-channel cross-correlations, totaling 23 GB of data and 46,000 S3 objects. We present the results of the 1-year stacked ZZ component of the cross-correlation, bandpass filtered between 1-10 seconds, sorted by the inter-station distance in Figure 7.4. We find the convergence of the correlation functions past 600 km of inter-station distances. Seismic tomography from these data products will involve extracting phase and group velocity measurements from these cross-correlations and inverting the frequency-dependent velocity curves into a shear-wave velocity model. Our experiment demonstrates the case of scalable data processing for tomography applications.

7.5.2 Workflow 2: Earthquake Catalog Building

Earthquake catalog building is a complicated, multi-step workflow that ingests raw time series data and outputs point clouds of earthquake locations and their attributes. The main steps are to detect events, identify the time at which seismic phases (typically *picking* P and S arrivals) arrive, associate them

with a specific origin (event), possibly incorporate location using 3D Earth velocity models, relocate them using double-difference relocation, and calculate source parameters, such as magnitude and focal mechanisms. Each step has been explored using machine learning. In particular, for phase picking, deep learning has proven highly successful. Network-based analysis typically requires gathering multiple-station data simultaneously, and has benefited greatly from U-Net and graph networks (e.g., Clements et al. ⁴⁷, Münchmeyer et al. ¹⁷⁹, Sun et al. ²⁶⁰). Most workflows are a sequence of modules (e.g., Retailleau et al. ²¹⁴, Walter et al. ²⁸⁴, Zhang et al. ³³⁰, Zhu et al. ³⁴²), where modules can be adapted according to the user preferences.

The computational efficiency of these workflows matters when considering large-scale deployment. Similar to the first workflow presented, we break down the computational efforts into two types of parallelization. The first step involves extracting features from raw data, including the arrival times of P and S waves, the amplitude, and possibly the polarity of these waves. This process can be independently calculated for each data window as a separate job and is amenable to massive parallelization. The second processing step requires aggregating these features across the stations and benefits from multi-threaded parallelization. Both steps can be easily orchestrated on cloud systems, which was first pioneered by Zhu et al. ³⁴² by including deep learning phase picking, association, and relocation using double difference, and by Pierleoni et al. ²⁰⁷ for Internet-of-Things (IoT) early warning systems whereby picking is done at the seismometer level, and location is done on the cloud. We present here another cloud-native workflow for the basic steps of earthquake catalog building, utilizing the `SeisBench` ecosystem³⁰⁷ for phase picking, which we refer to as `QuakeScope`.

`QuakeScope` orchestrates its seismic catalog workflow entirely on AWS Batch (Fargate) and a MongoDB-compatible DocumentDB cluster for outputting detection attributes and checkpointing. The basic unit of the workflow is a Python job, composed of four steps: (i) obtain a day-long time series of miniSEED data, (ii) process the continuous waveform with a phase picking model implemented in `SeisBench`, such as `PhaseNet`³³⁷ or `Earthquake Transformer (EqTransformer)`¹⁷⁴, (iii) remove the instrumental response to extract the amplitude of each detection, (iv) write the resulting detections to a DocumentDB. As jobs (i) and (iv) are I/O-bound, while jobs (ii) and (iii) are compute-bound, we implemented an asynchronous processing using the `asyncio` Python module. This means that the job

processes data simultaneously, loads the next day, and still writes the picks from the previous day. This increases throughput substantially and thereby reduces resource costs. The individual steps communicate through limited-size First-In-First-Out queues to avoid excessive memory overheads.

We present the results of the 2025 Data Mine experiment, where we scanned the complete EarthScope Consortium seismic data archive (1 PB) and the SCEDC and NCEDC AWS-hosted open data, each with 150 TBs, with QuakeScope. We describe in Figure 7.5 the evolution of our jobs on AWS. Our AWS allocation had limited quotas for 1,500 jobs (12,000 vCPUs) to be used simultaneously, a tripled quota relative to the default limits. We manually launch each year, from 2025 going back in time, as a queue of jobs, and experimented with the EarthScope back-end servers as a pilot experiment. During the initial stage, we stress-tested the EarthScope archive. We progressively launched the jobs on Fargate SPOT (a lower-cost queue that is less “on-demand”). In the EarthScope experiment, our progressive load of jobs in the queue demonstrated the resilience of the backend system, allowing us to launch a larger batch of jobs by hour 40 in the experiment. For the NCEDC and SCEDC experiments with a smaller dataset of approximately 300 TB, we launched all jobs simultaneously, reached our quotas, and completed them in 12 hours. The performance of cloud systems remained constant, demonstrating that this process can be easily accelerated by allowing for more concurrent processes. See Ni et al.¹⁹³ for a complete report of this experiment and the data product.

7.6 Community Model Visualization and Dissemination

Within the seismological community, there is an acute need for models of natural systems to support both basic and applied research. For example, models of seismic wave velocities within the Earth are essential for interpreting tectonic history and evolution⁵⁷, for generating accurate ground motion estimates using earthquake simulations (e.g., Glehman et al.⁸⁵), and for enabling a wide range of analyses related to seismic hazard and risk. Similarly, geometric models of earthquake-producing fault structures, along with associated metadata such as slip rates and earthquake histories^{208,236}, form the foundation of seismic hazard analysis²⁰⁵. These needs have driven the development of community models, which are collaboratively developed, synthesis-based representations of natural systems that are maintained and shared by a broad group of researchers with expertise in the relevant domain (e.g., Aagaard et al.¹).

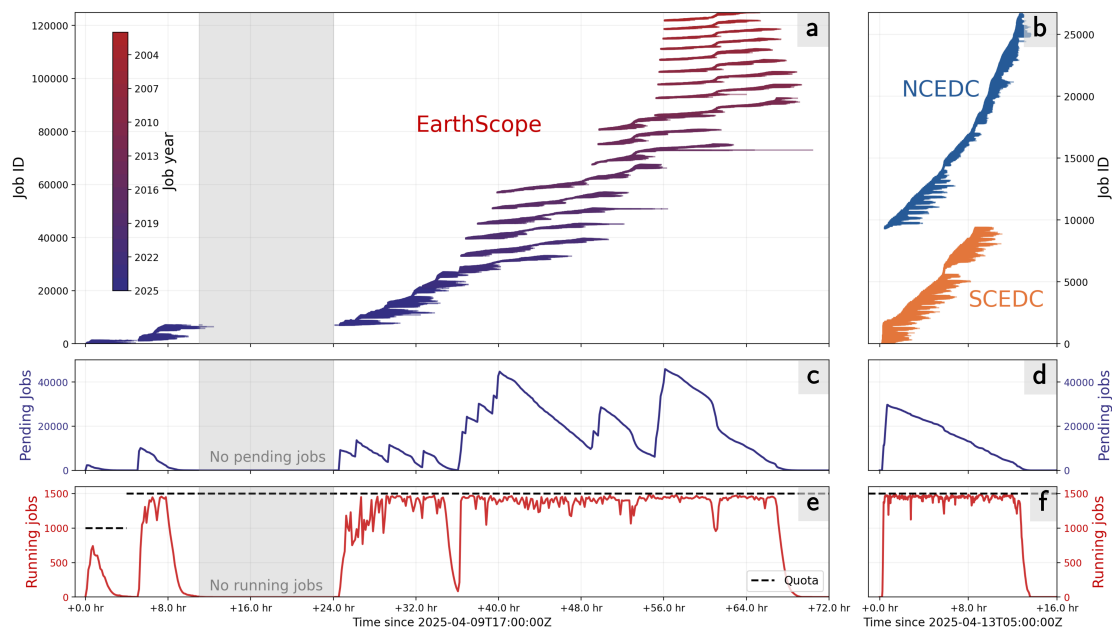


Figure 7.5: Evolution of job statistics in the 2025 Data Mine experiment: top panel has a line between the start and end time of each job that ran on Fargate. Each year’s worth of data is sent manually as a separate batch of jobs, color-coded for EarthScope. The middle panel shows the number of pending jobs in the Fargate queue. Jobs enter the queue and are scheduled to run until the quota of 1500 jobs is reached. The dips in the number of active jobs are attributed to our slower job orchestration. One that may be attributed to SPOT’s intermitency is at hour 60. Upon success, the experiment on the NCEDC and SCEDC went smoothly, maximizing the number of jobs until completion.

Community models are typically open-source and publicly available, designed to incorporate the best available data and understanding, and intended to serve as common foundations for scientific research, education, and practical applications (e.g., Plesch et al. ²⁰⁸, Shaw et al. ²⁴²). The generation of community models is a focus of various community science organizations such as the Statewide California Earthquake Center (SCEC) and the Cascadia Region Earthquake Science Center (CRESCENT) ^{1,162}.

To enable FAIR (Findable, Accessible, Interoperable, and Reusable) access to community models ³⁰⁰ and facilitate their use across a wide range of applications, these models are often distributed in multiple formats—such as structured text files, binary volumes, databases, or web-based APIs—to accommodate diverse user needs and computing environments. However, the complexity and size of these datasets—particularly 3D and 4D models of Earth’s interior or fault systems—can present significant technical barriers for many users ²⁴⁹. There is a growing need for software tools that allow users to interactively visualize, query, and subset models before downloading or integrating them into workflows.

These tools must balance performance with accessibility and support interoperability with widely used scientific programming languages (e.g., Python) and data standards. Ultimately, the effective use of community models depends not only on their scientific rigor but also on the availability of user-friendly software that lowers the barrier to entry.

The cloud is particularly well suited for the hosting, visualizing, and disseminating community models because it provides centralized, flexible infrastructure that meets the needs of distributed scientific teams⁸⁴. Community models are often large, dynamic datasets, and are accessed by users across multiple institutions. These characteristics make local storage solutions inefficient or inaccessible. Cloud storage enables elastic scaling, allowing both storage and computing power to dynamically adapt to the community model's growth, without the managerial burdens associated with on-premises infrastructure. It also facilitates consistent versioning, access control, and metadata management, which are required for transparency and reproducibility. In addition, cloud platforms integrate seamlessly with modern computational tools and workflows, enabling users to analyze and visualize models directly in the cloud without transferring large datasets. As such, the cloud is a natural fit for managing community-driven, data-intensive geoscientific resources.

As an example of such a tool in the seismological community, the CRESCENT Community Velocity Model (CVM) Viewer and Repository²⁰ is a cloud-based platform for storing, distributing, analyzing, and visualizing seismic wave velocity models of the Earth. It combines Python-based tools with a geospatial web interface to enable real-time, interactive exploration of datasets. By adhering to widely accepted metadata and file format standards, the platform ensures that datasets remain consistent, interoperable, and ready for use in both research and education.

The CVM Viewer is a web-based geospatial visualization tool built with Python, FastAPI, and CesiumJS⁵³. It allows users to explore CVM datasets interactively through a cloud-hosted 3D map. In addition to visualizing the spatial extent of seismic velocity models, the viewer displays known faults and earthquake hypocenters. Users can toggle terrain layers, adjust model boundaries, and navigate using rotation, zoom, and pan controls. Visualization tools include horizontal slices, vertical cross-sections, and depth profiles, offering intuitive ways to investigate subsurface structures.

The CVM Repository hosts multiple user-submitted seismic velocity models that have undergone

peer review and are published. Models are stored in netCDF-4 Classic and HDF5 formats on AWS S3, organized hierarchically to separate 3D model volumes and associated surface data. Automated compliance checks are performed before storage to ensure alignment with community metadata standards. The backend, built with `xarray` and `h5py`, handles data queries and retrieves model subsets based on user-defined geographic and depth ranges.

Users can extract horizontal slices, cross-sections, or full 3D volumes in various supported formats. These extraction and conversion tools are deployed using AWS Lambda, enabling efficient access to large netCDF or HDF5 files and seamless format conversion. Throughout these operations, geospatial metadata is preserved to ensure compatibility and compliance with metadata standards. The entire system is deployed using AWS Fargate, a serverless container platform that automatically scales computing resources in response to user demand.

In addition to the CVM Viewer, CRESCENT is developing a suite of complementary cloud-based tools to support the broader earthquake science community. The CRESCENT Community Fault Model Viewer²¹ enables interactive visualization and dissemination of fault geometries and associated metadata. Other tools currently under development include platforms for storing, analyzing, and distributing paleoseismic data and seismicity catalogs, all designed with an emphasis on scalability, accessibility, and adherence to community data standards. These emerging tools demonstrate how cloud infrastructure makes data and computational resources more accessible and promotes collaboration across disciplines and institutions.

7.7 Education and Training Initiatives

Open-source software and interactive cloud-based computing environments have transformed seismological research and education by providing free, accessible data analysis and modeling tools. Platforms like Binder, Google Colab, and institutional JupyterHub instances enable researchers and students to run open-source Jupyter Notebooks without requiring local installation, significantly lowering entry barriers. `Seismo-live`¹²⁷ is one example, offering a library of seismology-focused Jupyter Notebooks that can be executed directly in a web browser using Binder. Similarly, Google Colab provides a free cloud-based notebook environment with pre-installed libraries, enabling students to analyze seismic

datasets and run numerical simulations from any device. Google Colab is often linked in seismological software repositories to provide tutorials on free cloud services (e.g., in SeisBench³⁰⁷). By leveraging these free Jupyter-based platforms, the seismology community ensures that computational tools are widely available, fostering open science, reproducibility, and equitable access to high-performance research workflows.

Research projects may also lead to developing cloud-based workflows, and researchers may choose to provide guides for cloud usage. For example, Krauss et al.¹²⁴ utilized the Azure platform and compared pre-trained Machine Learning models and template matching for constructing an earthquake catalog offshore. Their work also provided informative, educational guidance for individual researchers in code development and containerization, cloud infrastructure, job design, and performance analysis.

In higher education, instructors leverage cloud platforms to create interactive, scalable learning environments for seismology and data science. Universities may deploy cloud-like infrastructure with virtual machines for classroom instruction, enabling students to access the course with affordable devices (e.g., tablets, laptops, or even phones) and run code and homework on cloud instances. Such cloud integrations enhance accessibility, enabling students to work with real-world data and modeling problems in an educational setting.

Hands-on training workshops are essential for advancing computational seismology skills, as they immerse participants in using modern software and large-scale computing resources. Recent community initiatives, such as the NSF-funded SCOPED project, have organized multimodal workshops to teach researchers and students how to use research-grade seismological software on cloud platforms⁵⁹. Engaging the community in multiple ways with cloud infrastructure has been beneficial: from running simple workflows on a provided cloud-based JupyterHub (e.g., the GeoLab workshop) to deploying an EC2 instance on their own (e.g., in HPS). Recent workshops have been dedicated to training several hundred participants, primarily graduate students, postdoctoral researchers, and research scientists.

7.8 Discussion and Outlook

7.8.1 Software as a Service

Cloud infrastructure lets seismologists rent powerful computing services only when needed, but those virtual servers still start life as “bare-bones” operating systems. Researchers must rebuild their software stack — libraries, compilers, scripts – every time they launch a new instance. To simplify this process, the common practice is to replicate working environments facilitated by environment management software, such as a lightweight solution of the package manager Anaconda (<https://anaconda.org>), or utilize a fully self-contained option such as Docker¹⁶³ and Singularity¹²⁹. Cloud platforms also provide ready-to-use virtual images that cater to general or geophysics-specific needs, often at a small additional cost.

A more user-friendly approach is Software-as-a-Service (SaaS). It is a delivery model that enables users to run scientific software easily while interacting through a web form or API. The backend software can be cloud-optimized, and computing resources are provisioned upon users’ request. Such a model enables users to access the software in a serverless setting without tedious configuration, while being elastic and cost-efficient for service providers. For example, Chen et al.³⁷ proposed a web application that allows users to submit requests to generate synthetic seismograms. The service receives requests along with source parameter settings and initiates the 3D elastic wave equation solver on the backend. Researchers of interest may utilize this service through a direct and convenient web interface, receiving synthetic seismograms without requiring any software configuration. The trade-off is flexibility—custom methods or novel algorithms still require direct access to code and data, which SaaS platforms may not expose.

7.8.2 The Self-imposed Open and Reproducible Science

Seismology already enjoys a culture of open data and open-source software—community archives expose waveforms through standard FDSN web services, and libraries such as ObsPy²⁷ and SeisBench³⁰⁷ make analysis scripts widely shareable. Yet, the full research replication requires running the full-stack workflow on any machine to obtain similar research results.

Cloud computing has become a critical enabler of reproducible research by forcing workflows to be explicit and portable. In traditional observational seismology, workflows often rely on trial and error, such as manually selecting bandpass filters based on domain expertise and visual data inspection.

While this approach benefits from expert judgment, it poses significant challenges to reproducibility — other researchers may struggle to replicate results if they do not use the same parameters or follow the same steps. Cloud computing, however, requires the creation of well-defined, version-controlled, and containerized workflows that are portable and executable in consistent environments. This shift toward standardized research workflows facilitates reproducibility by ensuring that the exact conditions under which the research was conducted can be easily replicated on different systems or by different researchers. This self-imposed reproducibility fosters a more rigorous and transparent scientific process, crucial in modern seismology, where big-data applications increasingly dominate.

7.8.3 The cost of the cloud

Most cloud service providers employ a pay-as-you-go pricing model, where users are only charged for using any related cloud resource. Because commercial cloud services are not directly sponsored or subsidized by government funds, their cost is priced by hardware rental fees, power and facility fees, and commercial profit, but are comparable to the “fully burdened” HPC costs after removing government subsidies. Researchers can estimate the cost of their workload to the first order by timing the duration of time that virtual machines run and the time that data is stored in cloud storage. For example, we summarize the spending of Clements and Denolle⁴⁵ based on AWS EC2 and S3 pricing policy: 1) downloading and uploading 50 TB of NCEDC data (~ 40 USD), 2) performing the single-station noise correlations (< 50 USD), and 3) storing all data over one week (~ 40 USD/week). Such pricing model holds cost advantages: 1) cloud resources are accessible to almost everyone, whilst on-premise equipment makes one-time spending unnecessary, 2) maintenance is performed by cloud service providers instead of full-time institutional IT employees, and 3) spending is better quantified and monitored through the billing statistics and may help future budgeting¹⁹⁵.

However, the pricing model can be complex when chargeable usages are vaguely defined for some services and resources. Consequently, budgeting for cloud infrastructure beyond the basics is often more complicated. For example, in the cloud-native NoisePy test (see Section 7.5.1), we utilize AWS S3 to save pre-stacked correlation functions before stacking. Besides the ephemeral storage cost from pre-stacked correlation functions, S3 write (PUT) and read (GET) operations also come at a flat rate

(usually several USDs per thousand requests). Despite being minimal, such cost shall not be omitted when performing a large-N and large-T ambient noise interferometry study with potentially millions of such operations. Moreover, data bytes traveling across cloud regions will incur a perceivable egress cost, unless waived under specific agreements (e.g., the Open Data Sponsorship Program, which supports the SCEDC S3 archive presented by Yu et al. ³²²). Additional charges may also be applicable for on-demand servers. For instance, I/O operations with the standard pricing model are not free for DocumentDB clusters. With this type of cluster, spending may be significant when I/O usage is extensive (e.g., phase picks, insertions, and metadata queries that scale with the job, see Section 7.5.2). Such scaling terms should be identified and optimized to avoid unexpected spending in a cloud-native workflow. Despite these challenges, our experience has been fortunate to overbudget, strategize to optimize, and conclude with a lower overall cost.

Explicitly comparing cloud compute costs to the costs for on-premise HPC is challenging, as numerous factors need to be taken into account for on-premise HPC, such as procurement cost, energy, facilities, and labor. In addition, such costs will vary strongly on a country-by-country basis. However, for a rough estimate, we compare the costs for AWS Fargate service, the service used for the phase picking workflow above (Section 7.5.2), to estimates from different universities. A CPU core hour (SPOT instance as in our workflow) on AWS costs around 0.012 USD ($\sim$ 0.010 EUR). The costs charged (for internal users) by different universities vary between 0.008 EUR and 0.012 EUR⁴. We note that the charge for main memory on the university deployments was overall slightly higher than on AWS Fargate; however, the cost is dominated by the cost per CPU core. While large-scale users should consider a more precise, task-specific assessment of costs, taking into account local conditions and the expected long-term resource utilization, these numbers illustrate that for workflows as the ones presented, the cost of on-premise and cloud resources is in a similar range. In all instances of researchers using institutional computing centers or cloud services, federal research grants can support these costs (e.g., as was the case for the numerical experiment shown here through a National Science Foundation CloudBank grant¹⁹⁵).

In summary, researchers in seismology will benefit cloud resources if 1) they need on-demand resources as in early warning or rapid post-event response without provisioning them over long time-

⁴<https://hpc.rtu.lv/hpc/hpc-services/price-list/?lang=en>, <https://hpc.ut.ee/pricing/calculate-costs>, <https://hpc-docs.uni.lu/policies/usage-charging/> (last access August 11, 2025 for all)

frames, 2) many researchers are working on the same data (e.g., shared open-access data archive or post-earthquake response archive), and 3) the data archival is supported by cloud providers (e.g., AWS open data), non-profits (e.g., RadiantEarth <https://radiant.earth/>), or funded by governments (e.g., National Science Foundation Seismological Facility for the Advancement of Geoscience (SAGE) facility, now operated by the EarthScope Consortium) and on the order of peta-scale. Cloud services may be too costly for researchers who work directly with TBs of data, whose analyses are not time-constrained, who can guarantee a stable utilization of on-premise resources, and who mostly want to share that data within their institutional research groups (i.e., a limited number of users).

7.8.4 Disadvantages of Cloud Computing for Science

No medium offers true permanence—disk drives typically require replacement after 5–8 years—and cloud object stores are no exception. The difference is economic rather than technical: keeping a petabyte of data in a frequent-access state becomes an annual operating expense that can outstrip the one-off capital cost of a local tape library, especially once data-egress and retrieval fees are factored in. Although a common perception is that commercial clouds are less reliable than institutional computing centers, the underlying hardware is identical. The fundamental trade-off, therefore, is not reliability but the ongoing cost model and the penalties for frequent, bulk access to archival datasets.

Moreover, because cloud uses shared but logically isolated hardware, run-to-run performance may fluctuate as neighbouring users compete for network or disk bandwidth, whereas a dedicated on-prem workstation delivers near-identical wall times each day. Second, tightly coupled jobs still run more efficiently on institutional clusters wired with low-latency node communication; comparable services do exist on the cloud, but the hourly premium usually outweighs any elasticity benefit. Finally, the funding model feels unfamiliar: instead of drawing on a pre-paid allocation of CPU-hours, researchers receive a transparent bill that details costs, which can be unsettling when project budgets are built around fixed grant lines.

7.9 Conclusion

Cloud computing and storage offer a viable solution for analyzing petabyte-scale seismic archives, leveraging the concept that data is transferred only once but distributed to many users. Cloud-optimized and containerized workflows can be spun up on demand at a relatively affordable cost for the size of the task. Cloud computing and storage solve the problems of slow campus Ethernet networks and shared HPC queues, enabling peta-scale analysis completed in hours to days, instead of months. However, building a cloud-ready pipeline remains a significant undertaking for most scientists. Containerizing legacy code, wiring up object storage I/O, and automating provenance all require skills that fall between classical research and production software. Research software engineers remain essential: they translate scientific intent into robust, version-controlled artifacts, add automated tests, and keep pace with the rapid evolution of cloud services. Once that up-front investment is made, every run is self-documenting and trivially repeatable, lowering the long-term maintenance burden. Prototyping for cloud systems on local servers is beneficial (e.g., minimizing cost) and possible (e.g., developing reproducible software and environments), for example, using tools such as MinIO object storage systems and MongoDB databases. Researchers may focus on workflow reproducibility, open-sourcing software, and minimizing job requirements to increase speed and lower computing and environmental costs.

