JGR Machine Learning and Computation

RESEARCH ARTICLE
10.1029/2024JH000360

PEGSGraph: A Graph Neural Network for Fast Earthquake Characterization Based on Prompt ElastoGravity Signals

Céline Hourcade[1,2], **Kévin Juhel**[1,2], **and Quentin Bletery**[1]

[1]Observatoire de la Côte d'Azur, Université Côte d'Azur, IRD, CNRS, Géoazur, Sophia Antipolis, Valbonne, France,
[2]Laboratoire de Planétologie et Géosciences, Nantes Université, CNRS, LPG UMR 6112, Nantes, France

Céline Hourcade and Kévin Juhel contributed equally to this work.

Key Points:

- We apply a graph-based deep-learning algorithm relying on prompt elastogravity signals for earthquake early source determination in Alaska
- The approach enables robust earthquake magnitude and focal mechanism estimations, down to magnitude M_w 7.6–7.7, 70 s after onset time
- The graph-based architecture captures spatial relationships and outperforms state-of-the-art early warning algorithms

Supporting Information:

Supporting Information may be found in the online version of this article.

Correspondence to:

K. Juhel,
kjuhel.pro@gmail.com

Citation:

Hourcade, C., Juhel, K., & Bletery, Q. (2025). PEGSGraph: A graph neural network for fast earthquake characterization based on prompt ElastoGravity signals. *Journal of Geophysical Research: Machine Learning and Computation*, *2*, e2024JH000360. https://doi.org/10.1029/2024JH000360

Received 19 JUL 2024
Accepted 10 DEC 2024

Abstract State-of-the-art earthquake early warning systems use the early records of seismic waves to estimate the magnitude and location of the seismic source before the shaking and the tsunami strike. Because of the inherent properties of early seismic records, those systems systematically underestimate the magnitude of large events, which results in catastrophic underestimation of the subsequent tsunamis. Prompt elastogravity signals (PEGS) are low-amplitude, light-speed signals emitted by earthquakes, which are highly sensitive to both their magnitude and focal mechanism. Detected before traditional seismic waves, PEGS have the potential to produce unsaturated magnitude estimates faster than state-of-the-art systems. Accurate instantaneous tracking of large earthquake magnitude using PEGS has been proven possible through the use of a Convolutional Neural Network (CNN). However, the CNN architecture is sub-optimal as it does not allow to capture the geometry of the problem. To address this limitation, we design PEGSGraph, a novel deep learning model relying on a Graph Neural Network (GNN) architecture. PEGSGraph accurately estimates the magnitude of synthetic earthquakes down to M_w 7.6–7.7 and determines their focal mechanisms (thrust, strike-slip or normal faulting) within 70 s of the event's onset, offering crucial information for predicting potential tsunami wave amplitudes. Our comparative analysis on Alaska and Western Canada data shows that PEGSGraph outperforms PEGSNet, providing more reliable rapid magnitude estimates and enhancing tsunami warning reliability.

Plain Language Summary Earthquake early warning systems use the first recorded seismic waves to rapidly estimate the magnitude of an earthquake and the size of a potential tsunami. They are limited by the speed of these waves and produce systematic underestimations of the magnitude of large events and subsequent tsunamis. The recent discovery of gravitational perturbations caused by earthquakes (called PEGS) may lead to a solution. PEGS travel at the speed of light, much faster than seismic waves, and can provide early and accurate information on the size of an earthquake. Recent studies showed that PEGS can be used to rapidly characterize large earthquakes through the use of a Convolutional Neural Network (CNN). However, CNN architectures are not optimal for the targeted task as they do not capture the geometry of the problem. To address this issue, we designed a Graph Neural Network (GNN), called PEGSGraph, that handles complex data structures. Tests on data from Alaska and Western Canada show that PEGSGraph outperforms PEGSNet. PEGSGraph can accurately estimate the magnitude of an earthquake (of magnitude above 7.6–7.7) 70 s after its initiation. This fast, reliable information presents a new window of opportunity to forecast the size of tsunamis and improve early warning systems.

1. Introduction

Earthquakes generate different types of seismic waves. The fastest ones, called P-waves travel through the Earth at speeds of approximately 6 km per second. These waves are the first to be detected by seismometers, providing the initial data for conventional Earthquake Early Warning (EEW) systems. The waves responsible for the damaging ground shaking felt during an earthquake, called S-waves, travel slower, at about 3–4 km per second. EEW systems capitalize on the speed difference between P-waves (the information carrier) and S-waves (the damage carrier). When an earthquake occurs, P-waves are detected first, providing a few seconds of warning before the most destructive S-waves arrive. This short lead time is critical for automated safety measures, such as stopping trains and shutting down industrial processes, as well as for public alerts (Allen & Melgar, 2019; Minson et al., 2018). Most EEW algorithms estimate the magnitude and location of the ongoing event, from which an alert may be issued in places located within a given radius (which depends on the estimated magnitude) from the estimated epicenter (Lara et al., 2023). Other EEW algorithms directly estimate the peak ground acceleration, that is, the intensity of the shaking (Joshi et al., 2024; Y. Liu et al., 2024; Saad et al., 2024).

Offshore earthquakes generate water uplift that, for the largest events, result in devastating tsunamis, which typically cause most fatalities and damage (Mori et al., 2022; Wirth et al., 2022). For tsunamis, the time available for warning is longer than for the shaking but still critical. Depending on the location of the earthquake relative to the coastline, there can be a lead time ranging from tens of minutes to several hours before the tsunami waves reach the shore. Accurate and rapid magnitude estimates are essential to predict tsunami wave heights and issue timely warnings to the exposed populations (Nicolsky et al., 2017; Whitmore et al., 2008).

However, traditional EEW systems face challenges in accurately predicting the ultimate magnitude of large earthquakes. Most regional EEW systems use point source algorithms and initial P-wave data in the first seconds after an earthquake to swiftly estimate its magnitude and source characteristics (Allen & Kanamori, 2003; Wu & Zhao, 2006). The early onset of P-waves however lacks sufficient information to reliably forecast the full extent of an earthquake's magnitude (Meier et al., 2017; Renou et al., 2019). The EEW predictions thus tend to saturate for large events and systematically underestimate the magnitude of those events, which is critical for tsunami warning. Modern tsunami warning systems use another type of wave known as the W-phase (Kanamori, 1993; Kanamori & Rivera, 2008). They provide better estimates of magnitude—in particular for large events—than P-wave-based systems but are significantly slower (magnitude estimates available ~ 10 min after the earthquake using regional seismological networks) (Duputel et al., 2011, 2012).

Global Navigation Satellite System (GNSS) has emerged as a potential valuable tool for enhancing earthquake and tsunami early warning systems (Crowell et al., 2016; Grapenthin et al., 2014; Melgar et al., 2016). GNSS records the displacement of geodetic stations, which can be significant for large earthquakes. GNSS measurements do not saturate for large events and displacement information is carried out at the P-wave speed, meaning that GNSS has the potential to provide unsaturated magnitude estimates (and therefore reliable tsunami warning) much faster than the W-phase (Minson et al., 2014). Challenges in integrating GNSS into EEW systems come from a high sensitivity of the measurements to the details of the seismic rupture (in particular the spatial distribution of the co-seismic slip) (Minson et al., 2018) and from the volume of data required to transmit for real-time monitoring, which can make such a system fail in the conditions of a catastrophic event such as a large earthquake.

Prompt elastogravity signals (coined PEGS) are gravitational fluctuations caused by the dynamic redistribution of the Earth's mass during an earthquake (Harms et al., 2015; Heaton, 2017; Montagner et al., 2016; Vallée et al., 2017). These signals manifest before traditional seismic waves and carry valuable information on the earthquake's characteristics, in particular on its magnitude and focal mechanism (Vallée & Juhel, 2019; S. Zhang et al., 2020). Leveraging the information contained in PEGS could significantly enhance the speed and accuracy of earthquake monitoring and early warning systems (Juhel et al., 2023; Vallée & Juhel, 2019; S. Zhang et al., 2020).

Given the low signal-to-noise ratios (PEGS amplitudes are a few nm/s^2 at most) and the complexity and volume of data generated by PEGS, harnessing deep learning tools is essential to effectively process and extract meaningful patterns from these signals. Recently, deep learning techniques, especially CNNs, have made significant progress in automatically detecting structures and meaningful features within data (Bergen et al., 2019; Mousavi & Beroza, 2022, 2023). This has led to the creation of powerful and computationally efficient models for tasks like time series analysis. In this context, Licciardi et al. (2022) designed a PEGS-based, deep convolutional neural network (CNN) called PEGSNet, able to track without time-delay the magnitude of $M_w \geq 8.3$ thrust earthquakes along the Japanese subduction fault. The algorithm was also applied to the Chilean subduction zone (Arias et al., 2023). However, due to constraints related to the network geometry and the number of available stations, PEGSNet's lower sensitivity was limited to $M_w \geq 8.7$ in that region. More recently, a modified version of PEGSNet (Juhel, Licciardi, & Bletery, 2024) has been applied to the Alaska region (Juhel, Bletery, et al., 2024). In addition to the characterization of the earthquake magnitude and location, this version can be used to estimate its focal mechanism. Thanks to the extensive coverage of high-quality stations in the USArray network (Busby & Aderhold, 2020; Ruppert & West, 2020), the algorithm accurately estimates the magnitude and focal mechanism of $M_w \geq 7.8$ earthquakes within 2 min of their origin time, outperforming state-of-the-art early warning algorithms based on GNSS (Juhel, Bletery, et al., 2024).

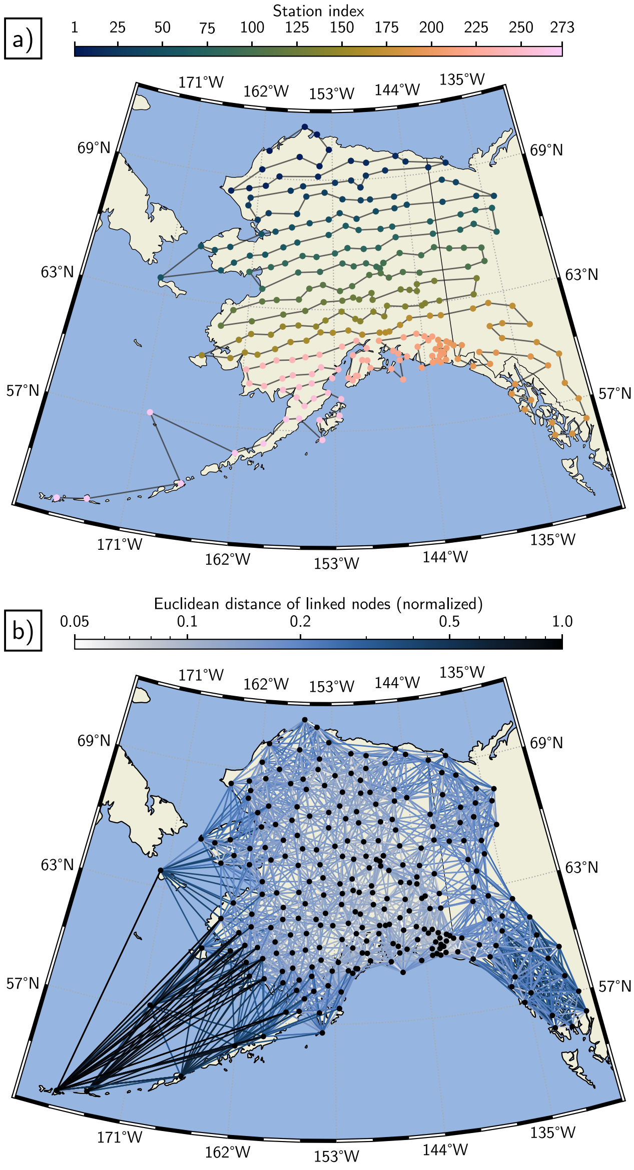

Figure 1. Location of the broadband seismometers used during training for the "complete" data set. (a) The sensors (dots) are color-coded by their corresponding index inside the input 2D images of the convolutional neural network (PEGSNet). (b) Example of a network structured as a graph, where each sensor or node (black dots) is connected to its $k = 18$ closest neighbors. The lines connecting the stations (i.e., the edges) are color-coded by the normalized distance between the nodes.

The CNN architecture of PEGSNet imposes fixed-input formats: data from seismic stations are arranged in a specific matrix structure. This requires an arbitrary station sorting to construct the input image (Figure 1a). As a result, nearby sensors may not be adjacent in the input matrix. Moreover, all the information relative to the geometry of the problem (relative distances between the potential sources and all the sensors) is lost in such a CNN architecture, which likely results in sub-optimal performances. In addition, this architecture makes PEGSNet's input inflexible in terms of size and structure, requiring a fixed format that includes all stations from a given data set. It implies that a malfunctioning sensor cannot be removed from the data set (the corresponding line in the input image is then filled with zeros, which may lead to a decrease in PEGSNet's efficiency), while the addition of any newly deployed sensors—either permanent or temporary—requires an all-new training of the convolutional neural network.

In order to leverage the lost geometrical information and provide flexibility, Geometric Deep Learning (GDL) can be employed (Bronstein et al., 2017, 2021). GDL extends deep learning techniques to non-Euclidean domains such as graphs, allowing for more effective modeling of complex relationships. Unlike conventional deep learning, which typically handles data in regular grid-like structures (e.g., images or sequences), GDL is designed to work with data where the relationships and interactions between elements are more complex and irregular like graphs. These graphs consist of nodes (or vertices) and edges that connect them, which can represent, for example, social networks (Fan et al., 2019; Shang et al., 2020), molecular structures (Atz et al., 2021; Jumper et al., 2021), or network of seismological stations.

Graph Neural Networks (GNNs), a subset of GDL, are neural networks specifically designed to handle data structured as graphs (Chami et al., 2022). GNNs process these structures to perform tasks such as node classification, link prediction, and graph classification. They leverage the connections between nodes to learn complex patterns that standard neural networks may overlook. In seismology, GNNs have been used to detect seismic events (Yano et al., 2021; Zhu et al., 2022), for phase association (McBrearty & Beroza, 2023) and source characterization (McBrearty & Beroza, 2022; Van Den Ende & Ampuero, 2020; X. Zhang et al., 2022). Those algorithms are all trained on conventional seismic waves. No GNN has been applied to PEGS yet.

In this article, we present the first GNN algorithm, called PEGSGraph (Juhel, Hourcade, & Bletery, 2024), relying on PEGS for rapid earthquake characterization. By modeling seismic networks as graphs (Figure 1b), PEGSGraph aims to capture the source-sensor spatial configurations more effectively. This approach could potentially enhance the robustness and flexibility of earthquake monitoring systems by providing more accurate and timely predictions of earthquake magnitude and focal mechanism. We compare PEGSGraph's performance with PEGSNet's in the Alaska region (Juhel, Bletery, et al., 2024). This comparison highlights the superiority of a GNN-based approach for seismic event monitoring capabilities.

2. Implementation and Training

2.1. PEGSGraph Architecture

PEGSNet's architecture consists of a series of convolutional blocks followed by fully connected layers (Licciardi et al., 2022). As input data, PEGSNet uses a matrix where each row represents the waveform from a different station. The output layer consists of nine elements corresponding to the event magnitude, the source location (latitude and longitude) and the six moment tensor parameters (Juhel, Bletery, et al., 2024; Juhel, Licciardi, & Bletery, 2024). PEGSNet and PEGSGraph detailed architectures are compared in Figure 2.

PEGSGraph's framework consists in the following steps: temporal features are first extracted from each individual recorded waveform, using a CNN with eight convolutional layers inspired by PEGSNet's architecture. We then create a k-nearest neighbors graph corresponding to a seismic event: each node inside the graph represents a seismological station and is connected to its $k = 18$ closest geographic neighbors (for more details on the impact of graph structure on performance, see Section 4.1.). We fill this initial input graph, at the node level, with the features obtained in the previous step. The learning targets at the graph level are the nine parameters corresponding to the considered seismic event: the event magnitude, the source location (latitude and longitude) and the seismic moment tensor (which includes six parameters). We then perform a graph-level prediction, and train the network to recover the earthquake parameters (the targets at the graph scale) with a succession of graph convolutions and fully connected neurons.