Our performance benchmarks illustrate the payoff. In our ambient-noise test case, 1.6 TB of correlation functions—6.2 million files—were generated in eleven hours for about 250 USD, all without touching a single on-premise disk. Likewise, the QuakeScope catalog builder scanned roughly 1.3 PB of global datasets in three days, limited only by computing quotas, and likely discovered 10 times more earthquakes than previously reported. Automatic retries on pre-emptible instances kept utilisation high and human intervention low, demonstrating that today’s managed services can rival dedicated HPC for embarrassingly parallel seismology at comparable cost.

Looking ahead, both DAS and the rapid proliferation of low-cost IoT seismometers will significantly increase data volumes and latency requirements, far exceeding today’s norms. In the earthquake early warning architecture, hundreds to thousands of edge devices can run lightweight pickers locally, then stream only compact, parametric data to the cloud. There, serverless functions fan-in those messages, trigger association and localization, and broadcast alerts—often within a few seconds of the origin

time. The rapid scaling of cloud resources is particularly suitable for handling rare and extreme events. Finally, Cloud providers' per-region carbon-intensity dashboards (e.g., AWS's Sustainability Pillar, Google's Carbon-Free-Energy scores, Microsoft's 2025 renewables target) can inform users to choose cloud computing regions with a lower carbon footprint without sacrificing latency and performance.

However, there are caveats about using the cloud for seismological research. Cloud services are expensive for direct prototyping and experimenting with orchestrating autoscaling. Given the investment in software development, it is also not necessarily advantageous to use the cloud for analyzing datasets on the order of 100 TBs or less. Furthermore, the cost of cloud storage is significant for individual researchers, except when benefiting from free open data programs (e.g., AWS open data), and when national facilities that typically invest heavily in on-premise servers find cost benefits in migrating services to cloud systems (e.g., EarthScope). Finally, technological advancements in commercial clouds are needed to compete with the tightly coupled nodes of conventional HPC centers.

Realizing that vision will take community effort—shared, containerized code, FAIR data in cloud-optimised formats, and collaborations between domain scientists and research software engineers. Computing skills and software best practices are constantly improving, especially with the rise of open-source programming languages and training materials (e.g., Community⁵²).

Publicly funded research networks such as ESnet (United States), GÉANT (Europe), and CSTNET (China), paired with high-speed transfer tools like Globus⁸ and GridFTP, enable researchers to move multi-terabyte to petabyte datasets at tens to hundreds of gigabits per second, with record demonstrations reaching 1 PB/day. These dedicated optical backbones interconnect supercomputing centers, data archives, and universities, providing low-latency, high-reliability transfers at no direct cost to scientists, critical for sharing seismic, satellite, and climate model data. In parallel, the compute-close-to-data paradigm allows researchers to run tens of thousands of jobs directly adjacent to petascale datasets on public HPC platforms (e.g., United States Department of Energy facilities, Partnership for Advanced Computing in Europe (PRACE), China National Grid (CNGrid)) or commercial clouds, minimizing transfer time entirely. Public HPC resources offer this capacity for free to approved projects but remain gated by competitive access and national affiliation, while commercial clouds are globally accessible yet cost-sensitive. A globally equitable ecosystem would combine open research datasets with subsidized

compute near data, ensuring that any researcher, anywhere, could run massive workflows without prohibitive costs or technical barriers.

To conclude, the heavy lift of the peta-scale era for seismological data analysis is within reach. Embracing cloud and other scalable computing solutions transforms this burden into a catalyst. It allows researchers to interrogate Earth processes at resolutions and timescales that were previously out of reach, opening new frontiers of discovery. It also permits researchers to focus on fundamental physical methods, rather than being limited by a given observational period and spatial extent. Petabyte-scale seismology is now practical.

Acknowledgment

This work is supported by the Seismic Computational Platform for Empowering Discovery (SCOPED) project under the National Science Foundation (award numbers OAC-2103701 (UW), OAC-2103494 (UT)). The Schmidt Futures Foundation also supported the development of NoisePy at the University of Washington’s Scientific Software Engineering Center. The EarthScope Consortium, through a Pass-Through Entity (PTE) Federal award no 2310069, partially supported this work. EarthScope data were accessed from the NSF SAGE data archive operated by the EarthScope Consortium (award number 1724509). The computing resources presented in this paper were obtained using CloudBank¹⁹⁵, which is supported by the National Science Foundation (award number CNS-1925001). The Harvard Data Science Initiative supported the development of the Julia Cloud workflow, developed by T. Clements and J. Schmitt, NSF EAR-1850015 award. JM has been funded by the European Union under the grant agreement n°101104996 (“DECODE”). We are grateful for discussions with Manochehr Bahavar and Loïc Bachelot surrounding visualization on the cloud. We thank Dr. Andreas Fichtner and another anonymous reviewer for their constructive comments. AI tools were used for grammar and spelling checks.

Code and Data Availability

The SCEDC and NCEDC data used in this study are publicly available through the AWS Open Data Sponsorship Program (<https://registry.opendata.aws/southern-california-earthquakes/>

for SCEDC and <https://registry.opendata.aws/northern-california-earthquakes/> for NCEDC). The EarthScope Consortium data could be accessed through the EarthScope FDSN web services. Softwares used in Section 7.5 are available at <https://github.com/noisepy> and <https://github.com/seisSCOPED/quakeScope/>.

8

Conclusion

Modern seismology is entering a new era of data-driven discovery, shaped by advances in machine learning, novel observational technologies such as fiber optic sensing, and the emergence of cloud computing and storage as an advanced computing infrastructure. Together, these developments create unprecedented opportunities to advance earthquake science from local networks to global archives, environments from mountain ranges to the offshore, and deployment contexts from centralized computing clusters to the far edge. This dissertation suggests that our community is ready to leverage these innovations to scale up earthquake monitoring systems and transform data-driven discovery in earthquake science.

In Chapter 2, I presented a curated seismic dataset drawn from 21 years of Pacific Northwest Seismic

Network (PNSN) recordings, compatible with the SeisBench³⁰⁷ framework. Deep learning phase pickers fine-tuned on this dataset show improved performance over pretrained models on the events from the Pacific Northwest. Overall, this high quality dataset provides a platform for training and benchmarking machine learning models on a wide range of seismological tasks. Looking forward, this dataset will be a resource for the community for developing both task-specific and general-purpose foundational models to better characterize seismic and aseismic phenomena.

In Chapter 3, I introduced a cloud-native data management framework designed to address the storage and processing challenges posed by the massive data volumes from Distributed Acoustic Sensing (DAS) deployments. By leveraging the flexibility and scalability of cloud object storage with cloud-optimized formats such as Zarr and TileDB, this framework delivers efficient data I/O at scale. To evaluate its effectiveness, I applied it to an ambient noise cross-correlation workflow using one month of SeaDAS-North data, processing over 2 million channel-pairs and completing the full ≈ 100 billion cross-correlations in under 24 hours. The viability demonstrated in this study will encourage the community to adopt cloud object storage for hosting and serving large-scale DAS datasets and beyond. Future work will focus on leveraging the ambient noise cross-correlation functions for time-lapse near-surface tomography.

In Chapter 4, I employed a machine learning algorithm for DAS wavefield reconstruction and lossy compression. By training a SHRED model³⁰³ on 2,615 earthquakes recorded by the GCI DAS cable in Cook Inlet, Alaska, I reconstructed the low-frequency, dispersive ocean wave signals, with the high-frequency, impulsive seismic wavefield separated as a residual. With this lightweight model, earthquake arrivals, especially for S-waves, are enhanced and can be easily detected by a simple kurtosis trigger. I further demonstrated that an ensemble phase picker (i.e., ELEP³²³), despite being trained on conventional seismic data, can be readily applied to pick phase arrivals on DAS wavefields. Overall, this work shows that machine learning can be a powerful tool for processing fiber-optic sensing data in the edge computing paradigm.

In Chapter 5, I presented a scalable cloud-native workflow for petabyte-scale event detection and phase picking. By deploying containerized jobs on AWS and reading waveforms directly from cloud-hosted data archives maintained by EarthScope, NCEDC, and SCEDC, I processed approximately 1.3 PB

of continuous recordings from 47,354 stations spanning January 2002 to March 2025. Phase arrivals were identified with PhaseNet³³⁷ implemented through SeisBench³⁰⁷, running up to 1,500 parallel jobs and storing results in a cloud-hosted database. The resulting global phase-pick database is publicly accessible through a web service. I anticipate this database will serve as a foundation for community-scale seismicity studies and that the cloud workflow template developed here can be readily adapted for future large-scale data-mining efforts.

In Chapter 6, I developed EQ-SHRED and EQ-FLOW, a multi-stage machine learning framework for reconstructing earthquake ground motion fields from sparse sensor observations. By training on 4,722 synthetic wavefield simulations using SEM²⁶⁸, EQ-SHRED uses a recurrent encoder with a compressive learning strategy to recover the full surface displacement, velocity, and ground motion intensity (PGV and PGA) fields from only $\sim 0.3\%$ of the surface grid, while simultaneously estimating probabilistic hypocenter locations. A subsequent flow matching generative model (EQ-FLOW) then refines the reconstructed wavefield with short-wavelength spatial complexity. I further demonstrated that hypocenter localization can be achieved within seconds of the event being detected, making the framework applicable for earthquake early warning.

In Chapter 7, I provided a comprehensive review of cloud computing and storage in seismology, synthesizing lessons learned from the workflows developed in Chapters 3 and 5. I surveyed cloud storage solutions, cloud-optimized data formats, and scalable database options, and benchmarked two archetypal big-data tasks: deep learning-based phase picking and ambient noise cross-correlation at scale. I further discussed the opportunities, costs, and challenges of adopting cloud platforms. This review, together with Krauss et al.¹²⁴, will serve as a practical reference for seismologists transitioning their workflows to the cloud.

Despite the progress described in this dissertation, significant challenges and limitations remain. These include the adoption and standardization of new data formats by key stakeholders, e.g., researchers, data providers, and the open-source community. Fiber optic sensing has yet to be widely integrated into seismic network operations, and its compatibility with the existing AQMS system remains uncertain. Looking further ahead, agentic Artificial Intelligence may transform data-driven seismology in ways that extend well beyond current machine learning approaches.

In summary, the contributions in this dissertation span the full data lifecycle: from acquisition and storage to analysis and interpretation, and across deployment environments ranging from the far edge to cloud infrastructure. These contributions reflect a common thread: scalable infrastructure, novel observation systems, and data-driven algorithms are reshaping what is computationally and scientifically possible in earthquake monitoring. I hope the datasets, workflows, and frameworks presented here will contribute to the development of next-generation seismic monitoring systems, which will ultimately advance our ability to characterize seismic hazard and deepen our understanding of earthquake physics.



Acronyms and Abbreviations

- AEC: Alaska Earthquake Center
- ANSS: Advanced National Seismic System
- API: Application Programming Interface
- AQMS: Advanced Quake Monitoring System
- ASDF: Adaptable Seismic Data Format
- AWS: Amazon Web Services
- CISN: California Integrated Seismic Network
- CLI: Command-Line Interface
- CNN: Convolutional Neural Network
- ComCat: Comprehensive Catalog
- CPU: Central Processing Unit
- CRESCENT: Cascadia Region Earthquake Science Center
- CSV: Comma-Separated Values
- DAS: Distributed Acoustic Sensing
- DB: Database
- EC2: Elastic Compute Cloud
- EEW: Earthquake Early Warning
- EGI: European Grid Infrastructure

- ELEP: Ensemble Learning Earthquake Phase Picker
- EQCCT: Earthquake Phase Classification and Correction Transformer
- FAIR: Findable, Accessible, Interoperable, Reusable
- FDSN: Federation of Digital Seismograph Networks
- F-K: Frequency-Wavenumber
- GCP: Google Cloud Platform
- GMPE: Ground Motion Prediction Equation
- GNSS: Global Navigation Satellite System
- GPU: Graphics Processing Unit
- GR: Gutenberg-Richter
- GRU: Gated Recurrent Unit
- HDD: Hard Disk Drive
- HDF5: Hierarchical Data Format version 5
- HEMEW^S: HEterogeneous Materials and Elastic Waves with Source variability
- HPC: High-Performance Computing
- IASP: International Association of Seismology and Physics of the Earth's Interior
- INR: Implicit Neural Representation
- INSTANCE: Italian Seismic Dataset for Machine Learning
- IRIS DMC: Incorporated Research Institutions for Seismology Data Management Center
- ISC: International Seismological Centre
- JSON: JavaScript Object Notation
- KKFL: Kodiak-Kenai Fiber Link
- LSTM: Long Short-Term Memory
- MAE: Mean Absolute Error
- ML: Machine Learning
- MLP: Multilayer Perceptron
- MPI: Message Passing Interface
- MSE: Mean Squared Error

- NAS: Network-Attached Storage
- NCEDC: Northern California Earthquake Data Center
- NCSN: Northern California Seismic Network
- NEIC: National Earthquake Information Center
- NSF: National Science Foundation
- OBS: Ocean-Bottom Seismometer
- ODE: Ordinary Differential Equation
- OOI: Ocean Observatories Initiative
- ORFEUS: Observatories and Research Facilities for European Seismology
- PCA: Principal Component Analysis
- PGA: Peak Ground Acceleration
- PGV: Peak Ground Velocity
- PNSN: Pacific Northwest Seismic Network
- PNW: Pacific Northwest
- POD: Proper Orthogonal Decomposition
- POSIX: Portable Operating System Interface
- PSNR: Peak Signal-to-Noise Ratio
- RAM: Random Access Memory
- RFFN: Random Fourier Feature Network
- RMS: Root Mean Square
- RMSE: Root Mean Square Error
- RNN: Recurrent Neural Network
- S3: Simple Storage Service
- SAGE: Seismological Facility for the Advancement of Geoscience
- SCEDC: Southern California Earthquake Data Center
- SCEC: Southern California Earthquake Center
- SCOPED: Seismic Computational Platform for Empowering Discovery
- SCSN: Southern California Seismic Network

- SEED: Standard for the Exchange of Earthquake Data
- SEM: Spectral Element Method
- SHRED: SHallow REcurrent Decoder
- SIREN: Sinusoidal Representation Network
- SLURM: Simple Linux Utility for Resource Management
- SNR: Signal-to-Noise Ratio
- SSD: Solid State Drive
- STA/LTA: Short-Term Average over Long-Term Average
- STEAD: Stanford Earthquake Dataset
- SVD: Singular Value Decomposition
- TERRA: Terrestrial for Every Rural Region
- URL: Uniform Resource Locator
- USGS: United States Geological Survey
- UTC: Coordinated Universal Time
- VM: Virtual Machine
- WS: Web Service
- XML: eXtensible Markup Language

B

Supplementary Materials for Chapter 2

AQMS event type use by the PNSN	ComCat label	Description
eq	earthquake	earthquake
le		local earthquake
re		regional earthquake
ts		teleseism
se		slow earthquake
lp		long period volcanic earthquake
lf		low-frequency event
ex	explosion	generic chemical blast
px		unconfirmed blast or explosion
sh		refraction/reflection survey shot
su	surface event	surface event
th	thunder	thunder
sn	sonic boom	sonic shockwave
pc	plane crash	plane crash
qb	quarry blast	quarry blast
nt	nuclear explosion	nuclear test
ve	volcanic eruption	volcanic eruption
co	mine collapse	mine/tunnel collapse
df	debris avalanche	debris flow/avalanche

av	snow avalanche	snow/ice avalanche
ls	landslide	landslide
rb	rock burst	rockburst
rs	rockslide	rockslide
bc	building collapse	building collapse/demolition
mi	meteor impact	meteor/comet impact
uk	unknown	unknown type

Table B.1: The event types and labels used in PNSN’s ANSS Quake Monitoring System (AQMS). Several event types are merged into one when being reported to the ComCat.

UTC Time	CISN Event ID	PNSN Event ID	Magnitude
2022-12-20 10:34:24	nc73821036	uw61899256	Mw 6.4
2022-12-20 10:52:39	nc73821106	uw61899266	Ml 3.1
2022-12-20 11:42:54	nc73821226	uw61899301	Ml 3.0
2022-12-20 12:21:43	nc73821346	uw61899311	Ml 3.0
2022-12-20 13:35:06	nc73821486	uw61899326	Ml 3.4
2022-12-20 13:53:19	nc73821516	uw61899331	Ml 3.3
2022-12-20 15:09:05	nc73821636	uw61899336	Mw 4.0
2022-12-20 15:30:01	nc73821656	uw61899346	Ml 3.0
2022-12-20 16:33:41	nc73821761	uw61899381	Ml 3.2
2022-12-20 22:06:34	nc73822026	uw61890602	Mw 4.4
2022-12-21 01:07:33	nc73822146	uw61890647	Ml 3.0
2022-12-21 07:17:15	nc73822341	uw61890697	Ml 3.0
2022-12-21 16:28:16	nc73822556	uw61890767	Ml 3.3
2022-12-22 08:47:13	nc73822961	uw61890997	Ml 3.3
2022-12-22 11:49:55	nc73823036	uw61891007	Mw 3.8
2022-12-24 19:33:44	nc73824236	uw61900186	Mw 4.2
2022-12-25 19:40:29	nc73824826	uw61900336	Md 3.0
2022-12-29 00:16:18	nc73826156	uw61900806	Md 3.0
2023-01-01 18:35:04	nc73827571	uw61901146	Mw 5.4
2023-01-06 12:27:19	nc73829331	uw61901521	Ml 3.3

Table B.2: The events selected from the 20 December 2022 Northern California earthquake sequence that are included in the data set. Source origin time, event ID, and magnitude are reported by the California Integrated Seismic Network (CISN), with corresponding PNSN event ID.

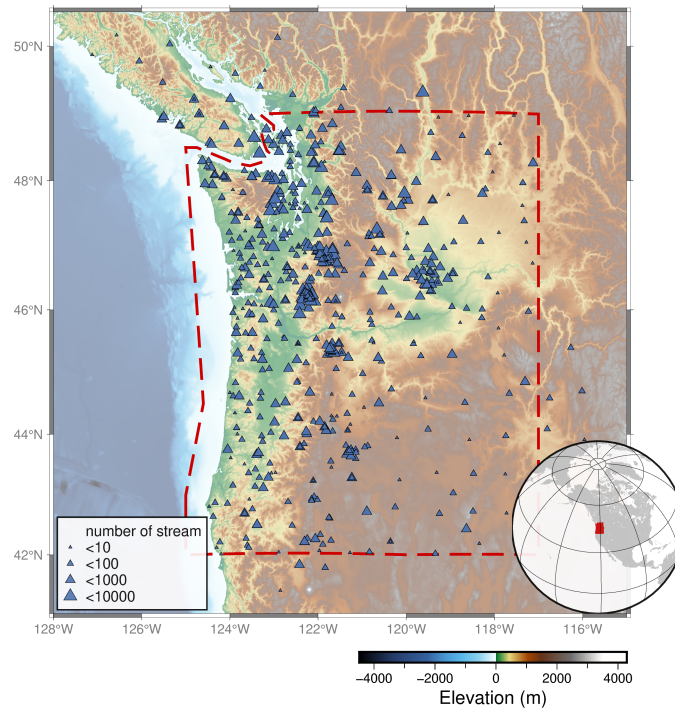


Figure B.1: Number of streams from the ComCat event catalog per station. The red dashed polygon denotes the authoritative region boundaries of PNSN.

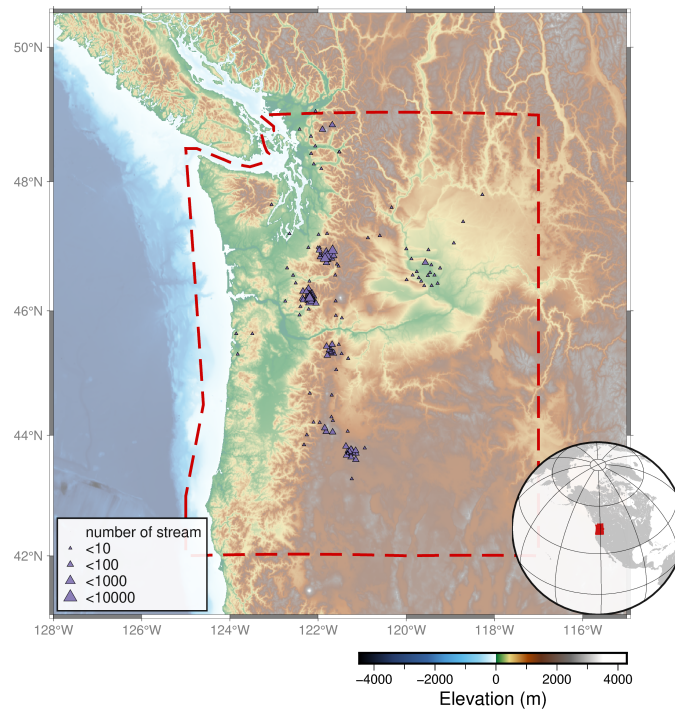


Figure B.2: Number of streams from the Exotic event catalog per station. The red dashed polygon denotes the authoritative region boundaries of PNSN. Note that 96% of the exotic events are surface events.

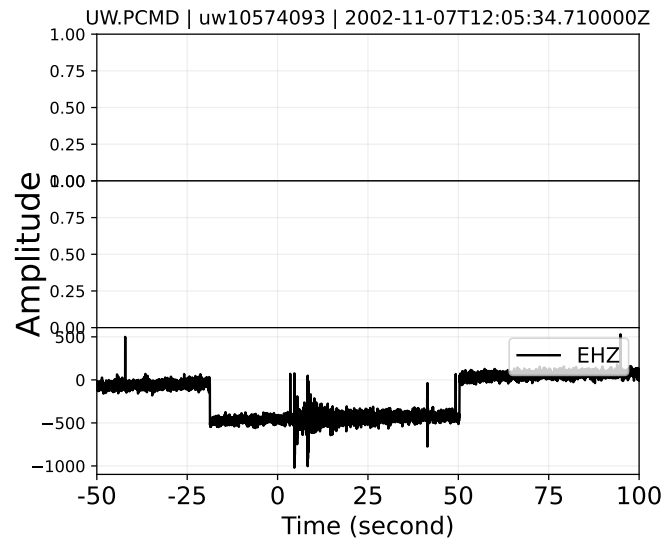


Figure B.3: Example of a stream with a step offset from short-period EH channel. FDSN network and station code, ComCat event ID, and source origin time are labeled on the top.

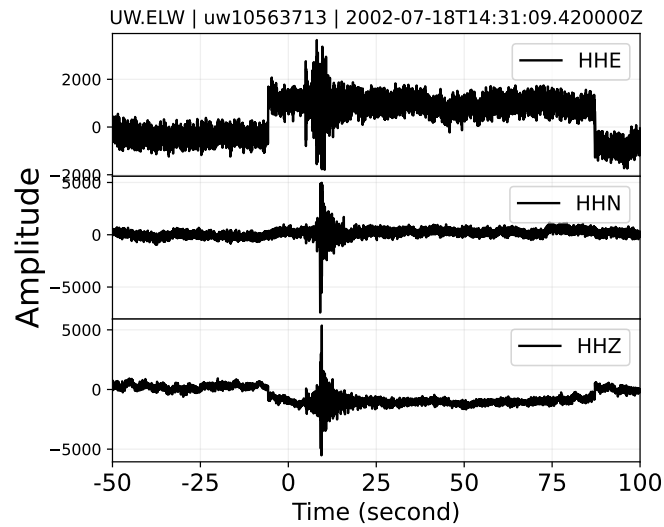


Figure B.4: Example of a stream with a step offset from broad-band HH channel. FDSN network and station code, ComCat event ID, and source origin time are labeled on the top.