The graph convolutions are performed by a Graph Convolutional Network (GCN) (Kipf & Welling, 2016). Graph convolutions, inside a GCN, are analogous to image convolutions in that the "filter" parameters are typically shared across all locations within the graph. Furthermore, GCNs utilize message passing methods, wherein nodes exchange information with their neighbors and transmit "messages" to one another (see Figure 3 for an illustration of the message passing principle). Each node initially generates a feature vector that represents the message to convey to its neighboring nodes. These messages are then transmitted to the neighbors, resulting in each node receiving one message from each adjacent node. An aggregation step allows each node to update its own feature representation based on information gathered from its immediate neighbors. A GCN layer is formally defined as follows:

$$H^{(l+1)} = \sigma\left(\hat{D}^{-1/2}\hat{A}\hat{D}^{-1/2}H^{(l)}W^{(l)}\right) \tag{1}$$

where $H^{(l)}$ and $H^{(l+1)}$ are the input and output node features of the given layer, respectively. The trainable weight parameters $W^{(l)}$ are used to transform the input features into messages, $H^{(l)}W^{(l)}$. $\hat{A}$ is the adjacency matrix A with inserted self-connections (i.e., $\hat{A} = A + I$, with I the identity matrix), such that each node receives messages from itself and its neighboring nodes. $\hat{D}$, defined as $D_{ii} = \sum_{j=0}\hat{A}_{ij}$, represents the weighted number of neighbors of node i. σ represents an arbitrary activation function, a ReLU-based activation function is typically employed.

Adding successive GCN layers allows nodes to integrate information from a larger number of neighbors over several hops in the graph. In each subsequent layer, nodes thus accumulate and refine their understanding of the graph structure and relationships, obtaining indirect information from nodes further down the graph. We experimented with various GNN architectures by adjusting key hyper-parameters, including the number of GCNConv layers (from three to seven layers) and the number of channels per layer (64 or 128). The performance outcomes for these configurations are detailed in Section 4.1. We finally chose to implement six layers of GCNConv with 128 channels.

2.2. Synthetic Data Set of Noisy PEGS

For comparison purposes, we train PEGSGraph on the data set presented in Juhel, Bletery, et al. (2024). This ensures consistency in evaluating and comparing the performance of the GNN and CNN models. This database is composed of synthetic sources based on the distribution of the historical shallow (<60 km) seismicity in Alaska reported by Global CMT since 1976 (Ekström et al., 2012) (Figure S1a in Supporting Information S1). Three distinct geographic regions have been created (see colored polygons in Figure S1b in Supporting Information S1): these regions correspond to synthetic strike-slip, thrust and normal fault sources. The synthetic sources are modeled as pure double-couple mechanisms. The corresponding density distributions of strike, dip and rake angles are detailed in Figure S2 in Supporting Information S1.

The entire process of creating the data set is detailed in Figure 4. This figure provides a step-by-step overview of the data pipeline, from the initial data collection and pre-processing to the final input construction. First, we calculate PEGS Green's functions (Figure 4.1) between each source and each sensor location, inside a 1D Earth model (AK135), using the QSSP algorithm (Wang et al., 2017; S. Zhang et al., 2020).

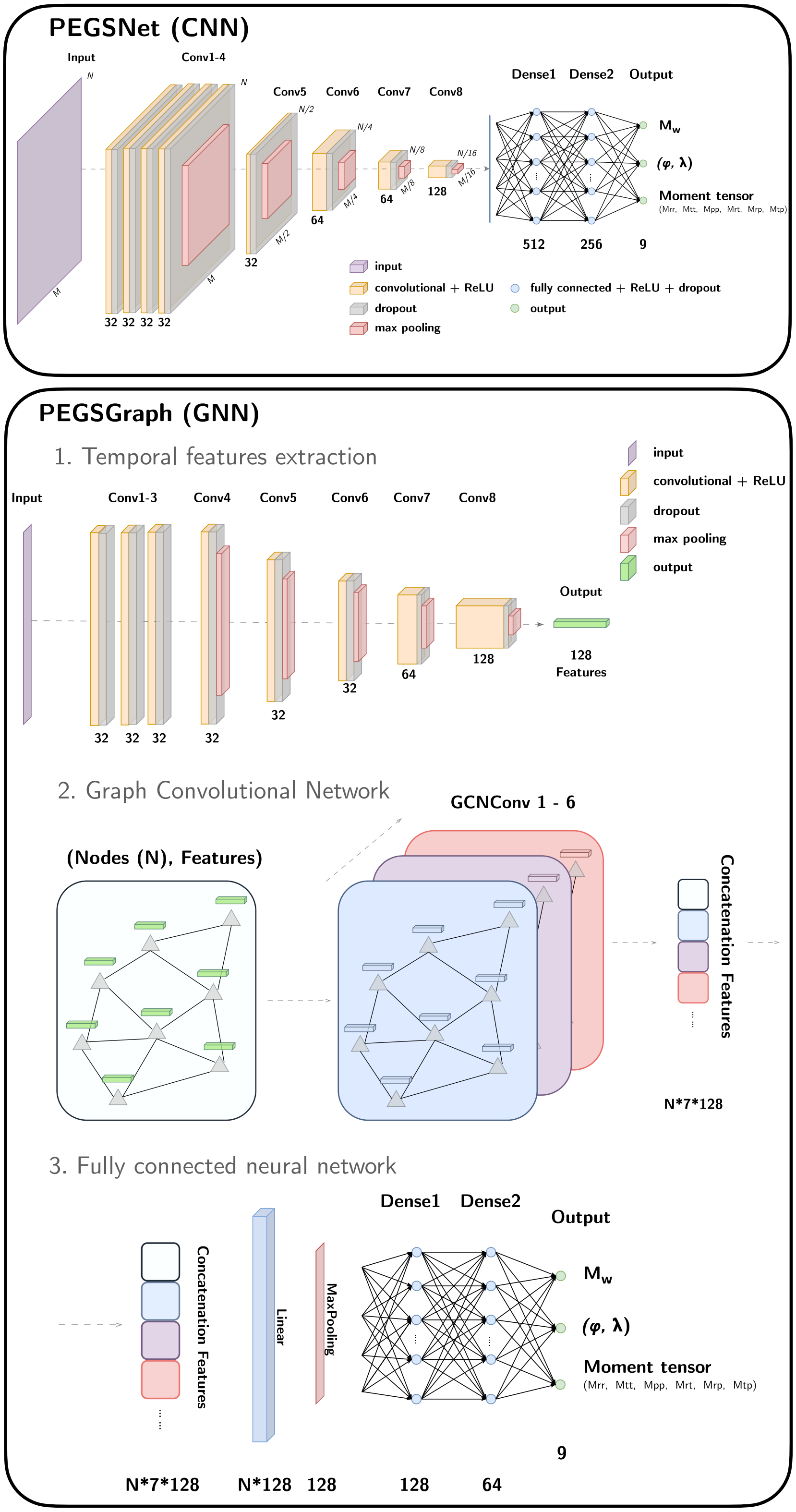

Figure 2. PEGSNet (CNN, top) and PEGSGraph (GNN, bottom) architectures. The number of channels in each convolutional layer is shown for clarity. The GNN architecture is divided into several parts: 1) a station-wise, temporal feature extraction, 2) several rounds of message passings to obtain node embeddings and 3) a readout layer to aggregate node embeddings into a graph embedding, followed by a final prediction layer.

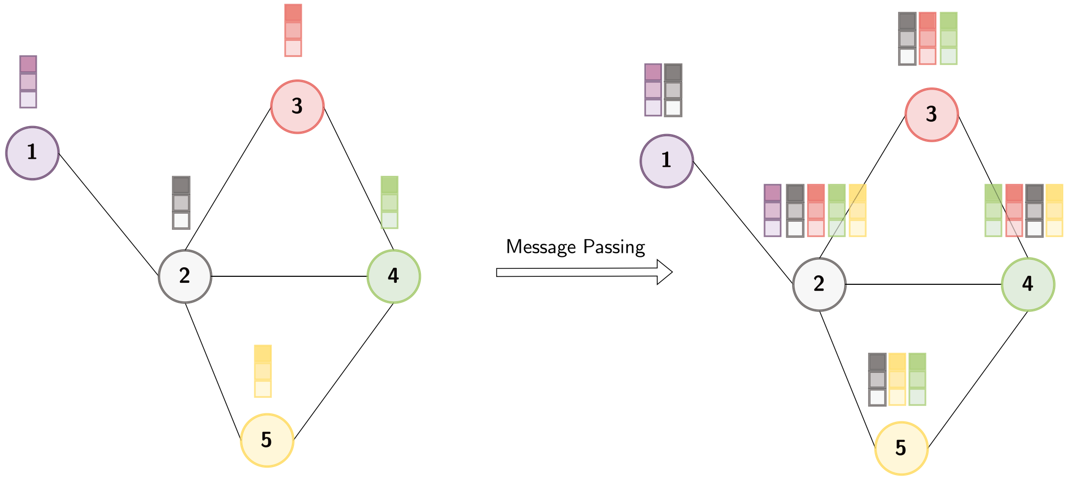

Figure 3. Illustration of the message passing principle in a Graph Convolutional Network (GCN). Each node initially generates a feature vector based on its own attributes (left). These feature vectors are then transmitted to neighboring nodes along the graph edges (right). Each node aggregates the received messages to update its own feature representation, incorporating information from its immediate neighbors. This process is repeated across multiple layers, allowing nodes to capture increasingly complex patterns by considering information from a wider neighborhood in the graph.

The moment-rate source time-function (STF) database is computed using the functional form $y(t) = \mu t \exp\left[-(\lambda t)^2/2\right]$ described by Meier et al. (2017), where the parameters μ (initial slope) and $\lambda(M_w)$ (inverse characteristic time scale) are fitted based on empirical STF data sets. To take into account the varying STF duration for a given final magnitude, $\lambda(M_w)$ is perturbed by a multiplicative term ϵ extracted from the normal distribution $\mathcal{N}\{\mu = 0, \sigma = 0.15\}$. The final moment magnitude M_w is extracted from three uniform distributions: $\mathcal{U}\{5.5,\ 9.5\}$ for thrust, $\mathcal{U}\{5.5,\ 8.7\}$ for strike-slip, and $\mathcal{U}\{5.5,\ 8.4\}$ for normal mechanisms. An example of a STF for a $\mathrm{M_w} = 8.0$ earthquake is shown in Figure 4.2. The STF is finally corrupted with multiplicative Brownian noise, to reproduce the variety of STFs observed in nature. The PEGS synthetic waveforms are finally convolved with the STF (Figure 4.3), and band-pass filtered between 2.0 mHz (using a high-pass Butterworth causal filter with two poles) and 30.0 mHz (using a low-pass Butterworth causal filter with six poles).

In order to obtain a database of realistic synthetic signals, we add empirical noise recorded by all the broadband sensors during common time windows (therefore preserving network-scale correlations of the noise recorded on the different stations). We apply a basic pre-processing (which can be applied in real-time in an operational early warning system) to all the obtained synthetic data: we remove the linear trend and instrumental response from the daily vertical recordings (BHZ channels), we band-pass filter the resulting traces between 2.0 and 30.0 mHz, and then decimate them to 1 Hz (Figure 4.4). Due to the very low amplitude of PEGS, we exclude the noisiest sensors from the data set, retaining only those sensors whose median hourly standard deviation does not exceed 1 nm/s^2 over a period of 1 year.

Finally, we gather the synthetic PEGS data with previously recorded noise signals. Each signal is then truncated at the P-wave arrival time, after which the remainder is replaced by zeros to produce a standardized signal length of 300 s (Figure 4.5). Figure 4.6 illustrates how these data are used as inputs for the two algorithms. For PEGSNet, the data are concatenated into a 2D image where each row corresponds to the signal from a single station, with a predefined station order (Figure 1a). For PEGSGraph, on the other hand, the data are processed individually by a 1D CNN at the input stage of the algorithm (Figure 2).

We gather two distinct data sets: a "complete" data set, corresponding to the full deployment of the station network (all good-quality broadband sensors operating between 2018 and 2020), and a "legacy" data set, corresponding to all good-quality broadband sensors still operating by the end of the 2021 field season (hence after the removal of the non-transferred, Alaska TA stations). Each training data sets comprises 512,000 synthetic earthquakes. Figure 1b illustrates the connections between the sensors of the "complete" network, structured as a graph (see Figure S3 in Supporting Information S1 for the corresponding connections in the "legacy"

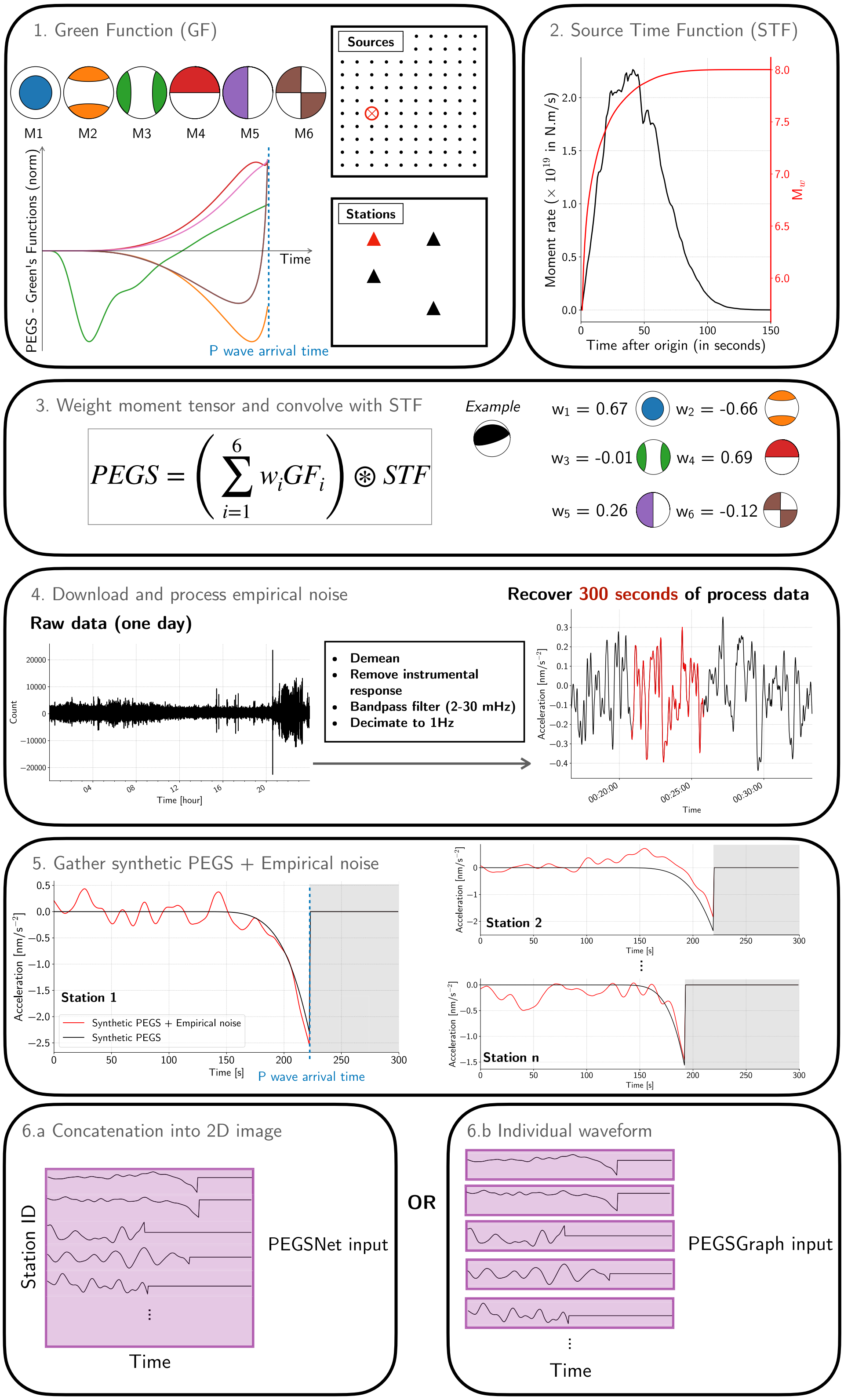

Figure 4. Overview of the data set creation process. See Section 2.2 of the main text for detailed explanations.

configuration). To ensure clarity, the majority of the results presented in this study correspond to training conducted using the "complete" data set.

2.3. Training

We divide the two synthetic data sets into three parts: the training sets, validation sets and test sets, which constitute 70%, 20%, and 10% of the data, respectively. During training, the Smooth L1 loss (with a cutoff for $\beta = 0.3$) is used to minimize the discrepancy between the ground truth and the predicted values over the course of 200 epochs, with batch sizes of 512. After each epoch, the model's performance is assessed using the validation set, and the model with the lowest validation loss is selected as the final model. This final model is then evaluated on the test set. In our approach, we trained the model independently 10 times and selected the best-performing model based on these runs. This method allowed us to identify the most optimal training outcome. However, alternative techniques, such as K-fold cross-validation, could also be applied to further enhance the robustness of the model evaluation by systematically training and validating on different subsets of the data. Since PEGSNet requires retraining for every change in network configuration, we conducted training sessions of PEGSGraph for both the "complete" and "legacy" configurations. The corresponding training and validation curves are presented in Figure S4 in Supporting Information S1.