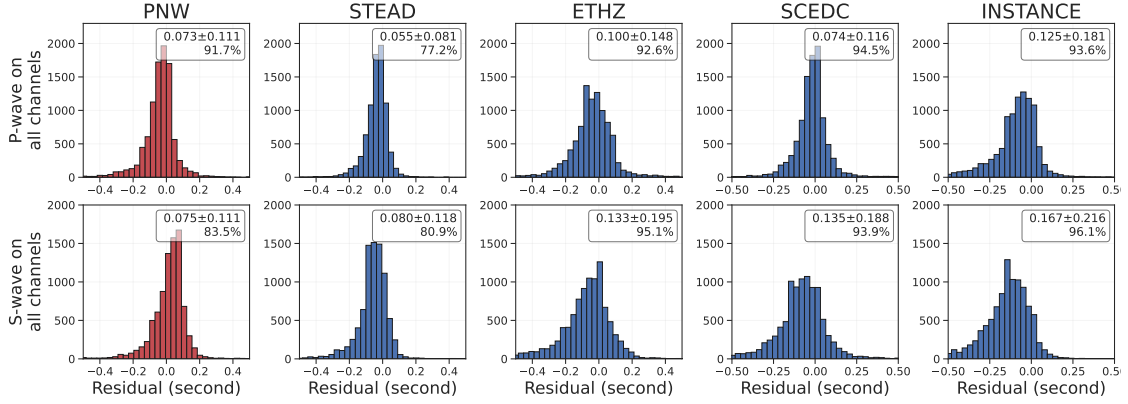


Figure B.5: Histogram of P- and S-wave picking residuals ($t_{ML} - t_{PNSN}$) from the benchmark test data set of strong motion channels. The number in the upper right corner of each subplot shows the mean absolute error (MAE), the root-mean-square error (RMSE) as the uncertainties to the MAE, and the picking completeness in percentage with respect to the ground truth. The PNW-retrained Earthquake Transformer outperforms than other four pre-trained models from SeisBench³⁰⁷ in both picking accuracy and detecting completeness.

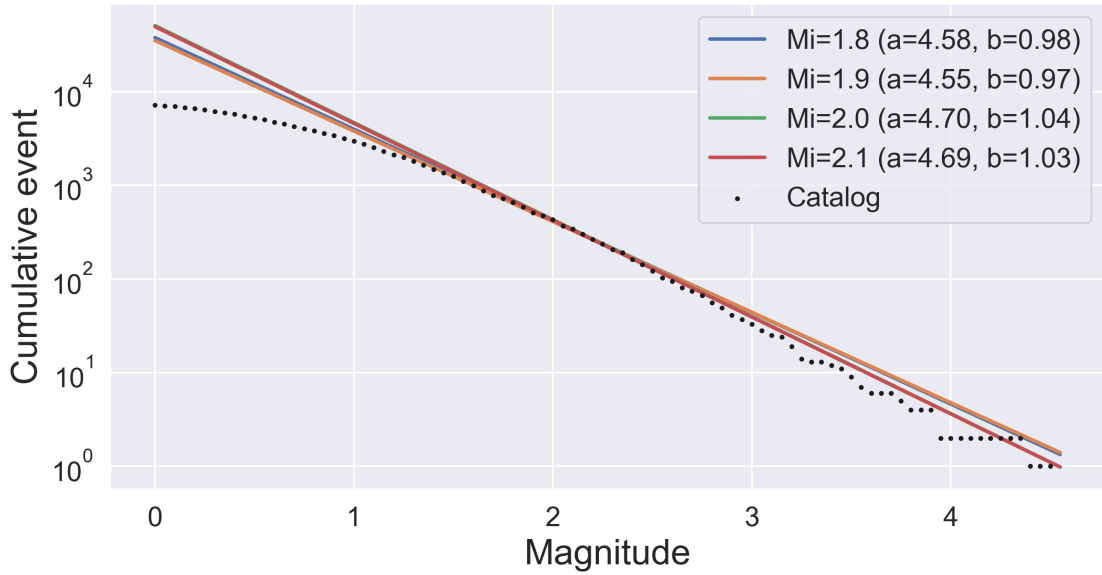


Figure B.6: Fitting of Gutenberg-Richter (GR) power law distribution of magnitudes (solid lines) with 3 years of earthquake events cataloged by PNSN (black dots). The plot indicates that the minimum magnitude of completeness is around 2.

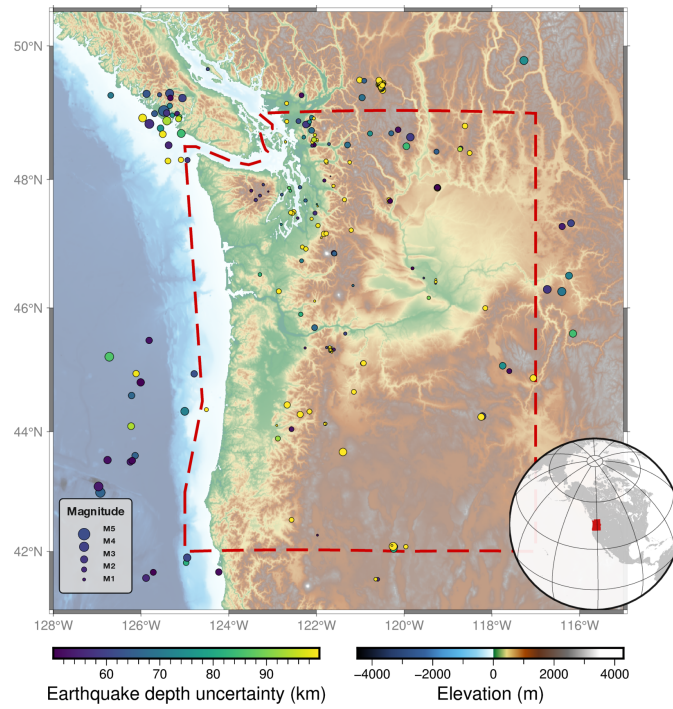


Figure B.7: Location of the events with location horizontal uncertainty larger than 20 km. The red polygon denotes the authoritative region boundaries of PNSN.

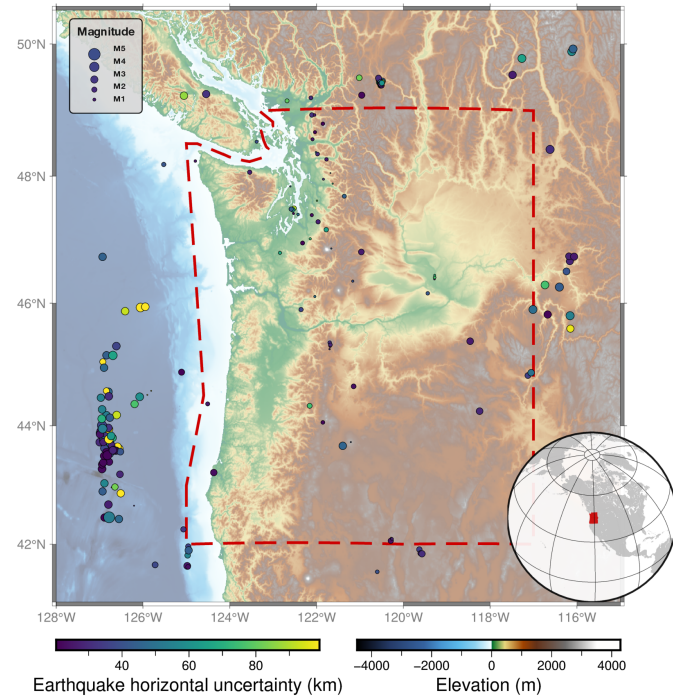


Figure B.8: Location of the events with location horizontal uncertainty larger than 20 km. The red polygon denotes the authoritative region boundaries of PNSN.

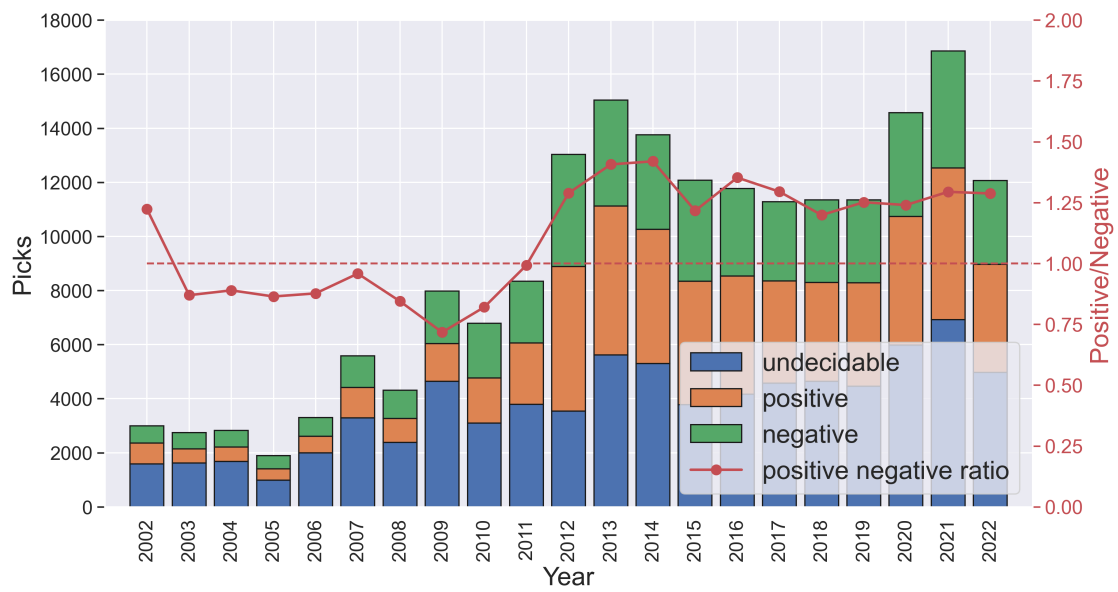


Figure B.9: Number of picked P-wave polarity as a function of time. The red line marks the positive-negative ratio of the P-wave polarities.



Supplementary Materials for Chapter 4

Configurations			Metrics		
N_{LSTM}	N_{latent}	N_{param}	PSNR $\uparrow$	MSE $\downarrow$	Time (sec) $\downarrow$
1	96	664k	25.52	0.0917	15.2
1	150	788k	25.80	0.0859	31.4
1	196	912k	25.90	0.0840	37.5
2	150	969k	26.31	0.0764	51.7
2	196	1221k	26.46	0.0738	55.9
3	150	1151k	26.25	0.0775	72.2

Table C.1: Performance comparison of SHRED under different configurations on a 10-minute long KKFL-S test recording. N_{LSTM} : the number of the LSTM layer. N_{latent} : the dimension of the latent space. N_{hidden} : the number of neurons in the hidden layer.

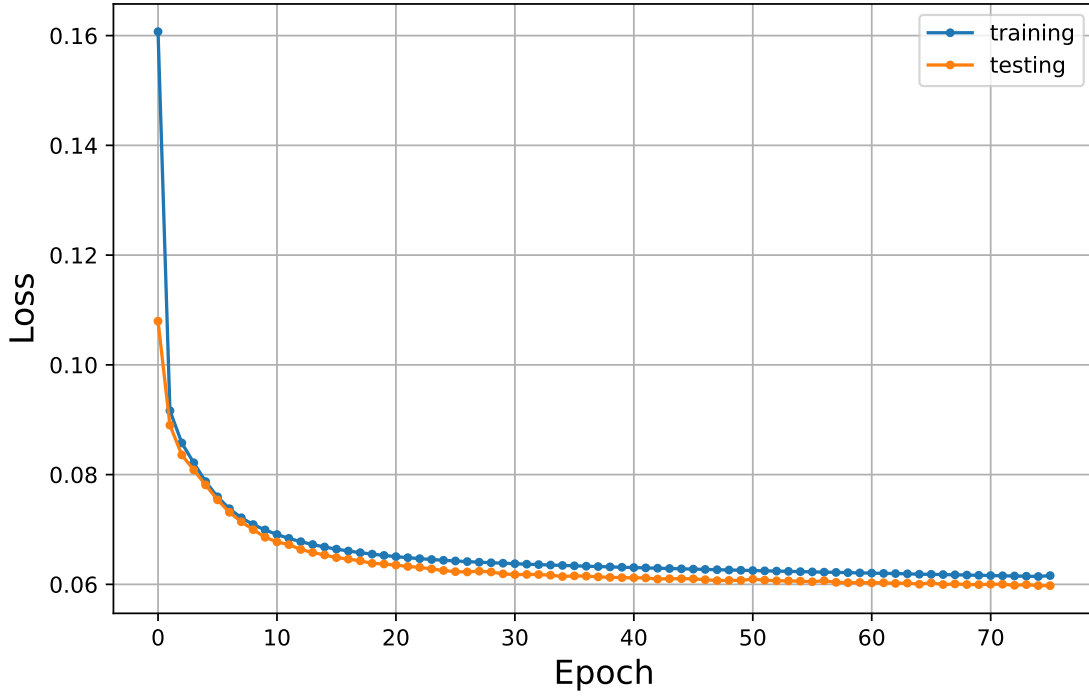


Figure C.1: Loss function during the SHallow REcurrent Decoder (SHRED) model training process. Mean-Squared-Error (MSE) is used to measure the misfit between the reconstructed and the raw data.

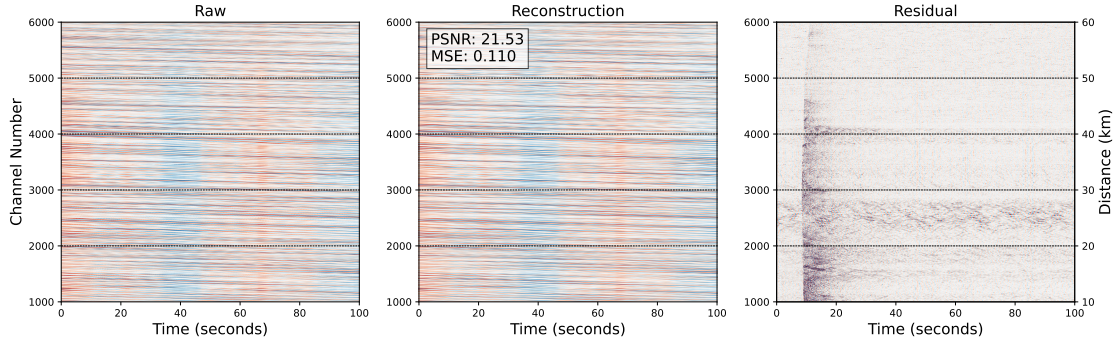


Figure C.2: A 100 s long example of the raw, reconstructed, and residual data from the Cook Inlet DAS experiment on the KKFL-S cable sampling at 25 Hz. Figures show the arrivals from an M1.9 earthquake near Anchor Point, Alaska, on 22 July 2023 (USGS event ID ak0239btba15). The amplitude coefficient b of this event is estimated to be 0.704. The same color scale is used for raw and reconstruction data, while the residual is boosted by a factor of two.

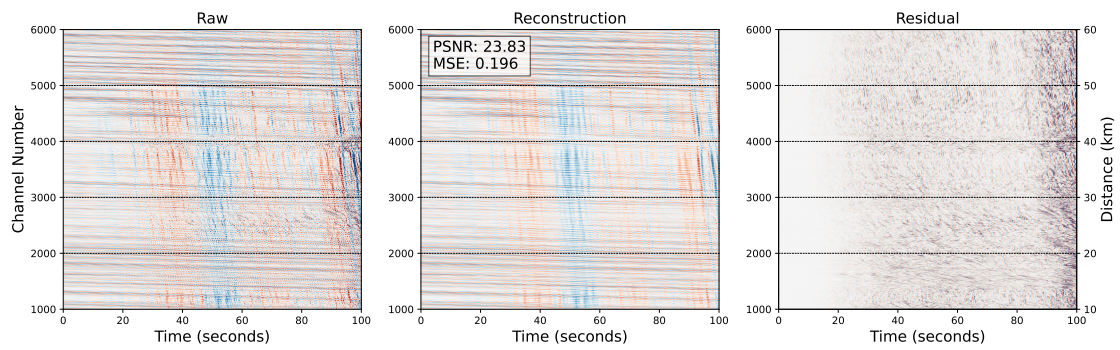


Figure C.3: A 100 s long example of the raw, reconstructed, and residual data from the Cook Inlet DAS experiment on the KKFL-S cable sampling at 25 Hz. Figures show the arrivals from an $M_w 7.2$ earthquake near Sand Point, Alaska, on 16 July 2023 (USGS event ID at00rxvmw1). The amplitude coefficient b of this event is estimated to be 0.643. The same color scale is used for raw and reconstruction data, while the residual is boosted by a factor of two.

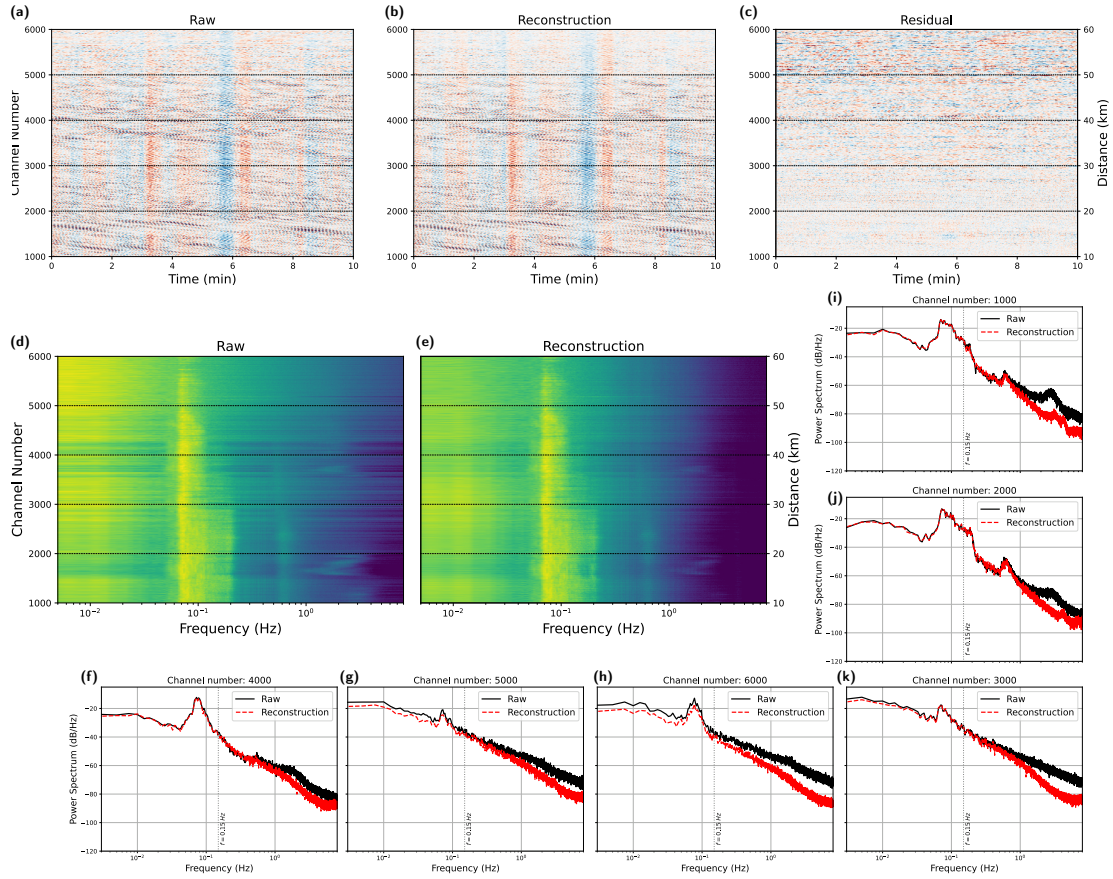


Figure C.4: (a-c) A 10-minute long example of the raw, reconstructed, and residual data from the Cook Inlet DAS experiment on the TERRA cable sampling at 25 Hz. The selected window does not contain earthquakes according to the USGS catalog. The same color scale is used for raw and reconstruction data, while the residual is boosted by a factor of two. (d-e) The power spectrum of the raw data (black solid line) and reconstruction (red solid line) across the cables, with six selected channels shown in (i-k).

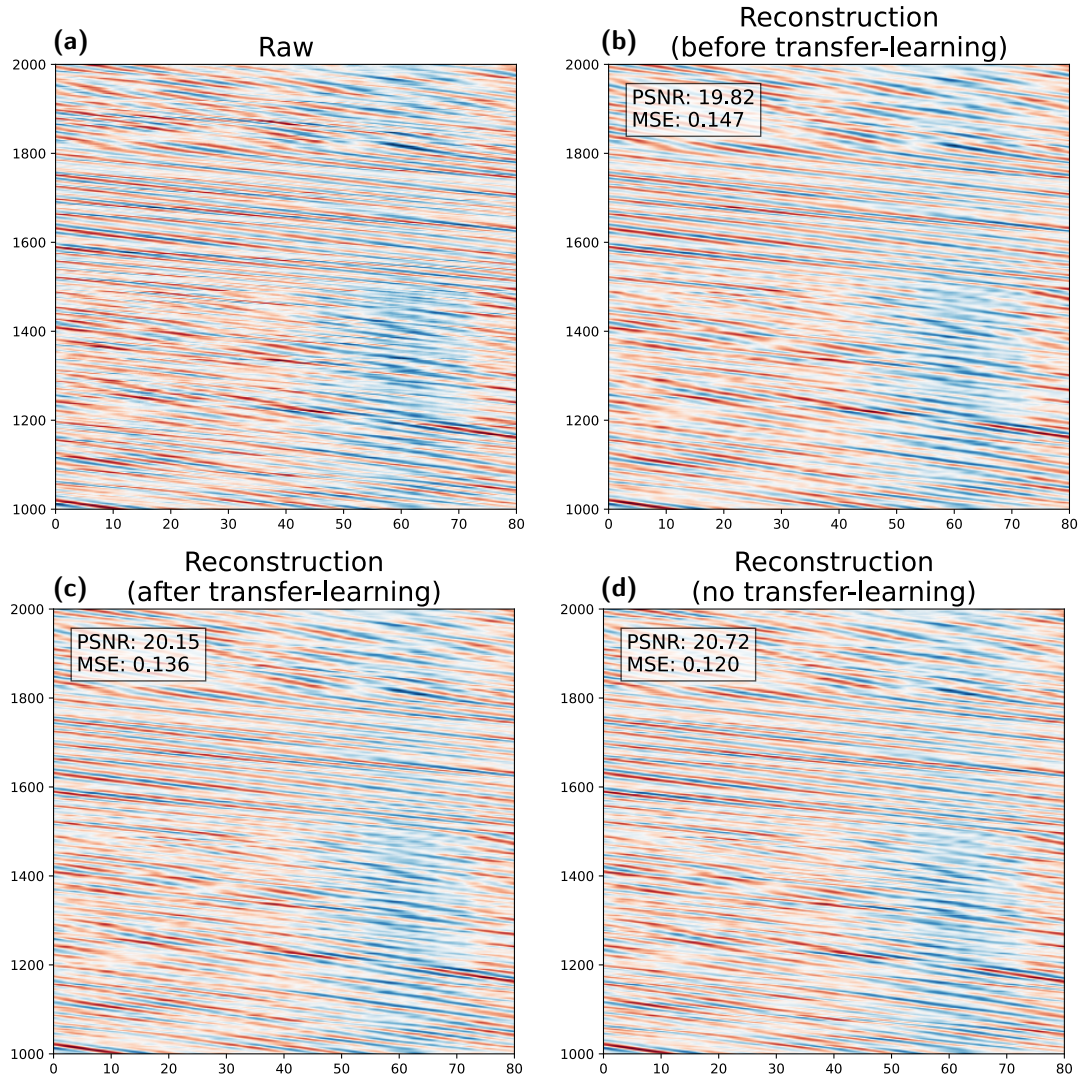


Figure C.5: Generalizability test of the SHRED model. (a) An 80 s long example of the raw data from the Cook Inlet DAS experiment on the TERRA cable sampling at 25 Hz. Figures show the reconstruction results of (b) the model trained by data from KKFL-S cable, (c) the model in (b) but transfer-learned with data from TERRA cable, and (d) the model trained by data from TERRA cable.