We also tested training the model on a reduced data set, using only 10% of the full data (results shown in Figure S5 in Supporting Information S1). The results indicate a noticeable drop in performance when training on a data set of 51,200 samples, with particularly larger errors observed in the location of seismic events. Given the complexity of our problem, which involves accurately characterizing a large variety of seismic events, it is clear that a substantial data set is required to fully capture the underlying patterns and relationships necessary for effective model training.

3. PEGSNet and PEGSGraph Performance Comparison

3.1. Performance Comparison on Synthetic Earthquakes

The magnitude regression performances obtained with PEGSNet and PEGSGraph for the "complete" data set are shown in Figures 5a–5f. Figures 5g–5i displays the accuracy ratio between PEGSNet and PEGSGraph for easier comparison. In this study, we consider a magnitude estimate to be accurate if the predicted time-dependent moment magnitude $M_w^{pred}(t)$ is within 0.4 units of the true magnitude $M_w^{true}(t)$ (as defined in Juhel, Bletery, et al. (2024) for easier comparison). The M_w estimates are separated depending on the predicted type of focal mechanism: thrust (plunge of tension axis above 45°), strike-slip (plunge of null axis above 45°) or normal mechanism (the remaining samples). First of all, we note that for both PEGSNet and PEGSGraph models, M_w estimates are more accurate for normal and strike-slip earthquakes than for thrust events, which are known to generate smaller PEGS for a given magnitude (Vallée & Juhel, 2019). However, PEGSGraph demonstrates a notable performance improvement for thrust events: the accuracy ratio shows a performance improvement of around 5%, 150 s after onset (after the origin time of the earthquake), for $8.5 \leq M_w^{pred} \leq 8.9$, at least a 10% advantage for PEGSGraph for $M_w^{pred} \leq 8.1$ and a 20% advantage for $M_w^{pred} \leq 7.6$ (Figure 5g). Gains are also observed for strike-slip and normal events, with PEGSGraph outperforming PEGSNet by at least 10% for magnitudes under 7.7 (Figures 5h and 5i). For PEGSGraph, the accuracy level exceeds 75% when the predicted magnitude is above $M_w^{pred} = 7.6$ for thrust events (compared to $M_w^{pred} = 8.0$ for PEGSNet), $M_w^{pred} = 7.5$ for strike-slip events (compared to 7.8 for PEGSNet) and $M_w^{pred} = 7.4$ for normal events (compared to 7.5 for PEGSNet).

In order to evaluate the performance of the moment tensor reconstruction, we use the geometrical similarity α (Rivera & Kanamori, 2014). This metric quantifies the similarity between two moment tensors, regardless of the seismic moment: $\alpha = 0$ indicates opposite tensors, while $\alpha = 1$ denotes identical tensors. We define a reconstruction as successful if $\alpha \geq 0.8$ (as defined in Juhel, Bletery, et al. (2024) for easier comparison), relative to the input focal mechanism (ground truth). The corresponding accuracy of the mechanism reconstruction for each event is depicted in Figure S6 in Supporting Information S1. Similar outcomes are observed between PEGSNet and PEGSGraph models.

We further compare PEGSNet and PEGSGraph performances on magnitude and focal mechanism estimations, but also in locating seismic events (Figure S7 in Supporting Information S1). For this purpose, we analyze various

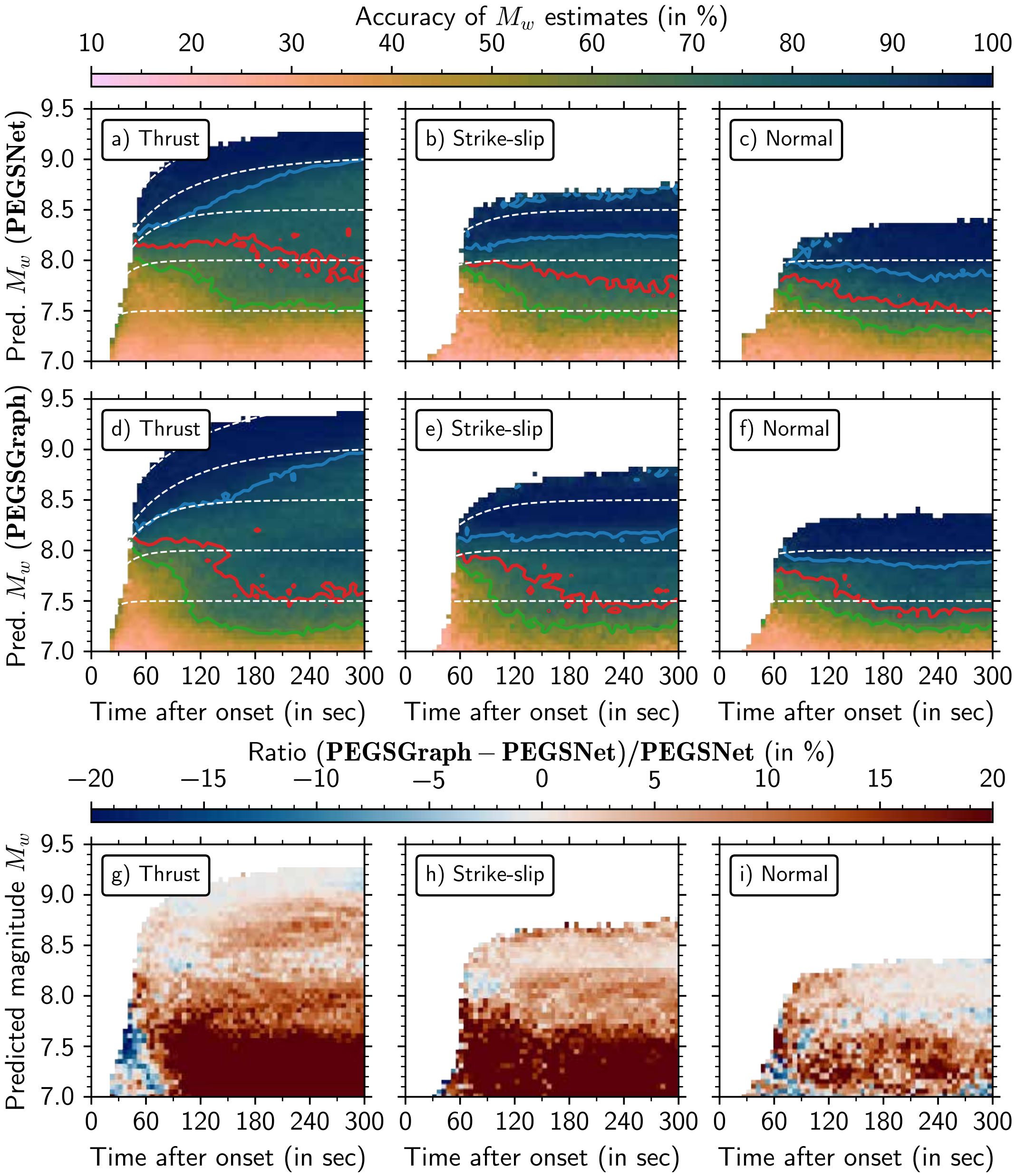

Figure 5. Accuracy of magnitude estimates for PEGSNet (a–c) and PEGSGraph (d–f) on the test set, for the "complete" sensor network. The test set is divided into *predicted* thrust (a, d), strike-slip (b, e) and normal (c, f) samples. A magnitude estimate is considered successful if $|M_w^{true}(t) - M_w^{pred}(t)| < 0.4$. Contour lines indicate accuracy of 90% (blue), 75% (red), and 60% (green), respectively. The accuracy ratio between PEGSGraph and PEGSNet is shown in (g–i).

ranges of M_w^{pred} for each focal mechanism using the median absolute error (MedAE), 4 min after onset time. Regarding magnitude estimation and event location, PEGSGraph systematically outperforms PEGSNet for thrust and strike-slip events, while the performances are comparable for normal fault events. For thrust events for instance, PEGSGraph achieves a MedAE lower than 0.2 magnitude units (PEGSNet's MedAE being systematically above 0.2), and reduces the MedAE of event location by 15 km compared to PEGSNet. Lastly, for source mechanism reconstruction, the results are relatively similar between PEGSGraph and PEGSNet.