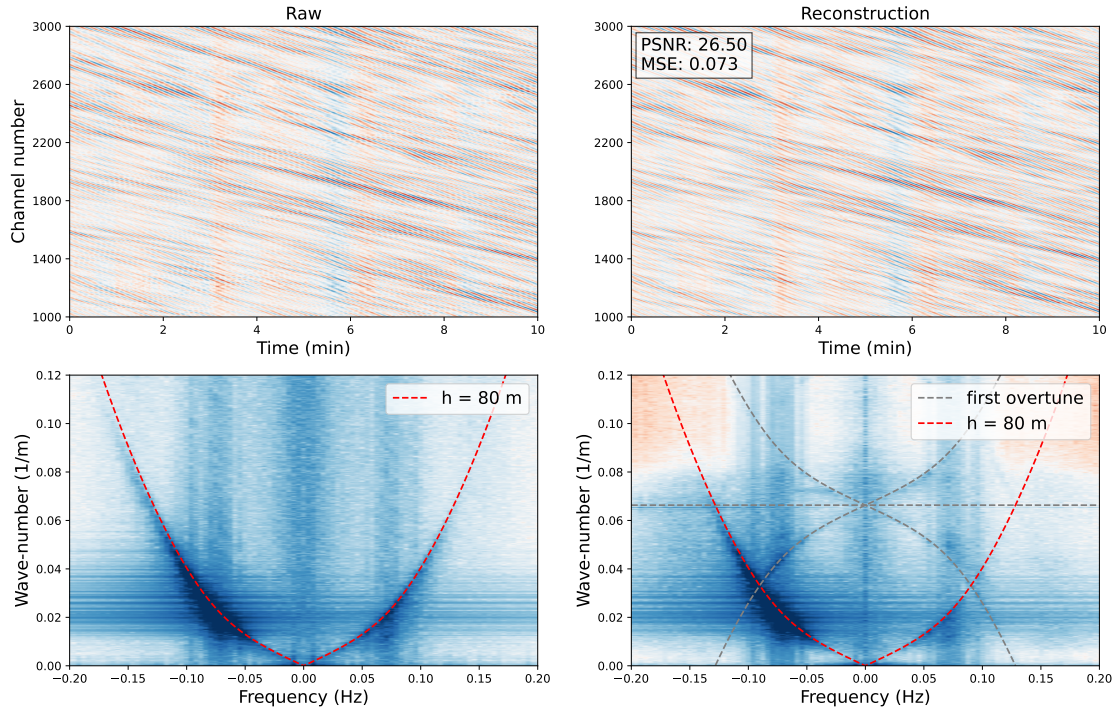


Figure C.6: (a-b) A 10-minute example of raw and reconstruction data from the Cook Inlet DAS experiment on the KKFL-S cable sampling at 25 Hz. The SHallow REcurrent Decoder (SHRED) model is trained using 101 out of 1000 decimated channels. Data from a segment of 2000 channels are shown. The corresponding data in the frequency-wavenumber (F-K) domain is shown in (c) and (d), respectively. Dispersion relation for ocean wave under an intermediate water depth setting ($h = 80$ m) is plotted as red dashed lines. The first overtone is modeled as dashed gray lines.

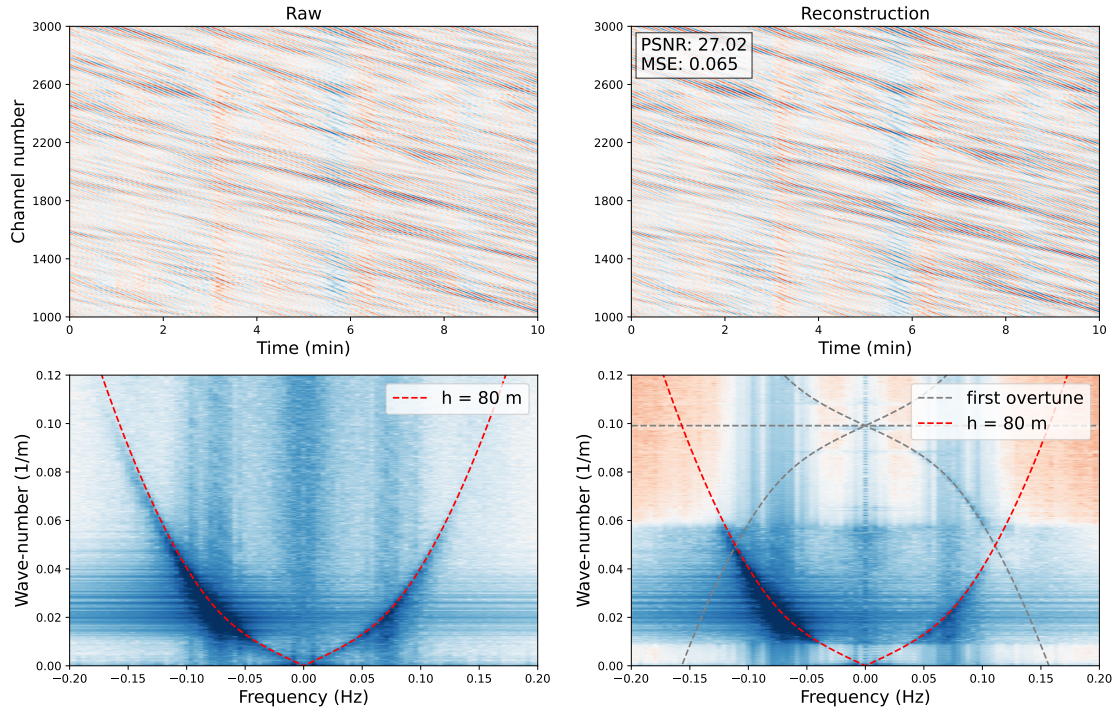


Figure C.7: (a-b) A 10-minute example of raw and reconstruction data from the Cook Inlet DAS experiment on the KKFL-S cable sampling at 25 Hz. The SHallow REcurrent Decoder (SHRED) model is trained using 151 out of 1000 decimated channels. Data from a segment of 2000 channels are shown. The corresponding data in the frequency-wavenumber (F-K) domain is shown in (c) and (d), respectively. Dispersion relation for ocean wave under an intermediate water depth setting ($h = 80$ m) is plotted as red dashed lines. The first overtone is modeled as dashed gray lines.

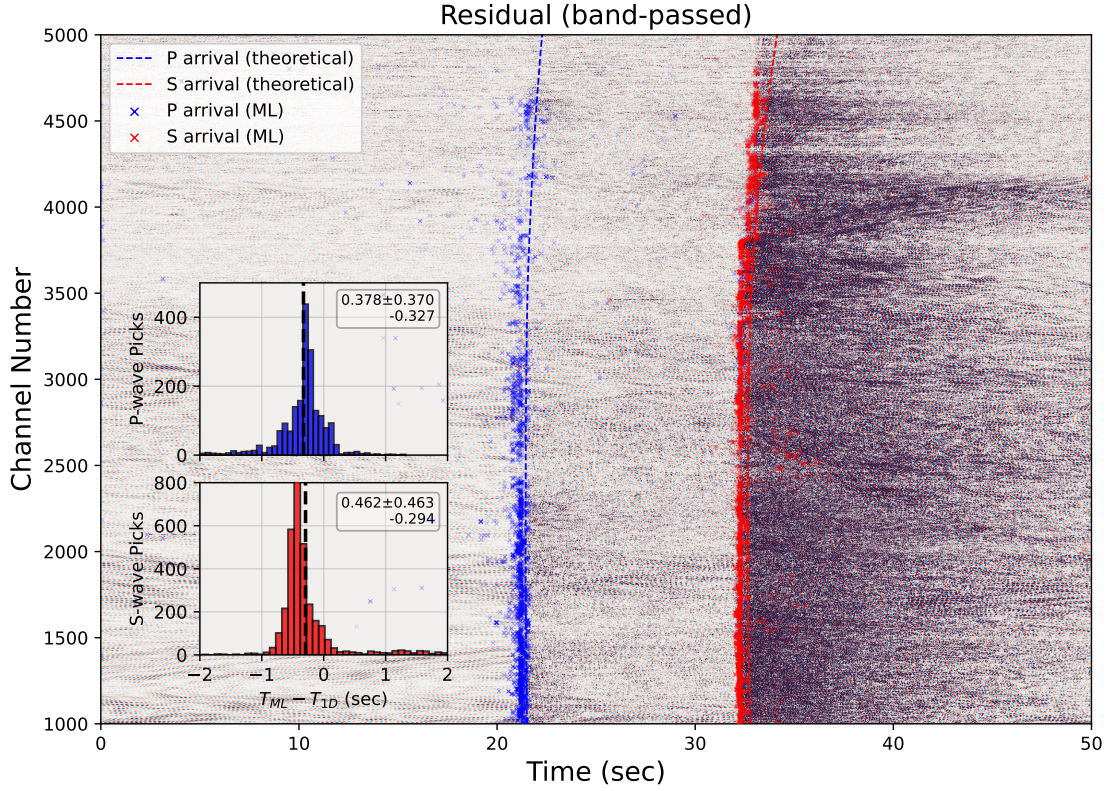


Figure C.8: A 50 s raw DAS recording from the Cook Inlet experiment. The selected window contains P- and S-wave arrivals from a regional M2.0 earthquake on 14 July 2023 (USGS event ID ak0238yvmdxr). A band-pass filter is applied for illustration purposes only, enhancing the earthquake onset. We use the EQCCT to pick the arrival time on the unfiltered residual data, marked by the blue and red crosses. Dashed lines show the theoretical arrival time calculated from Tau-P using the catalog hypocenter location and IASP-91 reference velocity model. The inset shows the difference between the ML picks (T_{ML}) and the theoretical arrival time (T_{1D}) for P- and S-wave onset. The mean absolute difference, the standard deviation as the uncertainties, and the mean value of the difference are reported in the upper right corner.

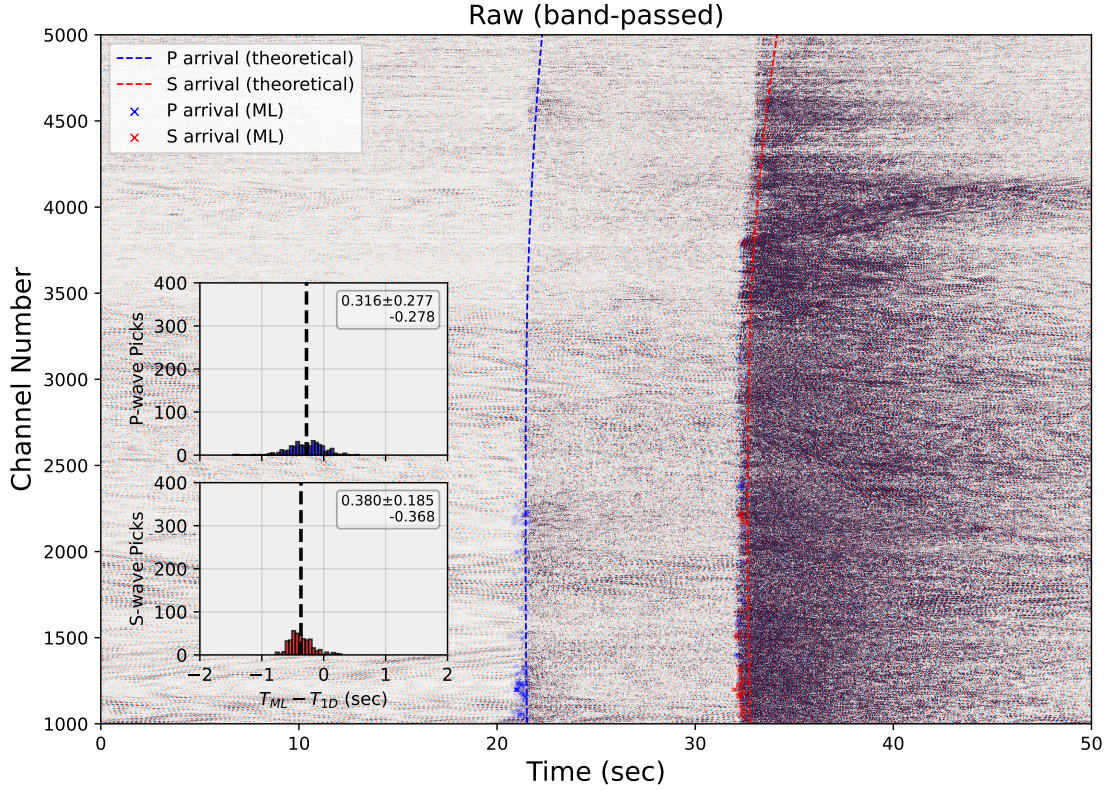


Figure C.9: A 50 s raw DAS recording from the Cook Inlet experiment. The selected window contains P- and S-wave arrivals from a regional M2.0 earthquake on 14 July 2023 (USGS event ID ak0238yvmdxr). A band-pass filter is applied for illustration purposes only, enhancing the earthquake onset. We use the ensemble learning earthquake phase picker (ELEM) to pick the arrival time on the unfiltered raw data, marked by the blue and red crosses. Dashed lines show the theoretical arrival time calculated from Tau-P using the catalog hypocenter location and IASP-91 reference velocity model. The inset shows the difference between the ML picks (T_{ML}) and the theoretical arrival time (T_{1D}) for P- and S-wave onset. The mean absolute difference, the standard deviation as the uncertainties, and the mean value of the difference are reported in the upper right corner.

D

Supplementary Materials for Chapter 5

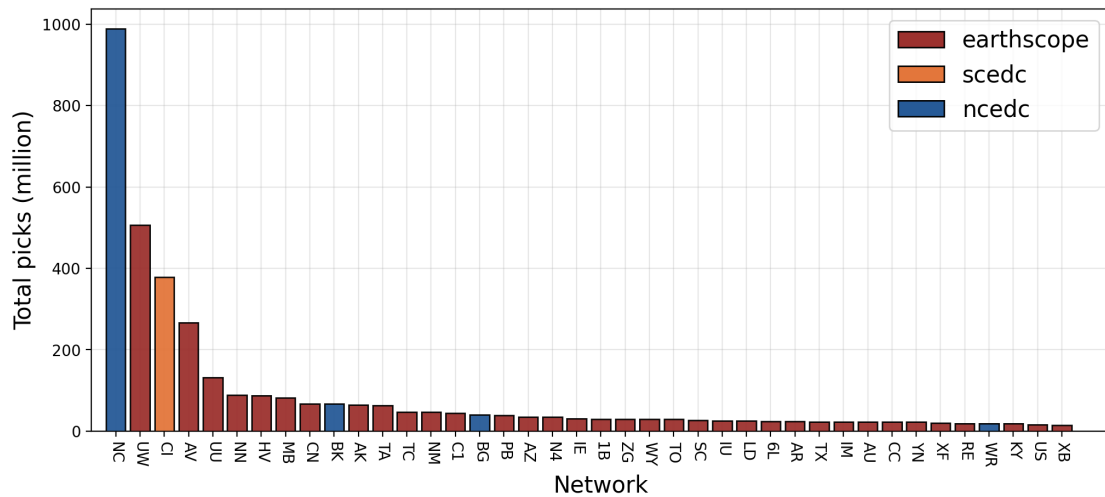
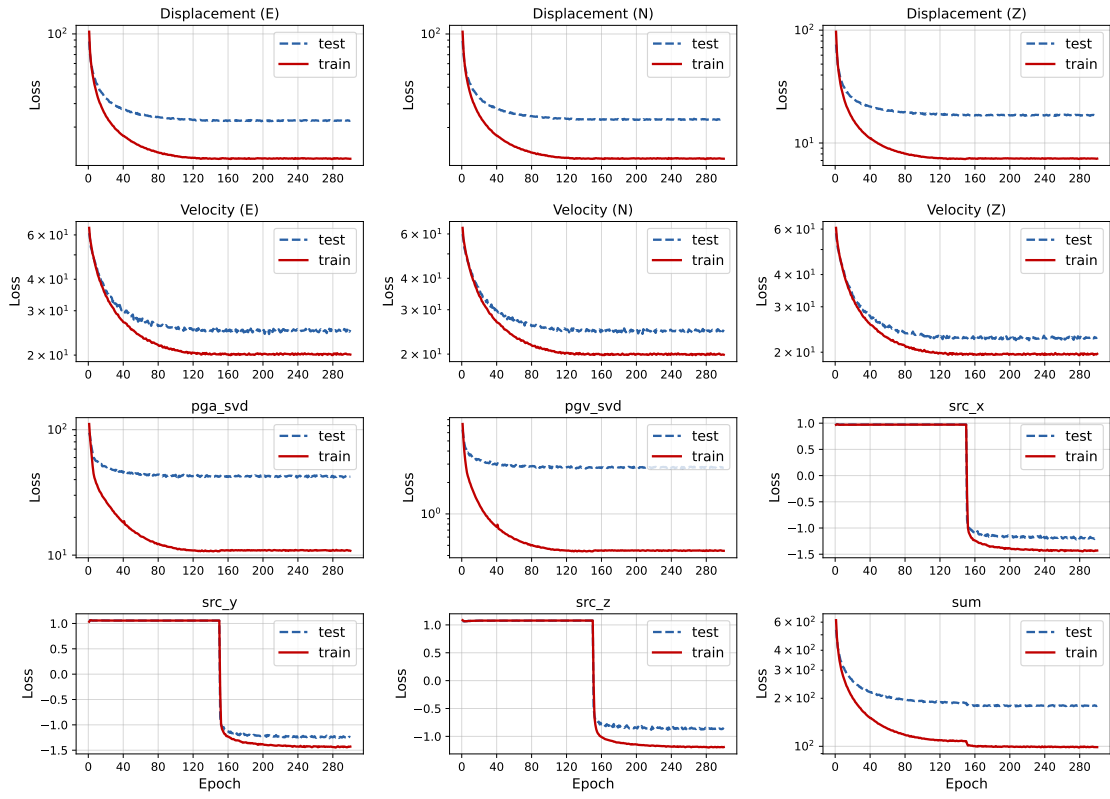


Figure D.1: Number of phase picks grouped by network codes, with the top 40 networks shown. Bars are color-coded by the hosting data center: EarthScope (red), SCEDC (orange), and NCEDC (blue).

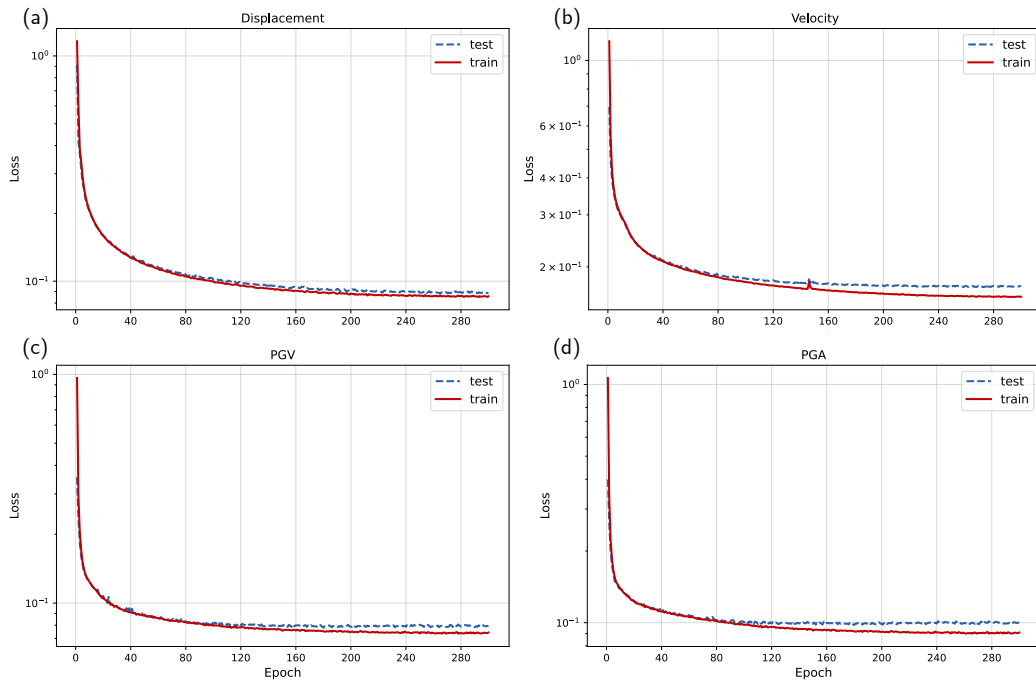
E

Supplementary Materials for Chapter 7



10

11 Figure E.1: Training (red line) and testing (dashed blue line) loss for Stage 1A wavefield reconstruction
 12 (epoch 1-150) and Stage 1B source location estimation (epoch 151-300).



14

15 Figure E.2: Training (red line) and testing (dashed blue line) loss for Stage 2 wavefield refinement of (a)
 16 displacement, (b) velocity, (c) PGV, and (d) PGA field, respectively.

References

- [1] Brad T Aagaard, Scott Marshall, Sarah Minson, Dan Boyd, Marine Denolle, Eric Fielding, Alice-Agnes Gabriel, Christine Goulet, Russell Graymer, Jeanne Hardebeck, et al. 2024 california community earth models for seismic hazard assessments workshop report. *arXiv preprint arXiv:2503.11545*, 2025. doi: 10.48550/arXiv.2503.11545.
- [2] A.A.Royer and M.G.Bostock. A comparative study of low frequency earthquake templates in northern cascadia. *Earth and Planetary Science Letters*, 402:247–256, 2014. doi: 10.1016/j.epsl.2013.08.040.
- [3] Ryan P. Abernathey, Tom Augspurger, Anderson Banihirwe, Charles C. Blackmon-Luca, Timothy J. Crone, Chelle L. Gentemann, Joseph J. Hamman, Naomi Henderson, Chiara Lepore, Theo A. McCaie, Niall H. Robinson, and Richard P. Signell. Cloud-native repositories for big scientific data. *Computing in Science & Engineering*, 23(2):26–35, 2021. doi: 10.1109/MCSE.2021.3059437.
- [4] Abien Fred Agarap. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, 2018. doi: 10.48550/arXiv.1803.08375.
- [5] Jonathan B Ajo-Franklin, Shan Dou, Nathaniel J Lindsey, Inder Monga, Chris Tracy, Michelle Robertson, Veronica Rodriguez Tribaldos, Craig Ulrich, Barry Freifeld, Thomas Daley, et al. Distributed acoustic sensing using dark fiber for near-surface characterization and broadband seismic event detection. *Scientific reports*, 9(1):1328, 2019. doi: 10.1038/s41598-018-36675-8.
- [6] Albuquerque Seismological Laboratory (ASL)/USGS. Global seismograph network - iris/usgs, 1988.
- [7] Albuquerque Seismological Laboratory (ASL)/USGS. United states national seismic network, 1990.
- [8] Bryce Allen, John Bresnahan, Lisa Childers, Ian Foster, Gopi Kandaswamy, Raj Kettimuthu, Jack Kordas, Mike Link, Stuart Martin, Karl Pickett, et al. Software as a service for data scientists. *Communications of the ACM*, 55(2):81–88, 2012. doi: 10.1145/2076450.2076468.
- [9] Rex Allen. Automatic phase pickers: Their present use and future prospects. *Bulletin of the Seismological Society of America*, 72(6B):S225–S242, 1982. doi: 10.1785/BSSA07206Bo225.

- [10] Richard M Allen and Hiroo Kanamori. The potential for earthquake early warning in southern california. *Science*, 300(5620):786–789, 2003. doi: 10.1126/science.1080912.
- [11] Richard M Allen, Paolo Gasparini, Osamu Kamigaichi, and Maren Bose. The status of earthquake early warning around the world: An introductory overview. *Seismological research letters*, 80(5): 682–693, 2009. doi: 10.1785/gssrl.80.5.682.
- [12] Bettina P Allmann, Peter M Shearer, and Egill Hauksson. Spectral discrimination between quarry blasts and earthquakes in southern california. *Bulletin of the Seismological Society of America*, 98(4):2073–2079, 2008. doi: 10.1785/0120070215.
- [13] Kate Allstadt. Extracting source characteristics and dynamics of the august 2010 mount meager landslide from broadband seismograms. *Journal of Geophysical Research: Earth Surface*, 118(3): 1472–1490, 2013. doi: 10.1002/jgrf.20110.
- [14] Kate E Allstadt, Robin S Matoza, Andrew B Lockhart, Seth C Moran, Jacqueline Caplan-Auerbach, Matthew M Haney, Weston A Thelen, and Stephen D Malone. Seismic and acoustic signatures of surficial mass movements at volcanoes. *Journal of Volcanology and Geothermal Research*, 364:76–106, 2018. doi: 10.1016/j.jvolgeores.2018.09.007.
- [15] KE Allstadt, B McVey, and S Malone. Seismogenic landslides, debris flows, and outburst floods in the western united states and canada from 1977 to 2017: Us geological survey data release, 2017.
- [16] Stephen J Arrowsmith, Daniel T Trugman, J MacCarthy, Karianne J Bergen, David Lumley, and Maria Beatrice Magnani. Big data seismology. *Reviews of Geophysics*, 60(2):e2021RG000769, 2022. doi: 10.1029/2021RG000769.
- [17] Richard C Aster, Adam T Ringler, Robert E Anthony, and Thomas A Lee. Increasing ocean wave energy observed in earth’s seismic wavefield since the late 20th century. *Nature Communications*, 14(1):6984, 2023. doi: 10.1038/s41467-023-42673-w.
- [18] James Atterholt, Zhongwen Zhan, and Yan Yang. Fault zone imaging with distributed acoustic sensing: Body-to-surface wave scattering. *Journal of Geophysical Research: Solid Earth*, 127(11): e2022JB025052, 2022. doi: 10.1029/2022JB025052.
- [19] Manochehr Bahavar, Kate E Allstadt, Mick Van Fossen, Stephen D Malone, and Chad Trabant. Exotic seismic events catalog (esec) data product. *Seismological Research Letters*, 90(3):1355–1363, 2019. doi: 10.1785/0220180402.

- [20] Manochehr Bahavar, Jonathan Delph, Asif Ashraf, Pieter-Ewald Share, Valerie Sahakian, Amanda Thomas, Loïc Bachelot, Emilie Hooft, Rasheed Ajala, and CRESCENT CVM Working Group. The cascadia region earthquake science center (crescent) community velocity model viewer, 03 2025.
- [21] Manochehr Bahavar, Richard Styron, Ashley Streig, Amanda Thomas, Loïc Bachelot, and CRESCENT CFM working group. The cascadia region earthquake science center (crescent) community fault model viewer, 03 2025.
- [22] Christian Baillard, Wayne C Crawford, Valérie Ballu, Clément Hibert, and Anne Mangeney. An automatic kurtosis-based p-and s-phase picker designed for local seismic networks. *Bulletin of the Seismological Society of America*, 104(1):394–409, 2014. doi: 10.1785/0120120347.
- [23] Noel M Bartlow. A long-term view of episodic tremor and slip in cascadia. *Geophysical Research Letters*, 47(3):e2019GL085303, 2020. doi: 10.1029/2019GL085303.
- [24] Robin Beckwith. Managing big data: Cloud computing and co-location centers. *Journal of Petroleum Technology*, 63(10):42–45, 2011. doi: 10.2118/1011-0042-JPT.
- [25] Yehuda Ben-Zion, Frank L Vernon, Yaman Ozakin, Dimitri Zigone, Zachary E Ross, Haoran Meng, Malcolm White, Juan Reyes, Dan Hollis, and Mitchell Barklage. Basic data features and results from a spatially dense seismic array on the San Jacinto fault zone. *Geophysical Journal International*, 202(1):370–380, 2015. doi: 10.1093/gji/ggv142.
- [26] Karianne J Bergen, Paul A Johnson, Maarten V de Hoop, and Gregory C Beroza. Machine learning for data-driven discovery in solid earth geoscience. *Science*, 363(6433):eaau0323, 2019. doi: 10.1126/science.aau0323.
- [27] Moritz Beyreuther, Robert Barsch, Lion Krischer, Tobias Megies, Yannik Behr, and Joachim Wassermann. Obspy: A python toolbox for seismology. *Seismological Research Letters*, 81(3): 530–533, 2010. doi: 10.1785/gssrl.81.3.530.
- [28] Michael J Bianco, Peter Gerstoft, Kim B Olsen, and Fan-Chi Lin. High-resolution seismic tomography of long beach, ca using machine learning. *Scientific reports*, 9(1):14987, 2019. doi: 10.1038/s41598-019-50381-z.
- [29] David M Boore, Jennie Watson-Lamprey, and Norman A Abrahamson. Orientation-independent

- measures of ground motion. *Bulletin of the seismological Society of America*, 96(4A):1502–1511, 2006. doi: 10.1785/0120050209.
- [30] David M Boore, Jonathan P Stewart, Emel Seyhan, and Gail M Atkinson. NGA-West2 equations for predicting PGA, PGV, and 5% damped PSA for shallow crustal earthquakes. *Earthquake Spectra*, 30(3):1057–1085, 2014. doi: 10.1193/070113EQS184M.
- [31] Roger D Borcherdt. Estimates of site-dependent response spectra for design (methodology and justification). *Earthquake Spectra*, 10(4):617–653, 1994. doi: 10.1193/1.1585791.
- [32] Thomas Bornstein, Dietrich Lange, Jannes Münchmeyer, Jack Woollam, Andreas Rietbrock, Grace Barcheck, Ingo Grevenmeyer, and Frederik Tilmann. Pickblue: Seismic phase picking for ocean bottom seismometers with deep learning. *Earth and Space Science*, 11(1):e2023EA003332, 2024. doi: 10.1029/2023EA003332.
- [33] Thomas Braun, Barbara Frigo, Bernardino Chiaia, Perry Bartelt, Daniela Famiani, and Joachim Wassermann. Seismic signature of the deadly snow avalanche of january 18, 2017, at rigopiano (italy). *Scientific reports*, 10(1):1–10, 2020. doi: 10.1038/s41598-020-75368-z.
- [34] Alexander Breuer, Yifeng Cui, and Alexander Heinecke. Petaflop seismic simulations in the public cloud. In *International Conference on High Performance Computing*, pages 167–185. Springer, 2019. doi: 10.1007/978-3-030-20656-7_9.
- [35] Cascades Volcano Observatory/USGS. Cascade chain volcano monitoring, 2001.
- [36] Derrick Chambers, Ge Jin, Ahmad Tourei, Abdul Hafiz Saeed Issah, Ariel Lellouch, Eileen Martin, Donglin Zhu, Aaron Girard, Shihao Yuan, Thomas Cullison, Tomas Snyder, Seunghoo Kim, Nicholas Danes, Nikhil Punithan, M. Shawn Boltz, and Manuel M. Mendoza. Dascore: a python library for distributed fiber optic sensing. *Seismica*, 3(2), 2024. doi: 10.26443/seismica.v3i2.1184.
- [37] Po Chen, En-Jui Lee, and Liqiang Wang. A cloud-based synthetic seismogram generator implemented using windows azure. *Earthquake Science*, 26:321–329, 2013. doi: 10.1007/s11589-013-0038-8.
- [38] Yangkang Chen. Sigrecover: Recovering signal from noise in distributed acoustic sensing data processing. *Seismological Research Letters*, 95(3):1976–1985, 2024. doi: 10.1785/0220230370.
- [39] Yangkang Chen and Sergey Fomel. Emd-seislet transform. *Geophysics*, 83(1):A27–A32, 2018. doi: 10.1190/geo2017-0554.1.

- [40] Yangkang Chen, Min Bai, and Yunfeng Chen. Obtaining free usarray data by multi-dimensional seismic reconstruction. *Nature Communications*, 10(1):4434, 2019. doi: 10.1038/s41467-019-12405-0.
- [41] Feng Cheng, Benxin Chi, Nathaniel J Lindsey, T Craig Dawe, and Jonathan B Ajo-Franklin. Utilizing distributed acoustic sensing and ocean bottom fiber optic cables for submarine structural characterization. *Scientific reports*, 11(1):5613, 2021. doi: 10.1038/s41598-021-84845-y.
- [42] Feng Cheng, Nathaniel J Lindsey, Valeriia Sobolevskaia, Shan Dou, Barry Freifeld, Todd Wood, Stephanie R James, Anna M Wagner, and Jonathan B Ajo-Franklin. Watching the cryosphere thaw: Seismic monitoring of permafrost degradation using distributed acoustic sensing during a controlled heating experiment. *Geophysical Research Letters*, 49(10):e2021GL097195, 2022. doi: 10.1029/2021GL097195.
- [43] Małgorzata Chmiel, Fabian Walter, Michaela Wenner, Zhen Zhang, Brian W McArdell, and Clement Hibert. Machine learning improves debris flow warning. *Geophysical Research Letters*, 48(3):e2020GL090874, 2021. doi: 10.1029/2020GL090874.
- [44] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014. doi: 10.48550/arXiv.1412.3555.
- [45] T. Clements and M. A. Denolle. The seismic signature of california’s earthquakes, droughts, and floods. *Journal of Geophysical Research: Solid Earth*, 128(1):e2022JB025553, 2023. doi: 10.1029/2022JB025553.
- [46] T. Clements, J. F. Schmitt, and Denolle. Cloud-native analysis of southern california waveform data. In *SCEC Annual Meeting, poster*, 2020. URL <https://central.scec.org/meetings/2020/am/poster/193>.
- [47] T Clements, Elizabeth S Cochran, A Baltay, SE Minson, and CE Yoon. Grapes: earthquake early warning by passing seismic vectors through the grapevine. *Geophysical Research Letters*, 51(9):e2023GL107389, 2024. doi: 10.1029/2023GL107389.
- [48] Timothy Clements and Marine A. Denolle. Seisnoise.jl: Ambient seismic noise cross correlation on the cpu and gpu in julia. *Seismological Research Letters*, 92(1):517–527, 09 2020. ISSN 0895-0695. doi: 10.1785/0220200192.

- [49] Andrew Collette. *Python and HDF5: unlocking scientific data*. " O'Reilly Media, Inc.", 2013.
- [50] Andrew Collette, Thomas Kluyver, Thomas A Caswell, James Tocknell, Jerome Kieffer, Anthony Scopatz, Darren Dale, Chen, Aleksandar Jelenak, payno, juliagarriga, Thomas VINCENT, Pierlauro Sciarelli, Valentin Valls, Satrajit Ghosh, Ulrik Kofoed Pedersen, jakirkham, Martin Raspaud, Aaron Parsons, Hameer Abbasi, John Readey, Andrey Paramonov, Lawrence Chan, V. Armando Solé, jialin, Cyril Danilevski, Yu Feng, Ghislain Antony Vaillant, Martin Teichmann, and Matthieu Brucher. *h5py/h5py*: 3.2.1, 2021.
- [51] EA Collins, KE Allstadt, C Groult, C Hibert, JP Malet, LD Toney, and EK Bessette-Kirton. Seismogenic landslides and other mass movements: Us geological survey data release, 2022.
- [52] The Turing Way Community. The turing way: A handbook for reproducible, ethical and collaborative research, 04 2025.
- [53] Cesium Consortium et al. Cesium-an open-source javascript library for world-class 3d globes and maps, 2018.
- [54] Robert S Crosson. Small earthquakes, structure, and tectonics of the puget sound region. *Bulletin of the Seismological Society of America*, 62(5):1133–1171, 1972. doi: 10.1785/BSSA0620051133.
- [55] H Philip Crotwell, Thomas J Owens, Jeroen Ritsema, et al. The taup toolkit: Flexible seismic travel-time and ray-path utilities. *Seismological Research Letters*, 70:154–160, 1999. doi: 10.1785/gssrl.70.2.154.
- [56] Tamara Dancheva, Unai Alonso, and Michael Barton. Cloud benchmarking and performance analysis of an hpc application in amazon ec2. *Cluster Computing*, 27(2):2273–2290, 2024. doi: 10.1007/s10586-023-04060-4.
- [57] Jonathan R Delph, Amanda M Thomas, and Alan Levander. Subcretionary tectonics: Linking variability in the expression of subduction along the cascadia forearc. *Earth and Planetary Science Letters*, 556:116724, 2021. doi: 10.1016/j.epsl.2020.116724.
- [58] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. doi: 10.1109/CVPR.2009.5206848.
- [59] Marine A. Denolle, Carl Tape, Ebru Bozdağ, Yinzhi Wang, Felix Waldhauser, Alice-Agnes Gabriel, Jochen Braunmiller, Bryant Chow, Liang Ding, Kuan-Fu Feng, Ayon Ghosh, Nathan Groebner,

- Aakash Gupta, Zoe Krauss, Amanda M. McPherson, Masaru Nagaso, Zihua Niu, Yiyu Ni, Rıdvan Örsvuran, Gary Pavlis, Felix Rodriguez-Cardozo, Theresa Sawi, David Schaff, Nico Schliwa, David Schneller, Qibin Shi, Julien Thurin, Chenxiao Wang, Kaiwen Wang, Jeremy Wing Ching Wong, Sebastian Wolf, and Congcong Yuan. Training the next generation of seismologists: Delivering research-grade software education for cloud and hpc computing through diverse training modalities. *Seismological Research Letters*, 06 2025. ISSN 0895-0695. doi: 10.1785/0220240413.
- [60] Clare Donaldson, Tom Winder, Corentin Caudron, and Robert S White. Crustal seismic velocity responds to a magmatic intrusion and seasonal loading in iceland’s northern volcanic zone. *Science Advances*, 5(11):eaax6642, 2019. doi: 10.1126/sciadv.aax6642.
- [61] Bin Dong, Verónica Rodríguez Tribaldos, Xin Xing, Suren Byna, Jonathan Ajo-Franklin, and Kesheng Wu. Dassa: Parallel data storage and analysis for subsurface event detection. In *2020 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pages 254–263. IEEE, 2020. doi: 10.1109/IPDPS47924.2020.00035.
- [62] Bin Dong, Alex Popescu, Verónica Rodríguez Tribaldos, Suren Byna, Jonathan Ajo-Franklin, Kesheng Wu, and the Imperial Valley Dark Fiber Team. Real-time and post-hoc compression for data from distributed acoustic sensing. *Computers & Geosciences*, 166:105181, 2022. doi: 10.1016/j.cageo.2022.105181.
- [63] John R Dormand and Peter J Prince. A family of embedded runge-kutta formulae. *Journal of computational and applied mathematics*, 6(1):19–26, 1980. doi: 10.1016/0771-050X(80)90013-3.
- [64] Alexander S Douglass, Shima Abadi, and Bradley P Lipovsky. Distributed acoustic sensing for detecting near surface hydroacoustic signals. *JASA Express Letters*, 3(6), 2023. doi: 10.1121/10.0019703.
- [65] Herb Dragert, Kelin Wang, and Thomas S James. A silent slip event on the deeper cascadia subduction interface. *Science*, 292(5521):1525–1528, 2001. doi: 10.1126/science.1060152.
- [66] Ariane Ducellier and Kenneth C Creager. An 8-year-long low-frequency earthquake catalog for southern cascadia. *Journal of Geophysical Research: Solid Earth*, 127(4):e2021JB022986, 2022. doi: 10.1029/2021JB022986.
- [67] Bogdan Dumitrescu and Paul Irofti. *Dictionary learning algorithms and applications*. Springer, 2018. doi: 10.1007/978-3-319-78674-2.

- [68] Emilien Dupont, Adam Goliński, Milad Alizadeh, Yee Whye Teh, and Arnaud Doucet. Coin: Compression with implicit neural representations. *arXiv preprint arXiv:2103.03123*, 2021. doi: 10.48550/arXiv.2103.03123.
- [69] Emilien Dupont, Hrushikesh Loya, Milad Alizadeh, Adam Goliński, Yee Whye Teh, and Arnaud Doucet. Coin++: Neural compression across modalities. *arXiv preprint arXiv:2201.12904*, 2022. doi: 10.48550/arXiv.2201.12904.
- [70] Megan R Ebers, Jan P Williams, Katherine M Steele, and J Nathan Kutz. Leveraging arbitrary mobile sensor trajectories with shallow recurrent decoder networks for full-state reconstruction. *arXiv preprint arXiv:2307.11793*, 2023. doi: 10.48550/arXiv.2307.11793.
- [71] N Benjamin Erichson, Lionel Mathelin, Zhewei Yao, Steven L Brunton, Michael W Mahoney, and J Nathan Kutz. Shallow neural networks for fluid flow reconstruction with limited sensors. *Proceedings of the Royal Society A*, 476(2238):20200097, 2020. doi: 10.1098/rspa.2020.0097.
- [72] Jiaqi Fang, Yan Yang, Zhichao Shen, Ettore Biondi, Xin Wang, Ethan F Williams, Matthew W Becker, Dominic Eslamian, and Zhongwen Zhan. Directional sensitivity of das and its effect on rayleigh-wave tomography: A case study in oxnard, california. *Seismological Society of America*, 94(2A):887–897, 2023. doi: 10.1785/0220220235.
- [73] Mohammed Outhmane Faouzi Zizi and Pierre Turquais. A dictionary learning method for seismic data compression. *Geophysics*, 87(2):V101–V116, 2022. doi: 10.1190/geo2020-0948.1.
- [74] Kurt Feigl, Nicole Taverna, and Michael Rossol. Porotomo natural laboratory horizontal and vertical distributed acoustic sensing data, 2016.
- [75] Yitian Feng, Weiqiang Zhu, and Xinzheng Lu. Quakeformer: A uniform approach to earthquake ground motion prediction using masked transformers. *arXiv preprint arXiv:2412.00815*, 2024. doi: 10.48550/arXiv.2412.00815.
- [76] Zheng-yi Feng. The seismic signatures of the surge wave from the 2009 xiaolin landslide-dam breach in taiwan. *Hydrological Processes*, 26(9):1342–1351, 2012. doi: 10.1002/hyp.8239.
- [77] Andreas Fichtner, Laura Ermert, and Alexey Gokhberg. Seismic noise correlation on heterogeneous supercomputers. *Seismological Research Letters*, 88(4):1141–1145, 05 2017. ISSN 0895-0695. doi: 10.1785/0220170043.

- [78] Sergey Fomel and Yang Liu. Seislet transform and seislet frame. *Geophysics*, 75(3):V25–V38, 2010. doi: 10.1190/1.3380591.
- [79] Shuwei Gan, Shoudong Wang, Yangkang Chen, Yizhuo Zhang, and Zhaoyu Jin. Dealiased seismic data interpolation using seislet transform with low-frequency constraint. *IEEE Geoscience and remote sensing letters*, 12(10):2150–2154, 2015. doi: 10.1109/LGRS.2015.2453119.
- [80] Shuwei Gan, Shoudong Wang, Yangkang Chen, Xiaohong Chen, Weiling Huang, and Hanming Chen. Compressive sensing for seismic data reconstruction via fast projection onto convex sets based on seislet transform. *Journal of Applied Geophysics*, 130:194–208, 2016. doi: 10.1016/j.jappgeo.2016.03.033.
- [81] Haiying Gao and Yang Shen. A preliminary full-wave ambient-noise tomography model spanning from the juan de fuca and gorda spreading centers to the cascadia volcanic arc. *Seismological Research Letters*, 86(5):1253–1260, 2015. doi: 10.1785/0220150103.
- [82] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. Convolutional sequence to sequence learning. In *International conference on machine learning*, pages 1243–1252. PMLR, 2017. doi: 10.48550/arXiv.1705.03122.
- [83] Paul G. Somerville Gene A. Ichinose, Hong Kie Thio. Rupture process and near-source shaking of the 1965 seattle-tacoma and 2001 nisqually, intraslab earthquakes. *Geophysical Research Letters*, 31(10):1–4, 2004. doi: 10.1029/2004GL019668.
- [84] Chelle Leigh Gentemann, Chris Holdgraf, Ryan Abernathey, Daniel Crichton, James Colliander, Edward Joseph Kearns, Yuvi Panda, and Richard P Signell. Science storms the cloud. *AGU Advances*, 2(2):e2020AV000354, 2021. doi: 10.1029/2020AV000354.
- [85] Jonatan Glehman, Alice-Agnes Gabriel, Thomas Ulrich, Marlon Dale Ramos, Yihe Huang, and Eric O Lindsey. Partial ruptures governed by the complex interplay between geodetic slip deficit, rigidity, and pore fluid pressure in 3d cascadia dynamic rupture simulations. *EarthArXiv*, 2024. doi: 10.31223/X5GH66.
- [86] J Gomberg, B Sherrod, M Trautman, E Burns, and D Snyder. Contemporary seismicity in and around the yakima fold-and-thrust belt in eastern washington. *Bulletin of the Seismological Society of America*, 102(1):309–320, 2012. doi: 10.1785/0120110065.