To further assess PEGSGraph's performance compared to PEGSNet's (Juhel, Bletery, et al., 2024), we specifically examine the same subset of test samples falling within defined final magnitude ranges: $M_w^{true} = 9.0 \pm 0.05$, $M_w^{true} = 7.8 \pm 0.05$, $M_w^{true} = 7.7 \pm 0.05$ and $M_w^{true} = 7.6 \pm 0.05$ (Figure 6). The mode distributions for

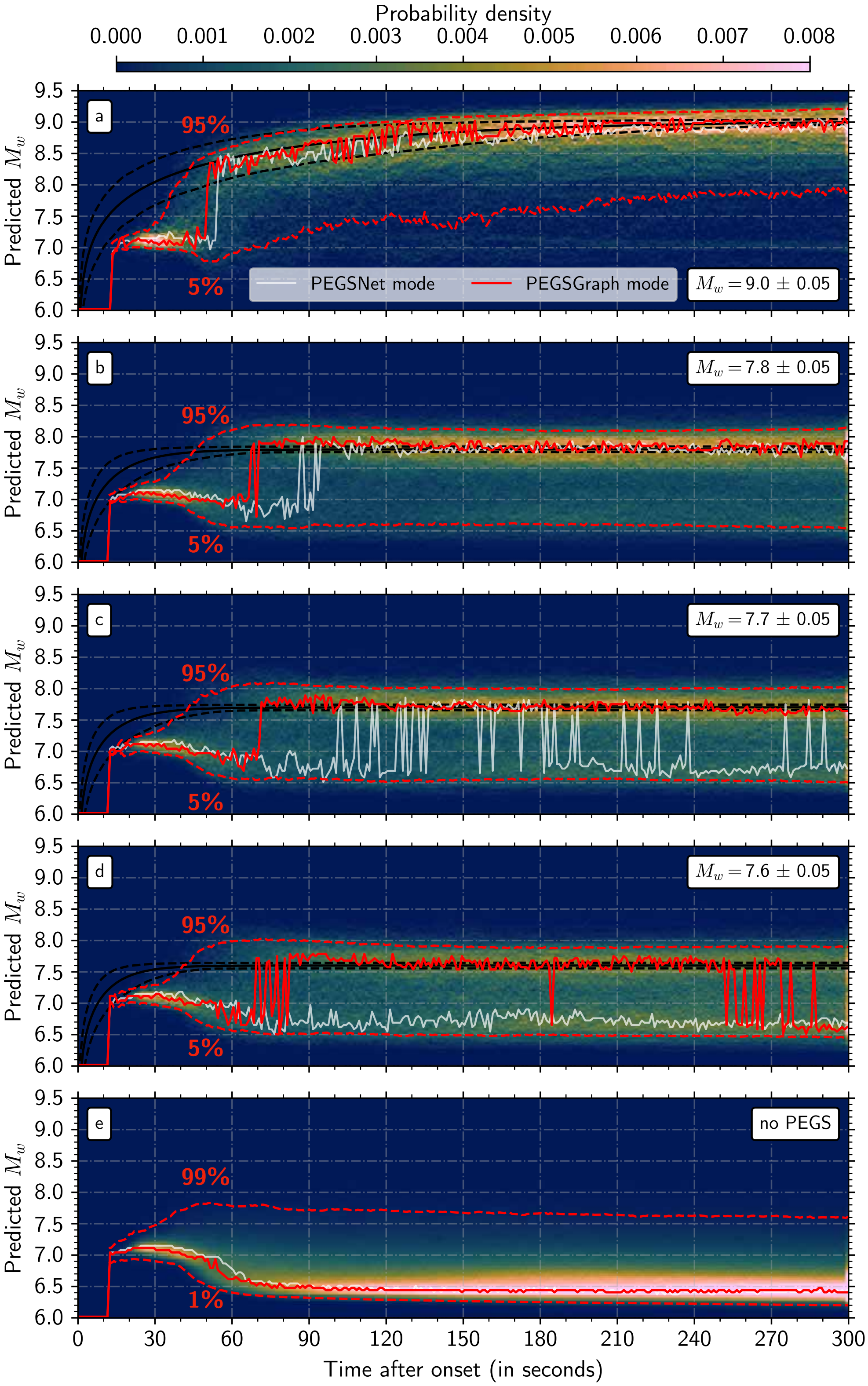

Figure 6. Magnitude estimates obtained with PEGSGraph for the "complete" network. (a–d) Probability density of magnitude estimates for samples with actual final magnitudes of $M_w = 9.0$, $M_w = 7.8$, $M_w = 7.7$, and $M_w = 7.6 \pm 0.05$, respectively. The red lines represent the mode (solid) and the 5th and 95th percentiles (dashed). The white line represents the distribution mode obtained with PEGSNet. The black lines indicate the median (solid) and the 5th and 95th percentiles (dashed) of the true magnitudes, for reference. (e) Probability density of magnitude estimates for all test set events when PEGS are excluded from the synthetic waveforms.

$M_w^{true} = 9.0 \pm 0.05$ events are quite similar between both architectures: PEGSGraph's mode jumps toward the target value 5 s before PEGSNet's mode, and subsequently oscillates closer to the ground truth. For events with $M_w^{true} = 7.8 \pm 0.05$, PEGSNet provides an initial estimate starting from $t = t_{EQ} + 90$ s, whereas the GNN outputs a robust solution starting from $t = t_{EQ} + 70$ s.

We observe significant performance improvements for lower magnitude events. For events of $M_w^{true} = 7.7 \pm 0.05$, PEGSNet exhibits a bimodal distribution mode, oscillating in between $M_w^{pred} \sim 6.7$ and the target value (thus indicating a $M_w = 7.8$ lower sensitivity for PEGSNet). In contrast, PEGSGraph consistently delivers a stable and reliable magnitude estimate, starting 70 s after onset time. For events with $M_w^{true} = 7.6 \pm 0.05$, PEGSNet most often fails to converge toward the target value, with the distribution mode remaining at $M_w^{pred} \sim 6.7$. PEGSGraph provides a robust estimation 80 s after onset time and up to 250 s, beyond which a bimodal distribution of the mode is observed. These results suggest a sensitivity threshold for moment magnitude estimation in between $M_w = 7.6$ and $M_w = 7.7$.

The predictions displayed in Figure 6e reaffirm that the estimates rely exclusively on PEGS-derived information. When PEGS are removed from the synthetic waveforms (while retaining P-wave arrival data), PEGSGraph consistently outputs a $M_w^{pred} \sim 6.5$ estimate (considered as the 0 baseline), similarly to PEGSNet (Juhel, Bletery, et al., 2024). Moreover, the noise samples at the 99% confidence level fall below the sensitivity threshold of $M_w = 7.7$, as determined earlier for $t \geq t_{EQ} + 120$ s.

3.2. Performance Comparison on Recent Real Earthquakes

We fed PEGSGraph with broadband data recorded during the four significant recent earthquakes that occurred during the period covered by the "complete" and the "legacy" networks: (a) the 29 July 2021 M_w 8.2 Chignik subduction earthquake (C. Liu et al., 2022; Ye et al., 2022), (b) the 23 January 2018 M_w 7.9 Kodiak strike-slip earthquake (Krabbenhoeft et al., 2018; Ruppert et al., 2018), (c) the 22 July 2020 M_w 7.8 Shumagin subduction earthquake (Crowell & Melgar, 2020; C. Liu et al., 2020), and (d) the 19 October 2020 M_w 7.6 Sand Point intraslab earthquake (Herman & Furlong, 2021; Santellanes et al., 2022). The location of all the earthquakes is shown in Figure S1a in Supporting Information S1. In Figure 7, we present a comparison of PEGSNet's and PEGSGraph's simulated real time performances.

For the 2021 Chignik earthquake (M_w 8.2), both models yield similar results. PEGSNet magnitude estimates approach the target value around 60 s after onset time (70 s for PEGSGraph), and stabilize by 120 s (150 s for PEGSGraph). Both algorithms indicate an accurate dip-slip mechanism very early on. Although magnitude estimation is slightly faster for PEGSNet, we observe more stable prediction oscillations for PEGSGraph for $t \geq t_{EQ} + 150$ s. We observe comparable results for the M_w 7.9 Kodiak earthquake, with both models providing accurate estimates of the focal mechanism and magnitude starting from 120 s after the onset for PEGSNet and 150 for PEGSGraph. We also note a more stable solution for PEGSGraph from $t \geq t_{EQ} + 150$ s. In the case of the M_w 7.8 Shumagin earthquake, PEGSGraph demonstrates notably better stability. The solution converges to $M_w = 7.8$ from 90 s after the onset (compared to 140 s for PEGSNet, with subsequent larger oscillations of the M_w estimates). Finally, we observe distinct performance differences for the M_w 7.6 Sand Point earthquake. This event is below PEGSNet's sensitivity threshold, resulting in inaccurate solutions. In contrast, PEGSGraph shows a stable solution close to the target magnitude around 150 s after onset time. However, none of algorithms is able to effectively reconstruct the focal mechanism of the source: the geometric similarity α is indeed 0.59 and 0.63 for PEGSNet and PEGSGraph, respectively.