- [87] Joan Gomberg and Paul Bodin. The productivity of cascadia aftershock sequences. *Bulletin of the Seismological Society of America*, 111(3):1494–1507, 2021. doi: 10.1785/0120200344.
- [88] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014. doi: 10.48550/arXiv.1406.2661.
- [89] Robert Graves, Thomas H Jordan, Scott Callaghan, Ewa Deelman, Edward Field, Gideon Juve, Carl Kesselman, Philip Maechling, Gaurang Mehta, Kevin Milner, et al. CyberShake: A physics-based seismic hazard model for southern California. *Pure and Applied Geophysics*, 168(3-4): 367–381, 2011. doi: 10.1007/s00024-010-0161-6.
- [90] William Gropp, Ewing Lusk, Nathan Doss, and Anthony Skjellum. A high-performance, portable implementation of the mpi message passing interface standard. *Parallel computing*, 22(6):789–828, 1996. doi: 10.1016/0167-8191(96)00024-5.
- [91] Antonio Guimarães, Luis Lacalle, Charles B. Rodamilans, and Edson Borin. High-performance io for seismic processing on the cloud. *Concurrency and Computation: Practice and Experience*, 33(18):e6250, 2021. doi: 10.1002/cpe.6250.
- [92] Ted Habermann, Chad Trabant, Tim Ronan, et al. Common data and metadata models for geophysical data in the cloud, 2021. Authorea, December 29, 2021.
- [93] Charles R Harris, K Jarrod Millman, Stéfan J Van Der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J Smith, et al. Array programming with numpy. *Nature*, 585(7825):357–362, 2020. doi: 10.1038/s41586-020-2649-2.
- [94] J.R. Hartog, P.A. Friberg, V.C. Kress, P. Bodin, and R. Bhadha. Open-source anss quake monitoring system software. *Seismological Research Letters*, 91:677–686, 2019. doi: 10.1785/0220190219.
- [95] Egill Hauksson, Clara Yoon, Ellen Yu, Jennifer R Andrews, Marcos Alvarez, Rayo Bhadha, and Valerie Thomas. Caltech/usgs southern california seismic network (scsn) and southern california earthquake data center (scedc): Data availability for the 2019 ridgecrest sequence. *Seismological Research Letters*, 91(4):1961–1970, 2020. doi: 10.1785/0220190290.
- [96] Gavin Hayes. Slab2 - A Comprehensive Subduction Zone Geometry Model, 2018. Type: dataset.
- [97] Gavin Hayes. Slab2 - A Comprehensive Subduction Zone Geometry Model, 2018.

- [98] Mike Hearne and Heather E Schovanec. libcomcat Software Release, 2020. Medium: Other.
- [99] Alexander Heinecke, Alexander Breuer, Sebastian Rettenberger, Michael Bader, Alice-Agnes Gabriel, Christian Pelties, Arndt Bode, William Barth, Xiang-Ke Liao, Karthikeyan Vaidyanathan, et al. Petascale high order dynamic rupture earthquake simulations on heterogeneous supercomputers. In *SC'14: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 3–14. IEEE, 2014. doi: 10.1109/SC.2014.6.
- [100] Emad B Helal, Omar M Saad, Ali G Hafez, Yangkang Chen, and Gamal M Dousoky. Seismic data compression using deep learning. *IEEE Access*, 9:58161–58169, 2021. doi: 10.1109/ACCESS.2021.3073090.
- [101] Robert B Herrmann. Fasthypo-a hypocenter location program. *Earthquake notes*, 50(2):25–38, 1979. doi: 10.1785/gssrl.50.2.25.
- [102] C Hibert, D Michéa, F Provost, J-P Malet, and M Geertsema. Exploration of continuous seismic recordings with a machine learning approach to document 20 yr of landslide activity in alaska. *Geophysical Journal International*, 219(2):1138–1147, 2019. ISSN 0956-540X. doi: 10.1093/gji/ggz354.
- [103] Clément Hibert, Göran Ekström, and Colin P Stark. Dynamics of the bingham canyon mine landslides from seismic signal analysis. *Geophysical research letters*, 41(13):4535–4541, 2014. doi: 10.1002/2014GL060592.
- [104] Clément Hibert, Anne Mangeney, Gilles Grandjean, Christian Baillard, Diane Rivet, Nikolai M Shapiro, Claudio Satriano, Alessia Maggi, Patrice Boissier, Valérie Ferrazzini, et al. Automated identification, location, and volume estimation of rockfalls at piton de la fournaise volcano. *Journal of Geophysical Research: Earth Surface*, 119(5):1082–1105, 2014. doi: 10.1002/2013JF002970.
- [105] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. doi: 10.48550/arXiv.2006.11239.
- [106] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. doi: 10.1162/neco.1997.9.8.1735.
- [107] Shigeki Horiuchi, Hiroaki Negishi, Kana Abe, Aya Kamimura, and Yukio Fujinawa. An automatic processing system for broadcasting earthquake alarms. *Bulletin of the Seismological Society of America*, 95(2):708–718, 2005. doi: 10.1785/0120030133.

- [108] S. Hoyer and J. Hamman. xarray: N-D labeled arrays and datasets in Python. *Journal of Open Research Software*, 5(1):10, 2017. doi: 10.5334/jors.148.
- [109] John D Hunter. Matplotlib: A 2d graphics environment. *Computing in science & engineering*, 9(03):90–95, 2007. doi: 10.1109/MCSE.2007.55.
- [110] Alexander R Hutko, Manochehr Bahavar, Chad Trabant, Robert T Weekly, Mick Van Fossen, and Timothy Ahern. Data products at the iris-dmc: Growth and usage. *Seismological Research Letters*, 88(3):892–903, 2017. doi: 10.1785/0220160190.
- [111] IRIS Transportable Array. Usarray transportable array, 2003.
- [112] Abdul Hafiz S. Issah and Eileen R. Martin. Impact of lossy compression errors on passive seismic data analyses. *Seismological Research Letters*, 2024. ISSN 0895-0695. doi: 10.1785/0220230314.
- [113] Paul C Jennings and Hiroo Kanamori. Effect of distance on local magnitudes found from strong-motion records. *Bulletin of the Seismological Society of America*, 73(1):265–280, 1983. doi: 10.1785/BSSA0730010265.
- [114] Chengxin Jiang and Marine A Denolle. Noisepty: A new high-performance python tool for ambient-noise seismology. *Seismological Research Letters*, 91(3):1853–1866, 2020. doi: 10.1785/0220190364.
- [115] Cyril Journeau, Amanda Thomas, Rachel Abercrombie, Brenton Hirao, Douglas Toomey, Emilie Hooft, Mingqi Liu, Sylvain Barbot, and Václav Kuna. Obs data mining and earthquake swarms analysis reveal the complex structure and dynamics of the blanco fracture zone. Technical report, Copernicus Meetings, 2025.
- [116] Y Kaneko and PM Shearer. Variability of seismic source spectra, estimated stress drop, and radiated energy, derived from cohesive-zone models of symmetrical and asymmetrical circular and elliptical ruptures. *Journal of Geophysical Research: Solid Earth*, 120(2):1053–1079, 2015. doi: 10.1002/2014JB011642.
- [117] Martin Käser, Cristobal Castro, Verena Hermann, and Christian Pelties. Seissol—a software for seismic wave propagation simulations. In *High Performance Computing in Science and Engineering, Garching/Munich 2009: Transactions of the Fourth Joint HLRB and KONWIHR Review and Results Workshop, Dec. 8-9, 2009, Leibniz Supercomputing Centre, Garching/Munich, Germany*, pages 281–292. Springer, 2010. doi: 10.1007/978-3-642-13872-0_24.

- [118] BLN Kennett and ER Engdahl. Traveltimes for global earthquake location and phase identification. *Geophysical Journal International*, 105(2):429–465, 1991. doi: 10.1111/j.1365-246X.1991.tb06724.x.
- [119] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. doi: 10.48550/arXiv.1412.6980.
- [120] Fred W Klein. User’s guide to hypoinverse-2000, a fortran program to solve for earthquake locations and magnitudes. Technical report, US Geological Survey, 2002.
- [121] Dimitri Komatitsch and Jeroen Tromp. Introduction to the spectral element method for three-dimensional seismic wave propagation. *Geophysical journal international*, 139(3):806–822, 1999. doi: 10.1046/j.1365-246x.1999.00967.x.
- [122] Qingkai Kong, Daniel T Trugman, Zachary E Ross, Michael J Bianco, Brendan J Meade, and Peter Gerstoft. Machine learning in seismology: Turning data into insights. *Seismological Research Letters*, 90(1):3–14, 2019. doi: 10.1785/0220180259.
- [123] Keith D. Koper, Monique M. Holt, Jonathan R. Voyles, Relu Burlacu, Moira L. Pyle, Ruijia Wang, and Brandon Schmandt. Discrimination of small earthquakes and buried single-fired chemical explosions at local distances (<150 km) in the western united states from comparison of local magnitude (ml) and coda duration magnitude (mc). *Bulletin of the Seismological Society of America*, 111(1):558–570, 2020. ISSN 0037-1106. doi: 10.1785/0120200188.
- [124] Zoe Krauss, Yiyu Ni, Scott Henderson, and Marine Denolle. Seismology in the cloud: guidance for the individual researcher. *Seismica*, 2(2), 08 2023. doi: 10.26443/seismica.v2i2.979.
- [125] Lion Krischer, Tobias Megies, Robert Barsch, Moritz Beyreuther, Thomas Lecocq, Corentin Caudron, and Joachim Wassermann. Obspy: a bridge for seismology into the scientific python ecosystem. *Computational Science & Discovery*, 8(1):014003, 2015. ISSN 1749-4699. doi: 10.1088/1749-4699/8/1/014003.
- [126] Lion Krischer, James Smith, Wenjie Lei, Matthieu Lefebvre, Youyi Ruan, Elliott Sales de Andrade, Norbert Podhorszki, Ebru Bozdağ, and Jeroen Tromp. An adaptable seismic data format. *Geophysical Supplements to the Monthly Notices of the Royal Astronomical Society*, 207(2):1003–1011, 2016. doi: 10.1093/gji/ggw319.

- [127] Lion Krischer, Yongki Andita Aiman, Timothy Bartholomaeus, Stefanie Donner, Martin van Driel, Kenneth Duru, Kristina Garina, Kilian Gessele, Tomy Gunawan, Sarah Hable, Céline Hadziioannou, Mathijs Koymans, John Leeman, Fabian Lindner, Angel Ling, Tobias Megies, Ceri Nunn, Ashim Rijal, Johannes Salvermoser, Sujania Talavera Soza, Carl Tape, Taufiq Taufiqurrahman, David Vargas, Joachim Wassermann, Florian Wölfl, Mitch Williams, Stephanie Wollherr, and Heiner Igel. seismo-live: An educational online library of jupyter notebooks for seismology. *Seismological Research Letters*, 89(6):2413–2419, 09 2018. ISSN 0895-0695. doi: 10.1785/0220180167.
- [128] Wenhuan Kuang, Congcong Yuan, and Jie Zhang. Real-time determination of earthquake focal mechanism via deep learning. *Nature communications*, 12(1):1432, 2021. doi: 10.1038/s41467-021-21670-x.
- [129] Gregory M Kurtzer, Vanessa Sochat, and Michael W Bauer. Singularity: Scientific containers for mobility of compute. *PLoS one*, 12(5):e0177459, 2017. doi: 10.1371/journal.pone.0177459.
- [130] J Nathan Kutz, Maryam Reza, Farbod Faraji, and Aaron Knoll. Shallow recurrent decoder for reduced order modeling of plasma dynamics. *arXiv preprint arXiv:2405.11955*, 2024. doi: 10.48550/arXiv.2405.11955.
- [131] Fernando Lara, Román Lara-Cueva, Julio C Larco, Enrique V Carrera, and Rubén León. A deep learning approach for automatic recognition of seismo-volcanic events at the cotopaxi volcano. *Journal of Volcanology and Geothermal Research*, 409:107142, 2021. doi: 10.1016/j.jvolgeores.2020.107142.
- [132] Pablo Lara, Quentin Bletery, Jean-paul Ampuero, Adolfo Inza, and Hernando Tavera. Earthquake early warning starting from 3 s of records on a single station with machine learning. *Journal of Geophysical Research: Solid Earth*, 128(11):e2023JB026575, 2023. doi: 10.1029/2023JB026575.
- [133] En-Jui Lee, Po Chen, Thomas H Jordan, Phillip B Maechling, Marine AM Denolle, and Gregory C Beroza. Full-3-d tomography for crustal structure in southern california based on the scattering-integral and the adjoint-wavefield methods. *Journal of Geophysical Research: Solid Earth*, 119(8):6421–6451, 2014. doi: 10.1002/2014JB011346.
- [134] Fanny Lehmann, Filippo Gatti, Michaël Bertin, and Didier Clouteau. Synthetic ground motions in heterogeneous geologies from various sources: the hemew s-3d database. *Earth System Science Data*, 16(9):3949–3972, 2024. doi: 10.5194/essd-16-3949-2024.

- [135] Nicolas Lemarchand and Jean-Robert Grasso. Interactions between earthquakes and volcano activity. *Geophysical Research Letters*, 34(24), 2007. doi: 10.1029/2007GL031438.
- [136] Zefeng Li, Zhigang Peng, Dan Hollis, Lijun Zhu, and James McClellan. High-resolution seismic event detection using local similarity for large-n arrays. *Scientific reports*, 8(1):1646, 2018. doi: 10.1038/s41598-018-19728-w.
- [137] Zefeng Li, Zhichao Shen, Yan Yang, Ethan Williams, Xin Wang, and Zhongwen Zhan. Rapid response to the 2019 ridgecrest earthquake with distributed acoustic sensing. *AGU Advances*, 2(2):e2021AV000395, 2021. doi: 10.1029/2021AV000395.
- [138] Jiun-Ting Lin, Amanda M. Thomas, Loïc Bachelot, Douglas Toomey, Jake Searcy, and Diego Melgar. Detection of hidden low-frequency earthquakes in southern vancouver island with deep learning. *Seismica*, 2(4), 10 2024. doi: 10.26443/seismica.v2i4.1134.
- [139] Nathaniel J Lindsey and Eileen R Martin. Fiber-optic seismology. *Annual Review of Earth and Planetary Sciences*, 49:309–336, 2021. doi: 10.1146/annurev-earth-072420-065213.
- [140] Nathaniel J Lindsey, T Craig Dawe, and Jonathan B Ajo-Franklin. Illuminating seafloor faults and ocean dynamics with dark fiber distributed acoustic sensing. *Science*, 366(6469):1103–1107, 2019. doi: 10.1126/science.aay5881.
- [141] Nathaniel J Lindsey, Siyuan Yuan, Ariel Lellouch, Lucia Gualtieri, Thomas Lecocq, and Biondo Biondi. City-scale dark fiber das measurements of infrastructure use during the covid-19 pandemic. *Geophysical research letters*, 47(16):e2020GL089931, 2020. doi: 10.1029/2020GL089931.
- [142] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022. doi: 10.48550/arXiv.2210.02747.
- [143] Bradley Lipovsky. Seadas-n: Distributed acoustic sensing experiment in seattle, washington, usa, 2023.
- [144] Bradley Lipovsky. Distributed acoustic sensing experiment in whidbey island, washington, usa, 2023.
- [145] Chuanming Liu, Shane Zhang, Anne F Sheehan, and Michael H Ritzwoller. Surface wave isotropic and azimuthally anisotropic dispersion across alaska and the alaska-aleutian subduction

- zone. *Journal of Geophysical Research: Solid Earth*, 127(11):e2022JB024885, 2022. doi: 10.1029/2022JB024885.
- [146] Tianlin Liu, Jannes Münchmeyer, Laura Laurenti, Chris Marone, Maarten V de Hoop, and Ivan Dokmanić. Seislm: a foundation model for seismic waveforms. *arXiv preprint arXiv:2410.15765*, 2024. doi: 10.48550/arXiv.2410.15765.
- [147] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. doi: 10.48550/arXiv.1608.03983.
- [148] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. doi: 10.48550/arXiv.1711.05101.
- [149] L. V. Luna and O. Korup. Seasonal landslide activity lags annual precipitation pattern in the pacific northwest. *Geophysical Research Letters*, 49(18):e2022GL098506, 2022. doi: 10.1029/2022GL098506. e2022GL098506 2022GL098506.
- [150] Jonathan MacCarthy, Omar Marcillo, and Chad Trabant. Putting the cloud to work for seismology. *Eos, Transactions American Geophysical Union (Online)*, 100(LA-UR-18-30252), 2019. doi: 10.1029/2019EO119741.
- [151] Jonathan MacCarthy, Omar Marcillo, and Chad Trabant. Seismology in the cloud: A new streaming workflow. *Seismological Research Letters*, 91(3):1804–1812, 03 2020. ISSN 0895-0695. doi: 10.1785/0220190357.
- [152] Jonathan MacCarthy, Omar Marcillo, and Chad Trabant. Seismology in the cloud: A new streaming workflow. *Seismological Research Letters*, 91(3):1804–1812, 2020. doi: 10.1785/0220190357.
- [153] Takuto Maeda, Kiwamu Nishida, Ryota Takagi, and Kazushige Obara. Reconstruction of a 2d seismic wavefield by seismic gradiometry. *Progress in Earth and Planetary Science*, 3(1):1–17, 2016. doi: 10.1186/s40645-016-0107-4.
- [154] Julien Mairal, Jean Ponce, Guillermo Sapiro, Andrew Zisserman, and Francis Bach. Supervised dictionary learning. *Advances in neural information processing systems*, 21, 2008. doi: 10.48550/arXiv.0809.3083.
- [155] Peter Makus and Christoph Sens-Schönfelder. Seismic - an open source python toolset to compute velocity changes from ambient seismic noise. *Seismica*, 3(1), 02 2024. doi: 10.26443/seismica.v3i1.1099.

- [156] Marielle Malfante, Mauro Dalla Mura, Jean-Philippe Métaxian, Jerome I Mars, Orlando Macedo, and Adolfo Inza. Machine learning for volcano-seismic signals: Challenges and perspectives. *IEEE Signal Processing Magazine*, 35(2):20–30, 2018. doi: 10.1109/MSP.2017.2779166.
- [157] Stéphane Mallat. *A wavelet tour of signal processing*. Elsevier, 1999. doi: 10.1016/B978-0-12-374370-1.X0001-8.
- [158] Andrea Manconi, Stefano Luigi Gariano, Velio Coviello, and Fausto Guzzetti. How many rainfall-induced landslides are detectable by a regional seismic monitoring network? In *Workshop on World Landslide Forum*, pages 161–168. Springer, 2017. doi: 10.1007/978-3-319-53487-9_18.
- [159] Eileen R Martin. A scalable algorithm for cross-correlations of compressed ambient seismic noise. In *SEG International Exposition and Annual Meeting*, page Do43S141R005. SEG, 2019. doi: 10.1190/segam2019-3216637.1.
- [160] Per-Gunnar Martinsson, Vladimir Rokhlin, and Mark Tygert. A randomized algorithm for the decomposition of matrices. *Applied and Computational Harmonic Analysis*, 30(1):47–68, 2011. doi: 10.1016/j.acha.2010.02.003.
- [161] Ian W McBrearty and Gregory C Beroza. Earthquake phase association with graph neural networks. *Bulletin of the Seismological Society of America*, 113(2):524–547, 2023. doi: 10.1785/0120220182.
- [162] Diego Melgar, Amanda M. Thomas, Valerie J. Sahakian, Andrew J. Meigs, Pieter E. Share, Harold J. Tobin, Timothy I. Melbourne, and Jill Elizabeth. The cascadia region earthquake science center strategic plan 2023-2027, 05 2024.
- [163] Dirk Merkel et al. Docker: lightweight linux containers for consistent development and deployment. *Linux j*, 239(2):2, 2014.
- [164] Yaolin Miao, Santiago Rabade, Meichen Liu, and Zack J Spica. Evaluating subsurface imaging performance with ocean-bottom distributed acoustic sensing and double beamforming. *Geophysical Journal International*, 242(2):ggaf190, 2025. doi: 10.1093/gji/ggaf190.
- [165] Alberto Michellini, Spina Cianetti, Sonja Gaviano, Carlo Giunchi, Dario Jozinovic, and Valentino Lauciani. Instance - the italian seismic dataset for machine learning. *Earth System Science Data*, 2021. doi: 10.5194/essd-2021-164.

- [166] Alistair Miles, John Kirkham, Martin Durant, James Bourbeau, Tarik Onalan, Joe Hamman, Zain Patel, shikharsg, Matthew Rocklin, raphael dussin, Vincent Schut, Elliott Sales de Andrade, Ryan Abernathey, Charles Noyes, sbalmer, pyup.io bot, Tommy Tran, Stephan Saalfeld, Justin Swaney, Josh Moore, Joe Jevnik, Jerome Kelleher, Jan Funke, George Sakkis, Chris Barnes, and Anderson Banihirwe. `zarr-developers/zarr-python: v2.4.0`, 2020.
- [167] Tigran Mkrtchyan, Krishnaveni Chitrapu, Vincent Garonne, Dmitry Litvintsev, Svenja Meyer, Paul Millar, Lea Morschel, Albert Rossi, and Marina Sahakyan. `dcache: Inter-disciplinary storage system`. In *EPJ Web of Conferences*, volume 251, page 02010. EDP Sciences, 2021. doi: 10.1051/epjconf/202125102010.
- [168] Sasmita Mohapatra, Weiming Yang, Zhengtang Yang, Chenxiao Wang, Jinxin Ma, Gary L. Pavlis, and Yinzhi Wang. Parallel seismic data processing performance with cloud-based storage. *arXiv preprint arXiv:2504.09075*, 2025. doi: 10.48550/arXiv.2504.09075.
- [169] Irene Molinari and Andrea Morelli. Epcrust: a reference crustal model for the european plate. *Geophysical Journal International*, 185(1):352–364, 2011. doi: 10.1111/j.1365-246X.2011.04940.x.
- [170] Seth C Moran. Seismicity associated with renewed dome building at mount st. helens, 2004–2005. *US Geological Survey professional paper*, 1750:27–60, 2008. doi: 10.3133/pp17502.
- [171] S Mostafa Mousavi and Gregory C Beroza. Deep-learning seismology. *Science*, 377(6607): eabm4470, 2022. doi: 10.1126/science.abm4470.
- [172] S Mostafa Mousavi, Yixiao Sheng, Weiqiang Zhu, and Gregory C Beroza. Stanford earthquake dataset (stead): A global data set of seismic signals for ai. *IEEE Access*, 7:179464–179476, 2019. doi: 10.1109/ACCESS.2019.2947848.
- [173] S Mostafa Mousavi, Weiqiang Zhu, Yixiao Sheng, and Gregory C Beroza. Cred: A deep residual network of convolutional and recurrent units for earthquake signal detection. *Scientific reports*, 9(1):10267, 2019. doi: 10.1038/s41598-019-45748-1.
- [174] S Mostafa Mousavi, William L Ellsworth, Weiqiang Zhu, Lindsay Y Chuang, and Gregory C Beroza. Earthquake transformer—an attentive deep-learning model for simultaneous earthquake detection and phase picking. *Nature communications*, 11(1):1–12, 2020. doi: 10.1038/s41467-020-17591-w.

- [175] Charles S Mueller. Earthquake catalogs for the usgs national seismic hazard maps. *Seismological Research Letters*, 90(1):251–261, 2019. doi: 10.1785/0220170108.
- [176] Jack B Muir and Zhongwen Zhan. Seismic wavefield reconstruction using a pre-conditioned wavelet–curvelet compressive sensing approach. *Geophysical Journal International*, 227(1):303–315, 2021. doi: 10.1093/gji/ggab222.
- [177] Jannes Münchmeyer. Pyocto: A high-throughput seismic phase associator. *Seismica*, 3(1), 2024. doi: 10.26443/seismica.v3i1.1130.
- [178] Jannes Münchmeyer, Dino Bindi, Ulf Leser, and Frederik Tilmann. Earthquake magnitude and location estimation from real time seismic waveforms with a transformer network. *Geophysical Journal International*, 226(2):1086–1104, 2021. doi: 10.1093/gji/ggab139.
- [179] Jannes Münchmeyer, Dino Bindi, Ulf Leser, and Frederik Tilmann. The transformer earthquake alerting model: A new versatile approach to earthquake early warning. *Geophysical Journal International*, 225(1):646–656, 2021. doi: 10.1093/gji/ggaa609.
- [180] Jannes Münchmeyer, Jack Woollam, Andreas Rietbrock, Frederik Tilmann, Dietrich Lange, Thomas Bornstein, Tobias Diehl, Carlo Giunchi, Florian Haslinger, Dario Jozinović, et al. Which picker fits my data? a quantitative evaluation of deep learning based seismic pickers. *Journal of Geophysical Research: Solid Earth*, 127(1):e2021JB023499, 2022. doi: 10.1029/2021JB023499.
- [181] Jannes Münchmeyer, Diego Molina-Ormazabal, David Marsan, Mickaël Langlais, Juan-Carlos Baez, Ben Heit, Diego González-Vidal, Marcos Moreno, Frederik Tilmann, Dietrich Lange, et al. Characterizing the atacama segment of the chile subduction margin (24 s–31 s) with 165,000 earthquakes. *Journal of Geophysical Research: Solid Earth*, 130(7):e2025JB031256, 2025. doi: 10.1029/2025JB031256.
- [182] Jannes Münchmeyer, Sophie Giffard-Roisin, Marielle Malfante, William Frank, Piero Poli, David Marsan, and Anne Socquet. Deep learning detects uncataloged low-frequency earthquakes across regions. *Seismica*, 3(1), 05 2024. doi: 10.26443/seismica.v3i1.1185.
- [183] T Nagata, K Nakai, Keigo Yamada, Yuji Saito, Taku Nonomura, Masayuki Kano, Shin-ichi Ito, and Hiromichi Nagao. Seismic wavefield reconstruction based on compressed sensing using data-driven reduced-order model. *Geophysical Journal International*, 233(1):33–50, 2023. doi: 10.1093/gji/ggac443.

- [184] Natural Resources Canada (NRCAN Canada). Canadian national seismograph network, 1975.
- [185] Avinash Nayak, Jonathan Ajo-Franklin, and Imperial Valley Dark Fiber Team. Distributed acoustic sensing using dark fiber for array detection of regional earthquakes. *Seismological Society of America*, 92(4):2441–2452, 2021. doi: 10.1785/0220200416.
- [186] Yiyu Ni, Shaowu Pan, Nathan Kutz, Bradley Paul Lipovsky, and Marine Denolle. Neural implicit compact representation to compress distributed acoustic sensing data. In *AGU Fall Meeting Abstracts*, volume 2022, pages OS25E–0963, 2022.
- [187] Yiyu Ni, Marine A. Denolle, Rob Fatland, Naomi Alterman, Bradley P. Lipovsky, and Friedrich Knuth. An object storage for distributed acoustic sensing. *Seismological Research Letters*, 95(1): 499–511, 10 2023. ISSN 0895-0695. doi: 10.1785/0220230172.
- [188] Yiyu Ni, Alexander Hutko, Francesca Skene, Marine Denolle, Stephen Malone, Paul Bodin, Renate Hartog, and Amy Wright. Curated pacific northwest ai-ready seismic dataset. *Seismica*, 2 (1), 2023. doi: 10.26443/seismica.v2i1.368.
- [189] Yiyu Ni, Joseph-Paul A Swinski, and Marine Denolle. Cloud-optimized asdf-h5 for seismology. *Authorea Preprints*, 2023. doi: 10.22541/essoar.168298674.44060085/v1.
- [190] Yiyu Ni, Marine A Denolle, Rob Fatland, Naomi Alterman, Bradley P Lipovsky, and Friedrich Knuth. An object storage for distributed acoustic sensing. *Seismological Research Letters*, 95(1): 499–511, 2024. doi: 10.1785/0220230172.
- [191] Yiyu Ni, Marine A Denolle, Qibin Shi, Bradley P Lipovsky, Shaowu Pan, and J Nathan Kutz. Wavefield reconstruction of distributed acoustic sensing: Lossy compression, wavefield separation, and edge computing. *Journal of Geophysical Research: Machine Learning and Computation*, 1(3): e2024JH000247, 2024. doi: 10.1029/2024JH000247.
- [192] Yiyu Ni, Marine A Denolle, Jannes Münchmeyer, Yinzhi Wang, Kuan-Fu Feng, Carlos Garcia Jurado Suarez, Amanda M Thomas, Chad Trabant, Alex Hamilton, and David Mencin. A review of cloud computing and storage in seismology. *Geophysical Journal International*, page ggaf322, 2025. doi: 10.1093/gji/ggaf322.
- [193] Yiyu Ni, Marine A Denolle, Amanda M Thomas, Alex Hamilton, Jannes Münchmeyer, Yinzhi Wang, Loïc Bachelot, Chad Trabant, and David Mencin. A global-scale database of seismic phases

- from cloud-based picking at petabyte scale. *Seismica*, 4(2), 09 2025. doi: 10.48550/arXiv.2505.18874.
- [194] Kiwamu Nishida, Ryota Takagi, and Akiko Takeo. Ambient noise multimode surface wave tomography. *Progress in Earth and Planetary Science*, 11(1):1–42, 2024. doi: 10.1186/s40645-023-00605-8.
- [195] Michael Norman, Vince Kellen, Shava Smallen, Brian DeMeulle, Shawn Strande, Ed Lazowska, Naomi Alterman, Rob Fatland, Sarah Stone, Amanda Tan, Katherine Yelick, Eric Van Dusen, and James Mitchell. Cloudbank: Managed services to simplify cloud access for computer science research and education. In *Practice and Experience in Advanced Research Computing 2021: Evolution Across All Dimensions*, PEARC ’21. Association for Computing Machinery, 2021. ISBN 9781450382922. doi: 10.1145/3437359.3465586.
- [196] SD Olinger and MA Denolle. Tracking the cracking: A holistic analysis of rapid ice shelf fracture using seismology, geodesy, and satellite imagery on the pine island glacier ice shelf, west antarctica. *Geophysical Research Letters*, 49(10):e2021GL097604, 2022. doi: 10.1029/2021GL097604.
- [197] Shaowu Pan, Steven L Brunton, and J Nathan Kutz. Neural implicit flow: a mesh-agnostic dimensionality reduction paradigm of spatio-temporal data. *Journal of Machine Learning Research*, 24(41):1–60, 2023. doi: 10.48550/arXiv.2204.03216.
- [198] Stavros Papadopoulos, Kushal Datta, Samuel Madden, and Timothy Mattson. The tiledb array data storage manager. *Proceedings of the VLDB Endowment*, 10(4):349–360, 2016. doi: 10.14778/3025111.3025117.
- [199] Yongsoo Park, Gregory C. Beroza, and William L. Ellsworth. Basement fault activation before larger earthquakes in oklahoma and kansas. *The Seismic Record*, 2(3):197–206, 08 2022. ISSN 2694-4006. doi: 10.1785/0320220020.
- [200] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703*, 2019. doi: 10.48550/arXiv.1912.01703.
- [201] Amandalynne Paullada, Inioluwa Deborah Raji, Emily M. Bender, Emily Denton, and Alex Hanna. Data and its (dis)contents: A survey of dataset development and use in machine learning research. *Patterns*, 2(11):100336, 2021. ISSN 2666-3899. doi: 10.1016/j.patter.2021.100336.

- [202] Andres Felipe Peña Castro, Brandon Schmandt, Michael G Baker, and Robert E Abbott. Tracking local sea ice extent in the beaufort sea using distributed acoustic sensing and machine learning. *The Seismic Record*, 3(3):200–209, 2023. doi: 10.1785/0320230019.
- [203] Thibaut Perol, Mykel J Gharbi, and Marine Denolle. Convolutional neural network for earthquake detection and location. *Science Advances*, 4(2):e1700578, 2018. doi: 10.1126/sciadv.1700578.
- [204] Daniel Peter, Dimitri Komatitsch, Yang Luo, Roland Martin, Nicolas Le Goff, Emanuele Casarotti, Pieyre Le Loher, Federica Magnoni, Qinya Liu, Céline Blitz, et al. Forward and adjoint simulations of seismic wave propagation on fully unstructured hexahedral meshes. *Geophysical Journal International*, 186(2):721–739, 2011. doi: 10.1111/j.1365-246X.2011.05044.x.
- [205] Mark D Petersen, Allison M Shumway, Peter M Powers, Charles S Mueller, Morgan P Moschetti, Arthur D Frankel, Sanaz Rezaeian, Daniel E McNamara, Nico Luco, Oliver S Boyd, et al. The 2018 update of the us national seismic hazard model: Overview of model and implications. *Earthquake Spectra*, 36(1):5–41, 2020. doi: 10.1177/8755293019878199.
- [206] N. Anders Petersson, Bjorn Sjogreen, Houjun Tang, and Ramesh Pankajakshan. geodynamic-s/sw4: Sw4, version 3.0, 09 2023. URL <https://doi.org/10.5281/zenodo.8322590>.
- [207] Paola Pierleoni, Roberto Concetti, Alberto Belli, Lorenzo Palma, Simone Marzorati, and Marco Esposito. A cloud-iot architecture for latency-aware localization in earthquake early warning. *Sensors*, 23(20), 2023. ISSN 1424-8220. doi: 10.3390/s23208431.
- [208] Andreas Plesch, Scott Marshall, and John Shaw. Scec community fault model (cfm), 09 2024.
- [209] Javier Quinteros, Jerry A. Carter, Jonathan Schaeffer, Chad Trabant, and Helle A. Pedersen. Exploring approaches for large data in seismology: User and data repository perspectives. *Seismological Research Letters*, 92(3):1531–1540, 01 2021. ISSN 0895-0695. doi: 10.1785/0220200390.
- [210] Javier Quinteros, Angelo Strollo, Peter L Evans, Winfried Hanka, Andres Heinloo, Susanne Hemmleb, Laura Hillmann, Karl-Heinz Jaeckel, Rainer Kind, Joachim Saul, et al. The geofon program in 2020. *Seismological Research Letters*, 92(3):1610–1622, 2021. doi: 10.1785/0220200415.
- [211] John M Rekoske, Morgan P Moschetti, and Eric M Thompson. Basin and site effects in the us pacific northwest estimated from small-magnitude earthquakes. *Bulletin of the Seismological Society of America*, 112(1):438–456, 2022. doi: 10.1785/0120210029.

- [212] John M Rekoske, Dave A May, and Alice-Agnes Gabriel. Reduced-order modelling for complex three-dimensional seismic wave propagation. *Geophysical Journal International*, 241(1):526–548, 2025. doi: 10.1093/gji/ggafo49.
- [213] Zhi-Ming Ren, Xue Dai, and Qian-Zong Bao. Source wavefield reconstruction based on an implicit staggered-grid finite-difference operator for seismic imaging. *Petroleum Science*, 19(5): 2095–2106, 2022. doi: 10.1016/j.petsci.2022.05.008.
- [214] Lise Retailleau, Jean-Marie Saurel, Weiqiang Zhu, Claudio Satriano, Gregory C Beroza, Simon Issartel, Patrice Boissier, OVPF Team, OVSM Team, et al. A wrapper to use a machine-learning-based algorithm for earthquake monitoring. *Seismological Research Letters*, 93(3):1673–1682, 2022. doi: 10.1785/0220210279.
- [215] Russ Rew and Glenn Davis. Netcdf: an interface for scientific data access. *IEEE computer graphics and applications*, 10(4):76–82, 1990. doi: 10.1109/38.56302.
- [216] M Reza, F Faraji, and JN Kutz. Data-driven inference of high-dimensional spatiotemporal state of plasma systems. *Journal of Applied Physics*, 136(18), 2024. doi: 10.1063/5.0230056.
- [217] Niles Ritter and Mike Ruth. The geotiff data interchange standard for raster geographic images. *International Journal of Remote Sensing*, 18(7):1637–1647, 1997. doi: 10.1080/014311697218340.
- [218] Michael H Ritzwoller, Fan-Chi Lin, and Weisen Shen. Ambient noise tomography with a large seismic array. *Comptes Rendus Geoscience*, 343(8-9):558–570, 2011. doi: 10.1016/j.crte.2011.03.007.
- [219] Stefano Riva, Carolina Introini, Antonio Cammi, and J Nathan Kutz. Robust state estimation from partial out-core measurements with shallow recurrent decoder for nuclear reactors. *Progress in Nuclear Energy*, 189:105928, 2025. doi: 10.1016/j.pnucene.2025.105928.
- [220] Matthew Rocklin. Dask: Parallel computation with blocked algorithms and task scheduling. In *Proceedings of the 14th Python in Science Conference*, pages 130–136. SciPy, 2015. doi: 10.25080/Majora-7b98e3ed-013.
- [221] Verónica Rodríguez Tribaldos and Jonathan B Ajo-Franklin. Aquifer monitoring using ambient seismic noise recorded with distributed acoustic sensing (das) deployed on dark fiber. *Journal of Geophysical Research: Solid Earth*, 126(4):e2020JB021004, 2021. doi: 10.1029/2020JB021004.
- [222] Garry Rogers and Herb Dragert. Episodic tremor and slip on the cascadia subduction zone: The chatter of silent slip. *science*, 300(5627):1942–1943, 2003. doi: 10.1126/science.1084783.

- [223] Barbara Romanowicz, Richard Allen, Knute Brekke, Li-Wei Chen, Yuancong Gou, Ivan Henson, Julien Marty, Doug Neuhauser, Brian Pardini, Taka'aki Taira, et al. Seafoam: A year-long das deployment in monterey bay, california. *Seismological Research Letters*, 94(5):2348–2359, 2023. doi: 10.1785/0220230047.
- [224] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. doi: 10.48550/arXiv.1505.04597.
- [225] Zachary E Ross, Men-Andrin Meier, Egill Hauksson, and Thomas H Heaton. Generalized seismic phase detection with deep learning. *Bulletin of the Seismological Society of America*, 108(5A):2894–2901, 2018. doi: 10.1785/0120180080.
- [226] Zachary E Ross, Daniel T Trugman, Egill Hauksson, and Peter M Shearer. Searching for hidden earthquakes in southern california. *Science*, 364(6442):767–771, 2019. doi: 10.1126/science.aaw6888.
- [227] Zachary E Ross, Yisong Yue, Men-Andrin Meier, Egill Hauksson, and Thomas H Heaton. Phaselink: A deep learning approach to seismic phase association. *Journal of Geophysical Research: Solid Earth*, 124(1):856–869, 2019. doi: 10.1029/2018JB016674.
- [228] Zachary E Ross, Men-Andrin Meier, Egill Hauksson, and Thomas H Heaton. P-wave arrival picking and first-motion polarity determination with deep learning. *Journal of Geophysical Research: Solid Earth*, 125(4):e2019JB018663, 2020. doi: 10.1029/2017JB015251.
- [229] Rutgers University. Ocean observatories initiative, 2013.
- [230] Omar M Saad, Yunfeng Chen, Daniel Siervo, Fangxue Zhang, Alexandras Savvaidis, Guochin Dino Huang, Nadine Igonin, Sergey Fomel, and Yangkang Chen. Eqcct: A production-ready earthquake detection and phase picking method using the compact convolutional transformer. *IEEE Transactions on Geoscience and Remote Sensing*, 2023. doi: 10.1109/TGRS.2023.3319440.
- [231] Korbinian Sager, Christian Boehm, Laura Ermert, Lion Krischer, and Andreas Fichtner. Global-scale full-waveform ambient noise inversion. *Journal of Geophysical Research: Solid Earth*, 125(4):e2019JB018644, 2020. doi: 10.1029/2019JB018644.
- [232] SCEDC. Southern California Earthquake Data Center, 2013. Type: dataset.

- [233] J. F. Schmitt, T. Clements, N. Wang, K. B. Olsen, and M. A. Denolle. Ground motion prediction using ambient seismic noise on a large-n array in the la basin. In *SCEC Annual Meeting*, 2020. URL <https://central.scec.org/meetings/2020/am/poster/212>.
- [234] Julian Schmitt, Timothy Clements, and Marine Denolle. Julians42/c4-project.jl: C4 project v0.1.0, 04 2025.
- [235] Danijel Schorlemmer, Fabian Euchner, Philipp Kästli, Joachim Saul, QuakeML Working Group, et al. Quakeml: status of the xml-based seismological data exchange format. *Annals of Geophysics*, 54(1), 2011. doi: 10.4401/ag-4874.
- [236] Hannu Seebeck, Russ Van Dissen, Nicola Litchfield, Philip M Barnes, Andrew Nicol, Robert Langridge, David JA Barrell, Pilar Villamor, Susan Ellis, Mark Rattenbury, et al. The new zealand community fault model—version 1.0: an improved geological foundation for seismic hazard modelling. *New Zealand journal of geology and geophysics*, 67(2):209–229, 2024. doi: 10.1080/00288306.2023.2181362.
- [237] Christoph Sens-Schönfelder and Ulrich Wegler. Passive image interferometry and seasonal variations of seismic velocities at merapi volcano, indonesia. *Geophysical research letters*, 33(21), 2006. doi: 10.1029/2006GL027797.
- [238] Christoph Sens-Schönfelder and Ulrich Wegler. Passive image interferometry for monitoring crustal changes with ambient seismic noise. *Comptes Rendus Geoscience*, 343(8):639–651, 2011. ISSN 1631-0713. doi: 10.1016/j.crte.2011.02.005.
- [239] Léonard Seydoux, Randall Balestrieri, Piero Poli, Maarten de Hoop, Michel Campillo, and Richard Baraniuk. Clustering earthquake signals and background noises in continuous seismic data with unsupervised deep learning. *Nature communications*, 11(1):3972, 2020. doi: 10.1038/s41467-020-17841-x.
- [240] Nikolai M Shapiro, Michel Campillo, Laurent Stehly, and Michael H Ritzwoller. High-resolution surface-wave tomography from ambient seismic noise. *Science*, 307(5715):1615–1618, 2005. doi: 10.1126/science.1108339.
- [241] Nikolai M. Shapiro, Michel Campillo, Laurent Stehly, and Michael H. Ritzwoller. High-resolution surface-wave tomography from ambient seismic noise. *Science*, 307(5715):1615–1618, 2005. doi: 10.1126/science.1108339.

- [242] John H Shaw, Andreas Plesch, Carl Tape, M Peter Suess, Thomas H Jordan, Geoffrey Ely, Egill Hauksson, Jeroen Tromp, Toshiro Tanimoto, Robert Graves, et al. Unified structural representation of the southern california crust and upper mantle. *Earth and Planetary Science Letters*, 415:1–15, 2015. doi: 10.1016/j.epsl.2015.01.016.
- [243] David Shean, JP Swinski, Ben Smith, Tyler Sutterley, Scott Henderson, Carlos Ugarte, Eric Lidwa, and Thomas Neumann. Sliderule: Enabling rapid, scalable, open science for the nasa icesat-2 mission and beyond. *Journal of Open Source Software*, 8(81):4982, 2023. doi: 10.21105/joss.04982.
- [244] Peter M Shearer. *Introduction to seismology*. Cambridge university press, 2019.
- [245] Zhichao Shen and Wenbo Wu. Ocean bottom distributed acoustic sensing for oceanic seismicity detection and seismic ocean thermometry. *Journal of Geophysical Research: Solid Earth*, 129(3):e2023JB027799, 2024. doi: 10.1029/2023JB027799.
- [246] Qibin Shi and Marine A Denolle. Improved observations of deep earthquake ruptures using machine learning. *Journal of Geophysical Research: Solid Earth*, 128(12):e2023JB027334, 2023. doi: 10.1029/2023JB027334.
- [247] Qibin Shi, Marine A. Denolle, Yiyu Ni, Ethan F. Williams, and Nan You. Denoising offshore distributed acoustic sensing using masked auto-encoders to enhance earthquake detection. *ESS Open Archive*, 2024. doi: 10.22541/essoar.172252976.63050899/v1.
- [248] Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. *Advances in neural information processing systems*, 33:7462–7473, 2020. doi: 10.48550/arXiv.2006.09661.
- [249] Patrick Small, David Gill, Philip J Maechling, Ricardo Taborda, Scott Callaghan, Thomas H Jordan, Kim B Olsen, Geoffrey P Ely, and Christine Goulet. The sceec unified community velocity model software framework. *Seismological Research Letters*, 88(6):1539–1552, 2017. doi: 10.1785/0220170082.
- [250] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. doi: 10.48550/arXiv.2011.13456.
- [251] Andreas S Spanias, Stefan B Jonsson, and Samuel D Stearns. Transform methods for seismic data

- compression. *IEEE Transactions on Geoscience and Remote Sensing*, 29(3):407–416, 1991. doi: 10.1109/36.79431.
- [252] Zack J Spica, Kiwamu Nishida, Takeshi Akuhara, François Pétrélis, Masanao Shinohara, and Tomoaki Yamada. Marine sediment characterized by ocean-bottom fiber-optic seismology. *Geophysical Research Letters*, 47(16):e2020GL088360, 2020. doi: 10.1029/2020GL088360.
- [253] Zack J Spica, Mathieu Perton, Eileen R Martin, Gregory C Beroza, and Biondo Biondi. Urban seismic site characterization by fiber-optic seismology. *Journal of Geophysical Research: Solid Earth*, 125(3):e2019JB018656, 2020. doi: 10.1029/2019JB018656.
- [254] Zack J Spica, Jonathan Ajo-Franklin, Gregory C Beroza, Biondo Biondi, Feng Cheng, Beatriz Gaite, Bin Luo, Eileen Martin, Junzhu Shen, Clifford Thurber, et al. Pubdas: A public distributed acoustic sensing datasets repository for geosciences. *Seismological Society of America*, 94(2A): 983–998, 2023. doi: 10.1785/0220220279.
- [255] William J Stephenson, Michael W Asten, Jack K Odum, and Arthur D Frankel. Shear-wave velocity in the seattle basin to 2 km depth characterized with the krspac microtremor array method: Insights for urban basin-scale imaging. *Seismological Research Letters*, 90(3):1230–1242, 2019. doi: 10.1785/0220180194.
- [256] Ian Stone, Erin A Wirth, and Arthur D Frankel. Topographic response to simulated m w 6.5–7.0 earthquakes on the seattle fault. *Bulletin of the Seismological Society of America*, 112(3):1436–1462, 2022. doi: 10.1785/0120210269.
- [257] Gilbert Strang. The discrete cosine transform. *SIAM review*, 41(1):135–147, 1999. doi: 10.1137/S0036144598336745.
- [258] Hongyu Sun and Laurent Demanet. Extrapolated full-waveform inversion with deep learning. *Geophysics*, 85(3):R275–R288, 2020. doi: 10.48550/arXiv.1909.11536.
- [259] Hongyu Sun, Zachary E Ross, Weiqiang Zhu, and Kamyar Azizzadenesheli. Phase neural operator for multi-station picking of seismic arrivals. *Geophysical Research Letters*, 50(24):e2023GL106434, 2023. doi: 10.1029/2023GL106434.
- [260] Hongyu Sun, Zachary E. Ross, Weiqiang Zhu, and Kamyar Azizzadenesheli. Phase neural operator for multi-station picking of seismic arrivals. *Geophysical Research Letters*, 50(24): e2023GL106434, 2023. doi: 10.1029/2023GL106434.