Both models were trained on a database of earthquakes with purely double-couple mechanisms. This impacts the predictions, as the model is limited by the lack of diversity in focal mechanisms within the training data. Consequently, PEGSGraph and PEGSNet struggle to accurately characterize real sources that deviate from the double-couple assumption. Based on the study by Jost and Herrmann (1989), a percentage of the double-couple component of a seismic source can be determined. For the Sand Point earthquake, this percentage reaches 71.05%. In comparison, the percentages for the Shumagin and Chignik events are 98.35% and 97.98%, respectively. The non-double-couple nature of the Sand Point source, combined with a small seismic moment and very slow STF onset, impacts the algorithms' ability to accurately reconstruct the focal mechanism. Overcoming this limitation would require creating a more diverse database including a more exhaustive variety of focal mechanisms. This

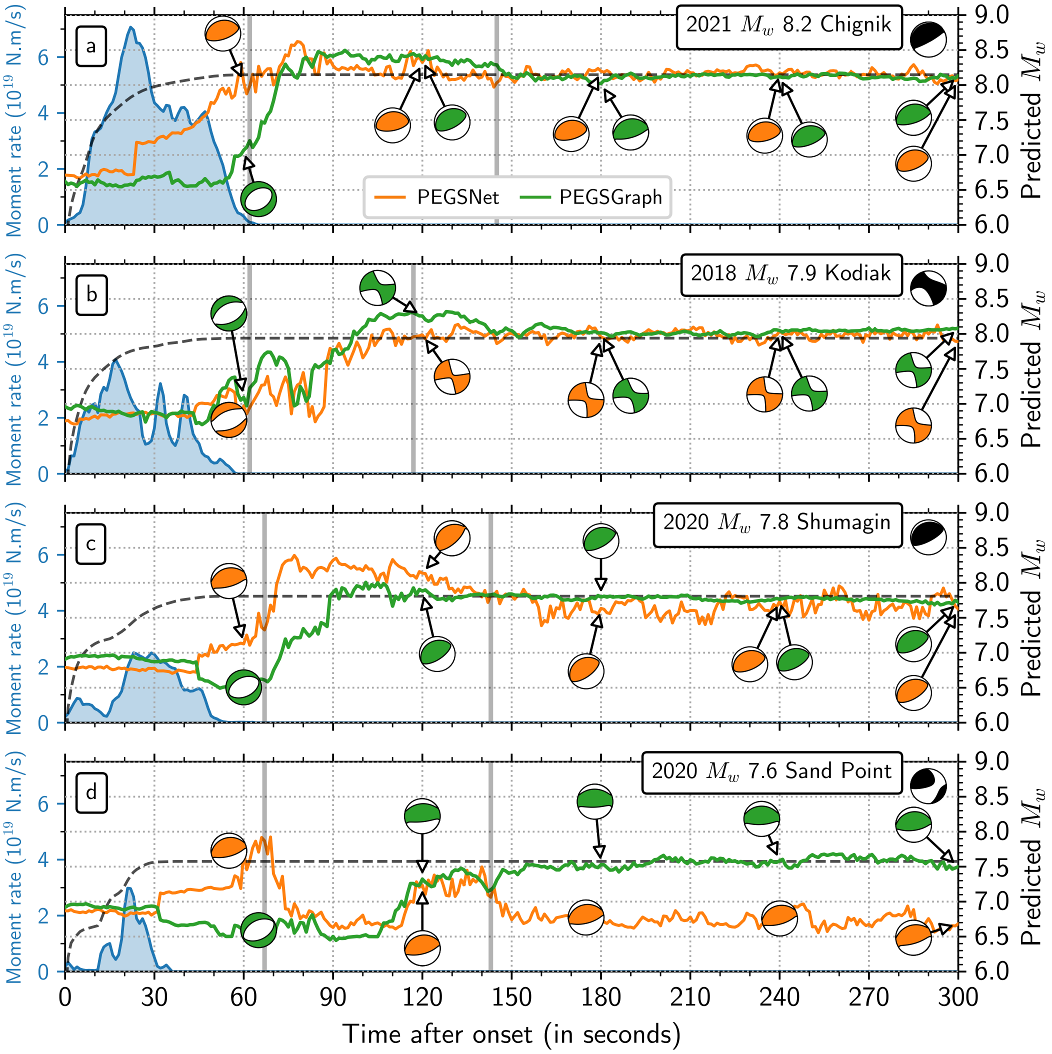

Figure 7. Magnitudes estimated by PEGSNet (orange) and PEGSGraph (green) as a function of time after the onset of (a) the 29 July 2021, M_w 8.2 Chignik earthquake, (b) the 23 January 2018, M_w 7.9 Kodiak earthquake, (c) the 22 July 2020, M_w 7.8 Shumagin earthquake, and (d) the 19 October 2020, M_w 7.6 Sand Point earthquake. The moment-rate source time functions from the SCARDEC database (Vallée & Douet, 2016) are depicted in blue, with the corresponding accumulated moment magnitudes shown in black (dashed) for reference. The estimated solutions of the moment tensor are illustrated by the orange and green plots of the focal mechanisms for PEGSNet and PEGSGraph, respectively. The GCMT solution is provided in black for reference for each event. The top panel uses the "legacy" channels, whereas the bottom three panels utilize the "complete" network. In each subplot, the gray vertical lines mark the P-wave arrival times at the 10th and 100th stations closest to the source.

would enhance the model's ability to generalize and accurately predict the characteristics of a broader range of seismic events (to the cost of a significant increase in training time).

To evaluate the robustness of the model under varying noise conditions, we designed tests across different noise scenarios. For this purpose, we generate synthetic PEGS using the source parameters estimated by GCMT (Ekström et al., 2012) for the four events mentioned above. We corrupt these synthetic signals by adding 1,000 randomly selected time-windows of empirical noise extracted from the test set. By collecting magnitude estimations for each of these newly generated test samples, we derive uncertainty estimates for each earthquake (see Figure 8). The results exhibit greater stability for PEGSGraph, with estimations clustered closer to the target values for the Chignik, Kodiak and Shumagin earthquakes. For the Sand Point earthquake, the distribution mode reached the target value within 120 s (Figure 8h), indicating consistent performance across different noise scenarios.