- [261] Wei-Fang Sun, Sheng-Yan Pan, Chun-Ming Huang, Zhuo-Kang Guan, I-Chin Yen, Chun-Wei Ho, Tsung-Chih Chi, Chin-Shang Ku, Bor-Shouh Huang, Ching-Chou Fu, et al. Deep learning-based earthquake catalog reveals the seismogenic structures of the 2022 mw 6.9 chihshang earthquake sequence. *Terrestrial, Atmospheric and Oceanic Sciences*, 35(1):5, 2024. doi: 10.1007/s44195-024-00063-9.
- [262] Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in Neural Information Processing Systems*, 33:7537–7547, 2020. doi: 10.48550/arXiv.2006.10739.
- [263] The HDF Group. Hierarchical data format, version 5, 1997-2023. <https://www.hdfgroup.org/HDF5/>.
- [264] Mika Thompson, Erin A Wirth, Arthur D Frankel, J Renate Hartog, and John E Vidale. Basin amplification effects in the puget lowland, washington, from strong-motion recordings and 3d simulations. *Bulletin of the Seismological Society of America*, 110(2):534–555, 2020. doi: 10.1785/0120190211.
- [265] Matteo Tomasetto, Jan P Williams, Francesco Braghin, Andrea Manzoni, and J Nathan Kutz. Reduced order modeling with shallow recurrent decoder networks. *Nature Communications*, 16(1):10260, 2025. doi: 10.1038/s41467-025-65126-y.
- [266] T Tonegawa, E Araki, H Matsumoto, T Kimura, K Obana, G Fujie, R Arai, K Shiraishi, M Nakano, Y Nakamura, et al. Extraction of p wave from ambient seafloor noise observed by distributed acoustic sensing. *Geophysical Research Letters*, 49(4):e2022GL098162, 2022. doi: 10.1029/2022GL098162.
- [267] Ivana Tošić and Pascal Frossard. Dictionary learning. *IEEE Signal Processing Magazine*, 28(2): 27–38, 2011. doi: 10.1109/MSP.2010.939537.
- [268] Sara Touhami, Filippo Gatti, Fernando Lopez-Caballero, Régis Cottureau, Lúcio de Abreu Corrêa, Ludovic Aubry, and Didier Clouteau. Sem3d: A 3d high-fidelity numerical earthquake simulator for broadband (0–10 hz) seismic response prediction at a regional scale. *Geosciences*, 12(3):112, 2022. doi: 10.3390/geosciences12030112.

- [269] Andrew Trafford, Robert Ellwood, Loris Wacquier, Alastair Godfrey, Chris Minto, Mark Coughlan, and Shane Donohue. Distributed acoustic sensing for active offshore shear wave profiling. *Scientific Reports*, 12(1):9691, 2022. doi: 10.1038/s41598-022-13962-z.
- [270] Takahiko Uchide. Focal mechanisms of small earthquakes beneath the japanese islands based on first-motion polarities picked using deep learning. *Geophysical Journal International*, 223(3): 1658–1671, 2020. doi: 10.1093/gji/ggaa401.
- [271] Leonardo Uieda, Dongdong Tian, Wei Ji Leong, Liam Toney, William Schlitzer, Michael Grund, Tyler Newton, Malte Ziebarth, Meghan Jones, and Paul Wessel. PyGMT: A Python interface for the Generic Mapping Tools, 2021.
- [272] University of Oregon. Pacific northwest seismic network - university of oregon, 1990.
- [273] University of Washington. Pacific northwest seismic network - university of washington, 1963.
- [274] U.S. Geological Survey. Advanced National Seismic System (ANSS) Comprehensive Catalog, 2017. U.S. Geological Survey.
- [275] Tokuji Utsu. A statistical study on the occurrence of aftershocks. *Geophys. Mag.*, 30:521–605, 1961.
- [276] Andrew P Valentine and Jeannot Trampert. Data space reduction, quality assessment and searching of seismograms: autoencoder networks for waveform data. *Geophysical Journal International*, 189(2):1183–1202, 2012. doi: 10.1111/j.1365-246X.2012.05429.x.
- [277] Martijn Van den Ende, Itzhak Lior, Jean-Paul Ampuero, Anthony Sladen, André Ferrari, and Cédric Richard. A self-supervised deep learning approach for blind denoising and waveform coherence enhancement in distributed acoustic sensing data. *IEEE Transactions on Neural Networks and Learning Systems*, 34(7):3371–3384, 2021. doi: 10.1109/TNNLS.2021.3132832.
- [278] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. doi: 10.48550/arXiv.1706.03762.
- [279] Sergi Ventosa, Martin Schimmel, and Eleonore Stutzmann. Towards the processing of large data volumes with phase cross-correlation. *Seismological Research Letters*, 90(4):1663–1669, 2019. doi: 10.1785/0220190022.

- [280] Loïc Viens, Mathieu Pertou, Zack J Spica, Kiwamu Nishida, Tomoaki Yamada, and Masanao Shinohara. Understanding surface wave modal content for high-resolution imaging of submarine sediments with distributed acoustic sensing. *Geophysical Journal International*, 232(3):1668–1683, 2023. doi: 10.1093/gji/ggac420.
- [281] John D Villasenor, RA Ergas, and PL Donoho. Seismic data compression using high-dimensional wavelet transforms. In *Proceedings of Data Compression Conference-DCC’96*, pages 396–405. IEEE, 1996. doi: 10.1109/DCC.1996.488345.
- [282] Pauli Virtanen, Ralf Gommers, Travis E Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, et al. Scipy 1.0: fundamental algorithms for scientific computing in python. *Nature methods*, 17(3):261–272, 2020. doi: 10.1038/s41592-019-0686-2.
- [283] Ole Henrik Waagaard, Erlend Rønnekleiv, Aksel Haukanes, Frantz Stabo-Eeg, Dag Thingbø, Stig Forbord, Svein Erik Aasen, and Jan Kristoffer Brenne. Real-time low noise distributed acoustic sensing in 171 km low loss fiber. *OSA continuum*, 4(2):688–701, 2021. doi: 10.1364/OSAC.408761.
- [284] Jacob I. Walter, Paul Ogwari, Andrew Thiel, Fernando Ferrer, and Isaac Woelfel. easyquake: Putting machine learning to work for your regional seismic network or local earthquake study. *Seismological Research Letters*, 92(1):555–563, 11 2020. ISSN 0895-0695. doi: 10.1785/0220200226.
- [285] Jacob I Walter, Paul Ogwari, Andrew Thiel, Fernando Ferrer, and Isaac Woelfel. easyquake: Putting machine learning to work for your regional seismic network or local earthquake study. *Seismological Research Letters*, 92(1):555–563, 2021. doi: 10.1785/0220200226.
- [286] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018. doi: 10.48550/arXiv.1804.07461.
- [287] Ruijia Wang, Brandon Schmandt, Monique Holt, and Keith Koper. Advancing local distance discrimination of explosions and earthquakes with joint p/s and ml-mc classification. *Geophysical Research Letters*, 48(23):e2021GL095721, 2021. doi: 10.1029/2021GL095721.
- [288] Weitao Wang, Baoshan Wang, and Xiufen Zheng. Public cloud computing for seismological research: Calculating large-scale noise cross-correlations using aliyun. *Earthquake Science*, 31(5-6): 227–233, 2018. ISSN 1674-4519. doi: 10.29382/eqs-2018-0227-2.

- [289] Xin Wang, Zhongwen Zhan, Ethan F Williams, Miguel González Herráez, Hugo Fidalgo Martins, and Martin Karrenbach. Ground vibrations recorded by fiber-optic cables reveal traffic response to covid-19 lockdown measures in pasadena, california. *Communications Earth & Environment*, 2(1):160, 2021. doi: 10.1038/s43247-021-00234-3.
- [290] Xinghao Wang, Feng Liu, Rui Su, Zhihui Wang, Lei Bai, and Wanli Ouyang. Seismollm: Advancing seismic monitoring via cross-modal transfer with pre-trained large language model. *arXiv preprint arXiv:2502.19960*, 2025. doi: 10.48550/arXiv.2502.19960.
- [291] Yinzhi Wang, Gary L Pavlis, Weiming Yang, and Jinxin Ma. Mspass: A data management and processing framework for seismology. *Seismological Research Letters*, 93(1):426–434, 2022. doi: 10.1785/0220210182.
- [292] Aaron G Wech and Noel M Bartlow. Slip rate and tremor genesis in cascadia. *Geophysical Research Letters*, 41(2):392–398, 2014. doi: 10.1002/2013GL058607.
- [293] Aaron G Wech, Kenneth C Creager, Heidi Houston, and John E Vidale. An earthquake-like magnitude-frequency distribution of slow slip in northern cascadia. *Geophysical Research Letters*, 37(22), 2010. doi: 10.1029/2010GL044881.
- [294] Jon Weers, Arlene Anderson, and Nicole Taverna. The geothermal data repository: Ten years of supporting the geothermal industry with open access to geothermal data. Technical report, National Renewable Energy Lab.(NREL), Golden, CO (United States), 2022.
- [295] Kathleen West, Fabian Lehmann, Vasilis Bountris, Ulf Leser, Yehia Elkhatib, and Lauritz Thamsen. Exploring the potential of carbon-aware execution for scientific workflows. *arXiv preprint arXiv:2503.13705*, 2025. doi: 10.48550/arXiv.2503.13705.
- [296] Malcolm CA White, Zhendong Zhang, Tong Bai, Hongrui Qiu, Hilary Chang, and Nori Nakata. Hdfseis: A storage and input/output solution for big multidimensional time series data from environmental sensors. *Geophysics*, 88(3):F29–F38, 2023. doi: 10.1190/geo2022-0448.1.
- [297] Stefan Wiemer and Max Wyss. Minimum magnitude of completeness in earthquake catalogs: Examples from alaska, the western united states, and japan. *Bulletin of the Seismological Society of America*, 90(4):859–869, 2000. doi: 10.1785/0119990114.
- [298] W. Wilcock and the Ocean Observatories Initiative. Rapid: A community test of distributed acoustic sensing on the ocean observatories initiative regional cabled array, 2023.

- [299] William SD Wilcock, Shima Abadi, and Bradley P Lipovsky. Distributed acoustic sensing recordings of low-frequency whale calls and ship noise offshore central oregon. *JASA Express Letters*, 3(2):026002, 2023. doi: 10.1121/10.0017104.
- [300] Mark D Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E Bourne, et al. The fair guiding principles for scientific data management and stewardship. *Scientific data*, 3(1):1–9, 2016. doi: 10.1038/sdata.2016.18.
- [301] Ethan F Williams, María R Fernández-Ruiz, Regina Magalhaes, Roel Vanthillo, Zhongwen Zhan, Miguel González-Herráez, and Hugo F Martins. Distributed sensing of microseisms and teleseisms with submarine dark fibers. *Nature communications*, 10(1):5778, 2019. doi: 10.1038/s41467-019-13262-7.
- [302] Ethan F Williams, Zhongwen Zhan, Hugo F Martins, María R Fernández-Ruiz, Sonia Martín-López, Miguel González-Herráez, and Jörn Callies. Surface gravity wave interferometry and ocean current monitoring with ocean-bottom das. *Journal of Geophysical Research: Oceans*, 127(5):e2021JC018375, 2022. doi: 10.1029/2021JC018375.
- [303] Jan P Williams, Olivia Zahn, and J Nathan Kutz. Sensing with shallow recurrent decoder networks. *arXiv preprint arXiv:2301.12011*, 2023. doi: 10.48550/arXiv.2301.12011.
- [304] Philipp A. Witte, Mathias Louboutin, Henryk Modzelewski, Charles Jones, James Selvage, and Felix J. Herrmann. An event-driven approach to serverless seismic imaging in the cloud. *IEEE Transactions on Parallel and Distributed Systems*, 31(9):2032–2049, 2020. doi: 10.1109/TPDS.2020.2982626.
- [305] Robert C Witter, Harvey M Kelsey, and Eileen Hemphill-Haley. Great cascadia earthquakes and tsunamis of the past 6700 years, coquille river estuary, southern coastal oregon. *Geological Society of America Bulletin*, 115(10):1289–1306, 2003. doi: 10.1130/B25189.1.
- [306] Jack Woollam, Andreas Rietbrock, Angel Bueno, and Silvio De Angelis. Convolutional Neural Network for Seismic Phase Classification, Performance Demonstration over a Local Seismic Network. *Seismological Research Letters*, 90(2A):491–502, 2019. ISSN 0895-0695, 1938-2057. doi: 10.1785/0220180312.
- [307] Jack Woollam, Jannes Münchmeyer, Frederik Tilmann, Andreas Rietbrock, Dietrich Lange, Thomas Bornstein, Tobias Diehl, Carlo Giunchi, Florian Haslinger, Dario Jozinović, et al. Seis-

- bench—a toolbox for machine learning in seismology. *Seismological Research Letters*, 93(3): 1695–1709, 2022. doi: 10.1785/0220210324.
- [308] Yih-Min Wu and Hiroo Kanamori. Rapid assessment of damaging potential of earthquakes in Taiwan from the beginning of p waves. *Bulletin of the Seismological Society of America*, 95(3): 1181–1185, 2005. doi: 10.1785/0120040193.
- [309] Andreas Wuestefeld, Zack J Spica, Kasey Aderhold, Hsin-Hua Huang, Kuo-Fong Ma, Voon Hui Lai, Meghan Miller, Lena Urmantseva, Daniel Zapf, Daniel C Bowden, et al. The global das month of february 2023. *Seismological Research Letters*, 95(3):1569–1577, 2024. doi: 10.1785/0220230180.
- [310] Yan Yan, Yifei Cui, Xin Tian, Sheng Hu, Jian Guo, Ziang Wang, Shuyao Yin, and Liufeng Liao. Seismic signal recognition and interpretation of the 2019 “7.23” shuicheng landslide by seismogram stations. *Landslides*, 17(5):1191–1206, 2020. doi: 10.1007/s10346-020-01358-x.
- [311] Liuqing Yang, Sergey Fomel, Shoudong Wang, Xiaohong Chen, Yunfeng Chen, and Yangkang Chen. Slknet: An attention-based deep-learning framework for downhole distributed acoustic sensing data denoising. *Geophysics*, 88(6):WC69–WC89, 2023. doi: 10.1190/geo2022-0724.1.
- [312] Yan Yang, Angela F Gao, Jorge C Castellanos, Zachary E Ross, Kamyar Azizzadenesheli, and Robert W Clayton. Seismic wave propagation and inversion with neural operators. *The Seismic Record*, 1(3):126–134, 2021. doi: 10.1785/0320210026.
- [313] Yan Yang, James W Atterholt, Zhichao Shen, Jack B Muir, Ethan F Williams, and Zhongwen Zhan. Sub-kilometer correlation between near-surface structure and ground motion measured with distributed acoustic sensing. *Geophysical Research Letters*, 49(1):e2021GL096503, 2022. doi: 10.1029/2021GL096503.
- [314] Yan Yang, Zhongwen Zhan, Zhichao Shen, and James Atterholt. Fault zone imaging with distributed acoustic sensing: Surface-to-surface wave scattering. *Journal of Geophysical Research: Solid Earth*, 127(6):e2022JB024329, 2022. doi: 10.1029/2022JB024329. e2022JB024329 2022JB024329.
- [315] Yingjie Yang and Michael H Ritzwoller. Characteristics of ambient seismic noise as a source for surface wave tomography. *Geochemistry, Geophysics, Geosystems*, 9(2), 2008. doi: 10.1029/2007GC001814.

- [316] William Luther Yeck, John M Patton, Zachary E Ross, Gavin P Hayes, Michelle R Guy, Nick B Ambruz, David R Shelly, Harley M Benz, and Paul S Earle. Leveraging deep learning in global 24/7 real-time earthquake monitoring at the national earthquake information center. *Seismological Society of America*, 92(1):469–480, 2021. doi: 10.1785/0220200178.
- [317] Alexey Yermakov, David Zoro, Mars Liyao Gao, and J Nathan Kutz. T-shred: Symbolic regression for regularization and model discovery with transformer shallow recurrent decoders. *arXiv preprint arXiv:2506.15881*, 2025. doi: 10.48550/arXiv.2506.15881.
- [318] Jiuxun Yin, Marcelo A Soto, Jaime Ramírez, Valey Kamalov, Weiqiang Zhu, Allen Husker, and Zhongwen Zhan. Real-data testing of distributed acoustic sensing for offshore earthquake early warning. *The Seismic Record*, 3(4):269–277, 2023. doi: 10.1785/0320230018.
- [319] Andy B Yoo, Morris A Jette, and Mark Grondona. Slurm: Simple linux utility for resource management. In *Workshop on job scheduling strategies for parallel processing*, pages 44–60. Springer, 2003. doi: 10.1007/10968987_3.
- [320] Clara E Yoon, Elizabeth S Cochran, Elizabeth A Vanacore, Victor Huerfano, Gisela Báez-Sánchez, John D Wilding, and Jonathan Smith. A detailed view of the 2020–2023 southwestern puerto rico seismic sequence with deep learning. *Bulletin of the Seismological Society of America*, 113(6): 2377–2415, 2023. doi: 10.1785/0120220229.
- [321] Ellen Yu, Aparna Bhaskaran, Shang-Lin Chen, Zachary E Ross, Egill Hauksson, and Robert W Clayton. Southern california earthquake data now available in the aws cloud. *Seismological Research Letters*, 92(5):3238–3247, 2021. doi: 10.1785/0220210039.
- [322] Ellen Yu, Aparna Bhaskaran, Shang-Lin Chen, Zachary E. Ross, Egill Hauksson, and Robert W. Clayton. Southern california earthquake data now available in the aws cloud. *Seismological Research Letters*, 92(5):3238–3247, 06 2021. ISSN 0895-0695. doi: 10.1785/0220210039.
- [323] Congcong Yuan, Yiyu Ni, Youzuo Lin, and Marine Denolle. Better together: Ensemble learning for earthquake detection and phase picking. *IEEE Transactions on Geoscience and Remote Sensing*, 2023. doi: 10.1109/TGRS.2023.3320148.
- [324] Zongsheng Yue, Jianyi Wang, and Chen Change Loy. Resshift: Efficient diffusion model for image super-resolution by residual shifting. *Advances in neural information processing systems*, 36: 13294–13307, 2023. doi: 10.48550/arXiv.2307.12348.

- [325] Emily E Zawacki, Rebecca Bendick, and Robert L Woodward. Advancing geophysics: Iris and unavco merge to form earthscope consortium, 2023.
- [326] Xiangfang Zeng, Clifford H Thurber, Herbert F Wang, Dante Fratta, and Kurt L Feigl. High-resolution shallow structure at brady hot springs using ambient noise tomography (ant) on a trenched distributed acoustic sensing (das) array. *Distributed Acoustic Sensing in Geophysics: Methods and Applications*, pages 101–110, 2021. doi: 10.1002/9781119521808.ch8.
- [327] Zhongwen Zhan. Distributed acoustic sensing turns fiber-optic cables into sensitive seismic antennas. *Seismological Research Letters*, 91(1):1–15, 2020. doi: 10.1785/0220190112.
- [328] Haijiang Zhang and Clifford Thurber. Development and applications of double-difference seismic tomography. *pure and applied geophysics*, 163(2):373–403, 2006. doi: 10.1007/s00024-005-0021-y.
- [329] Haijiang Zhang and Clifford H Thurber. Double-difference tomography: The method and its application to the hayward fault, california. *Bulletin of the Seismological Society of America*, 93(5): 1875–1889, 2003. doi: 10.1785/0120020190.
- [330] Miao Zhang, Min Liu, Tian Feng, Ruijia Wang, and Weiqiang Zhu. Loc-flow: An end-to-end machine learning-based high-precision earthquake location workflow. *Seismological Society of America*, 93(5):2426–2438, 2022. doi: 10.1785/0220220019.
- [331] Xingyue Zhang, Yuxuan Bao, Mars Liyao Gao, and J Nathan Kutz. Sendai: A hierarchical sparse-measurement, efficient data assimilation framework. *arXiv preprint arXiv:2601.21664*, 2026. doi: 10.48550/arXiv.2601.21664.
- [332] Xiong Zhang and Miao Zhang. Universal neural networks for real-time earthquake early warning trained with generalized earthquakes. *Communications Earth & Environment*, 5(1):528, 2024. doi: 10.1038/s43247-024-01718-8.
- [333] Yayun Zhang, Min Xu, Zhuo Xiao, Yong Zhou, Chuanhai Yu, Jian Lin, Hongfeng Yang, and Xuelin Qiu. Ambient noise interferometry using ocean bottom seismometer data from active source experiments conducted in the southernmost mariana trench. *Journal of Geophysical Research: Solid Earth*, 129(3):e2023JB027043, 2024. doi: 10.1029/2023JB027043.
- [334] YX Zhao, Y Li, and N Wu. Data augmentation and its application in distributed acoustic sensing data denoising. *Geophysical Journal International*, 228(1):119–133, 2022. doi: 10.1093/gji/ggab345.

- [335] Yiyuan Zhong and Yen Joe Tan. Deep-learning-based phase picking for volcano-tectonic and long-period earthquakes. *Geophysical Research Letters*, 51(12):e2024GL108438, 2024. doi: 10.1029/2024GL108438.
- [336] Junwei Zhou, Qian Wei, Chao Wu, and Guangzhong Sun. A high performance computing method for noise cross-correlation functions of seismic data. In *2021 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)*, pages 1179–1182. IEEE, 2021. doi: 10.1109/ISPA-BDCloud-SocialCom-SustainCom52081.2021.00162.
- [337] Weiqiang Zhu and Gregory C Beroza. Phasenet: A deep-neural-network-based seismic arrival-time picking method. *Geophysical Journal International*, 216(1):261–273, 2019. doi: 10.1093/gji/ggy423.
- [338] Weiqiang Zhu, S Mostafa Mousavi, and Gregory C Beroza. Seismic signal denoising and decomposition using deep neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, 57(11):9476–9488, 2019. doi: 10.1109/TGRS.2019.2926772.
- [339] Weiqiang Zhu, S Mostafa Mousavi, and Gregory C Beroza. Seismic signal augmentation to improve generalization of deep neural networks. In *Advances in geophysics*, volume 61, pages 151–177. Elsevier, 2020. doi: 10.1016/bs.agph.2020.07.003.
- [340] Weiqiang Zhu, Ian W McBrearty, S Mostafa Mousavi, William L Ellsworth, and Gregory C Beroza. Earthquake phase association using a bayesian gaussian mixture model. *Journal of Geophysical Research: Solid Earth*, 127(5):e2021JB023249, 2022. doi: 10.1029/2021JB023249.
- [341] Weiqiang Zhu, Ettore Biondi, Jiaxuan Li, Jiuxun Yin, Zachary E Ross, and Zhongwen Zhan. Seismic arrival-time picking on distributed acoustic sensing data using semi-supervised learning. *Nature Communications*, 14(1):8192, 2023. doi: 10.1038/s41467-023-43355-3.
- [342] Weiqiang Zhu, Alvin Brian Hou, Robert Yang, Avoy Datta, S Mostafa Mousavi, William L Ellsworth, and Gregory C Beroza. Quakeflow: a scalable machine-learning-based earthquake monitoring workflow with cloud computing. *Geophysical Journal International*, 232(1):684–693, 2023. doi: 10.1093/gji/ggac355.
- [343] Weiqiang Zhu, Haoyu Wang, Bo Rong, Ellen Yu, Stephane Zuzlewski, Gabrielle Tepp, Taka’aki Taira, Julien Marty, Allen Husker, and Richard M Allen. California earthquake dataset for machine learning and cloud computing, 2025.

- [344] Jiawei Zhuang, Daniel J Jacob, Haipeng Lin, Elizabeth W Lundgren, Robert M Yantosca, Judit Flo-Gaya, Melissa P Sulprizio, and Sebastian D Eastham. Enabling high-performance cloud computing for earth science modeling on over a thousand cores: Application to the geos-chem atmospheric chemistry model. *Journal of Advances in Modeling Earth Systems*, 12(5):e2020MS002064, 2020. doi: 10.1029/2020MS002064.