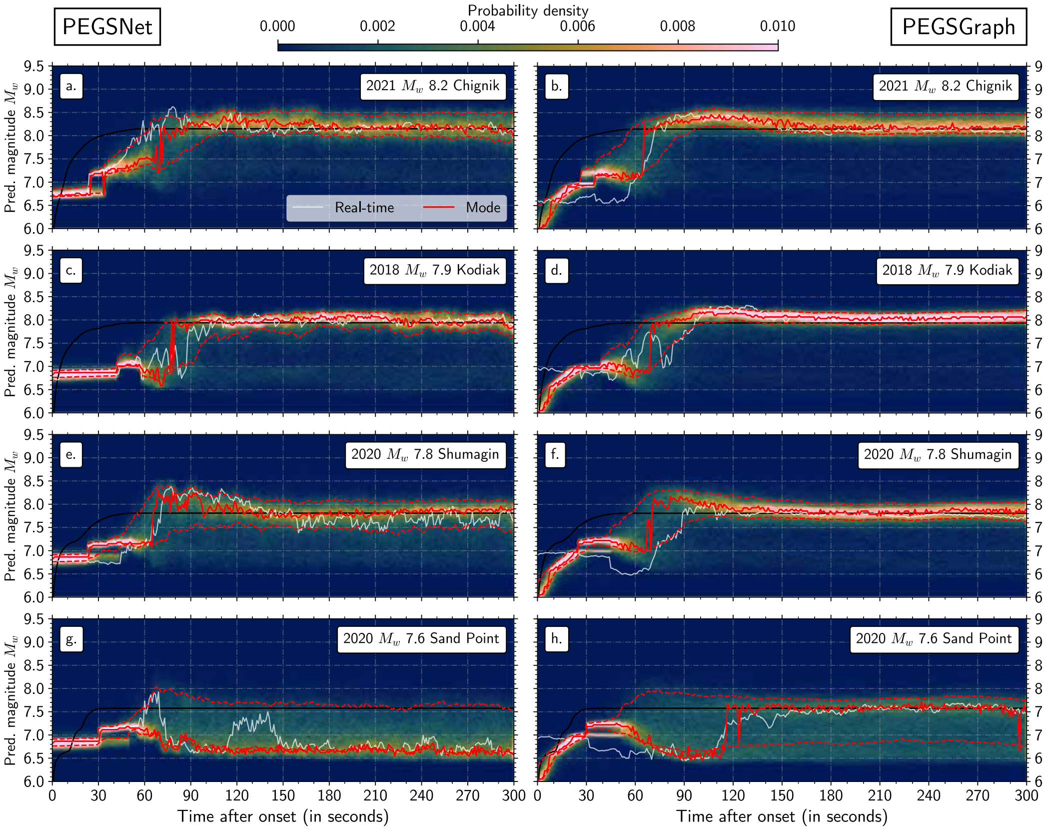

Figure 8. Magnitude estimates on 1,000 synthetic samples obtained by combining PEGS derived from the four studied earthquakes (based on their GCMT parameters) with noise time windows from the test set, shown for PEGSNet (left) and PEGSGraph (right). The red lines indicate the mode (solid) and the 25th to 95th percentiles (dashed), while the white lines represent the actual simulated real-time prediction (Figure 7). Cumulative moment magnitudes from the SCARDEC database (Vallée & Douet, 2016) are provided in black as reference for each event. The locations of all the earthquakes are shown in Figure S1a in Supporting Information S1.

There are no additional examples of earthquakes within the magnitude range of interest and recorded during the period covered by the "complete" and the "legacy" networks. To overcome this limitation, we simulated synthetic PEGS using the source parameters estimated by GCMT (Ekström et al., 2012) for four other historical seismic events: (a) the 03 November 2002 M_w 7.8 Central Alaska strike-slip earthquake, (b) the 30 November 1987 M_w 7.8 Gulf of Alaska strike-slip earthquake, (c) the 06 March 1988 M_w 7.7 Gulf of Alaska strike-slip earthquake and (d) the 05 January 2013 M_w 7.5 Southeastern Alaska strike-slip earthquake. The location of all these earthquakes is shown in Figure S1a in Supporting Information S1. As in previous experiments, these synthetic signals were corrupted by adding 1,000 randomly selected time-windows of empirical noise extracted from the test set. The results, shown in Figure S8 in Supporting Information S1, indicate that PEGSGraph maintains stable performance even in the presence of significant noise.

4. Controlling Factors' Impact on the Algorithms Performance

In this section, we assess the impact of several parameters on the performance of the two models: the graph architecture for PEGSGraph, the pre-event noise level, the source location and the station network density, and, finally, the effectiveness of the source time-function. Additionally, we examine the generalization capabilities of PEGSGraph and PEGSNet to other regions.

4.1. Impact of the Graph Structure and Architectures

To investigate the impact of the graph configuration on model performance, we changed the number of nearest neighbors, testing values of $k = 6, k = 12, k = 18, k = 24$, and $k = 30$. Given that a K-nearest neighbors (K-NN)

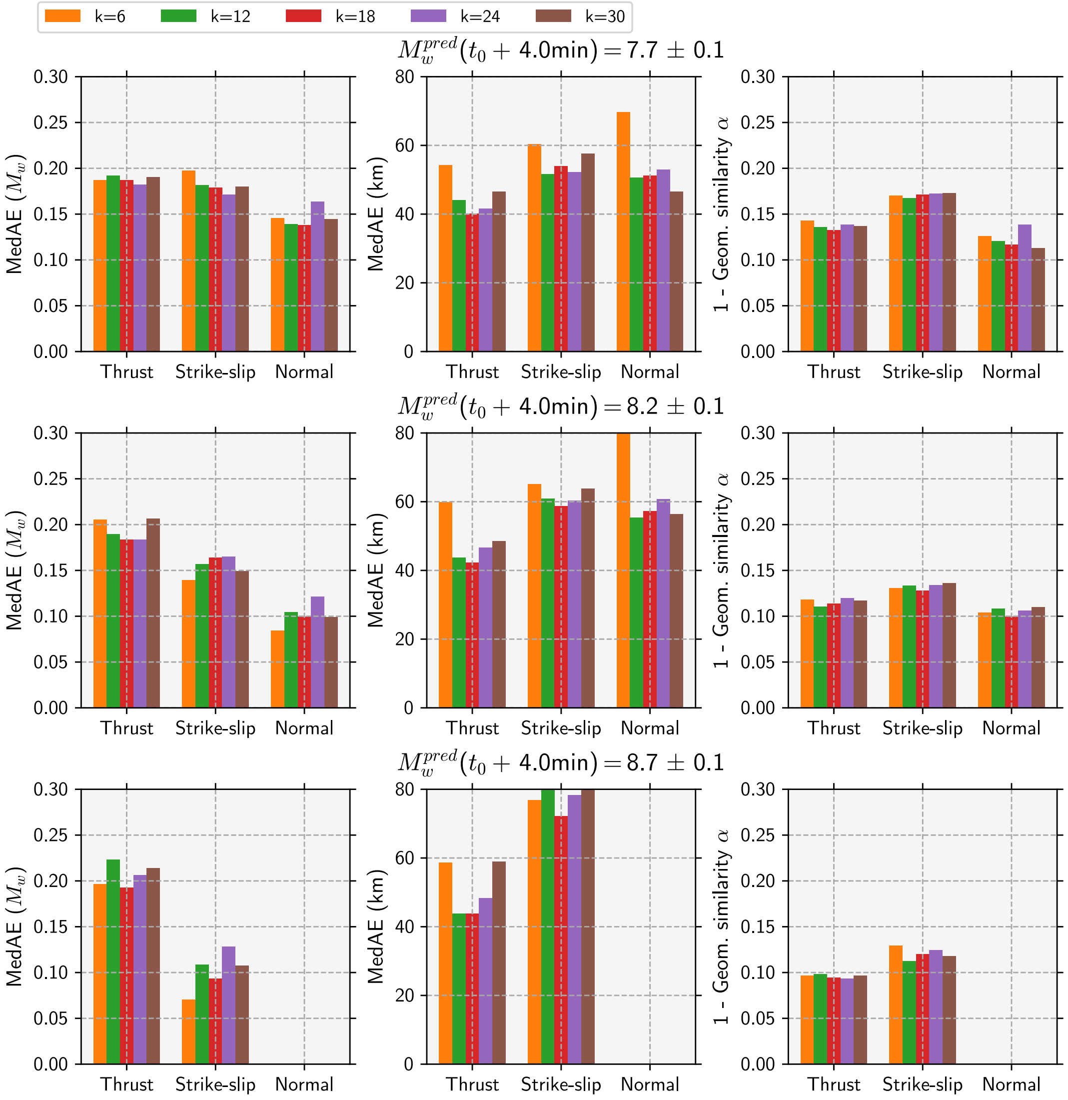

Figure 9. Median Absolute Error (MedAE) of the magnitude (left), location (center), and geometrical similarity (right) obtained using K-nearest neighbors graphs with different values of *k*. Results are shown for the different event types and predicted magnitudes, measured 4 min after the earthquake onset. A minimum of 10 samples is required to calculate the MedAE.

graph was employed, each seismic station is connected to its *k* closest neighboring stations based on geographic proximity. This configuration influences how spatial information is propagated through the graph. A smaller value of *k* results in a more localized graph structure, focusing primarily on nearby stations, whereas larger values introduce longer-range connections, potentially capturing broader spatial dependencies. The performance differences across these configurations are detailed in Figure 9, where the effects on magnitude, location and focal mechanism estimations are illustrated. We observe that using a smaller number of neighbors ($k = 6$) results in larger location errors. However, for larger values of *k* (e.g., $k = 12$, $k = 18$, $k = 24$, and $k = 30$), the results are similar, with a slight improvement in accuracy for $k = 18$. These results suggest that while larger graphs capture more spatial information, $k = 18$ achieves an optimal balance between including relevant connections between stations and avoiding the inclusion of less informative distant neighbors.

In addition to exploring the effects of varying *k* in the K-nearest neighbors graph, we investigate the impact of different architectural configurations on PEGSGraph's performance. This includes experimenting various Graph Convolutional Network (GCN) architectures, where we adjusted both the number of GCNConv layers and the number of channels per layer. The results of these comparisons are presented in Figure 10, illustrating the

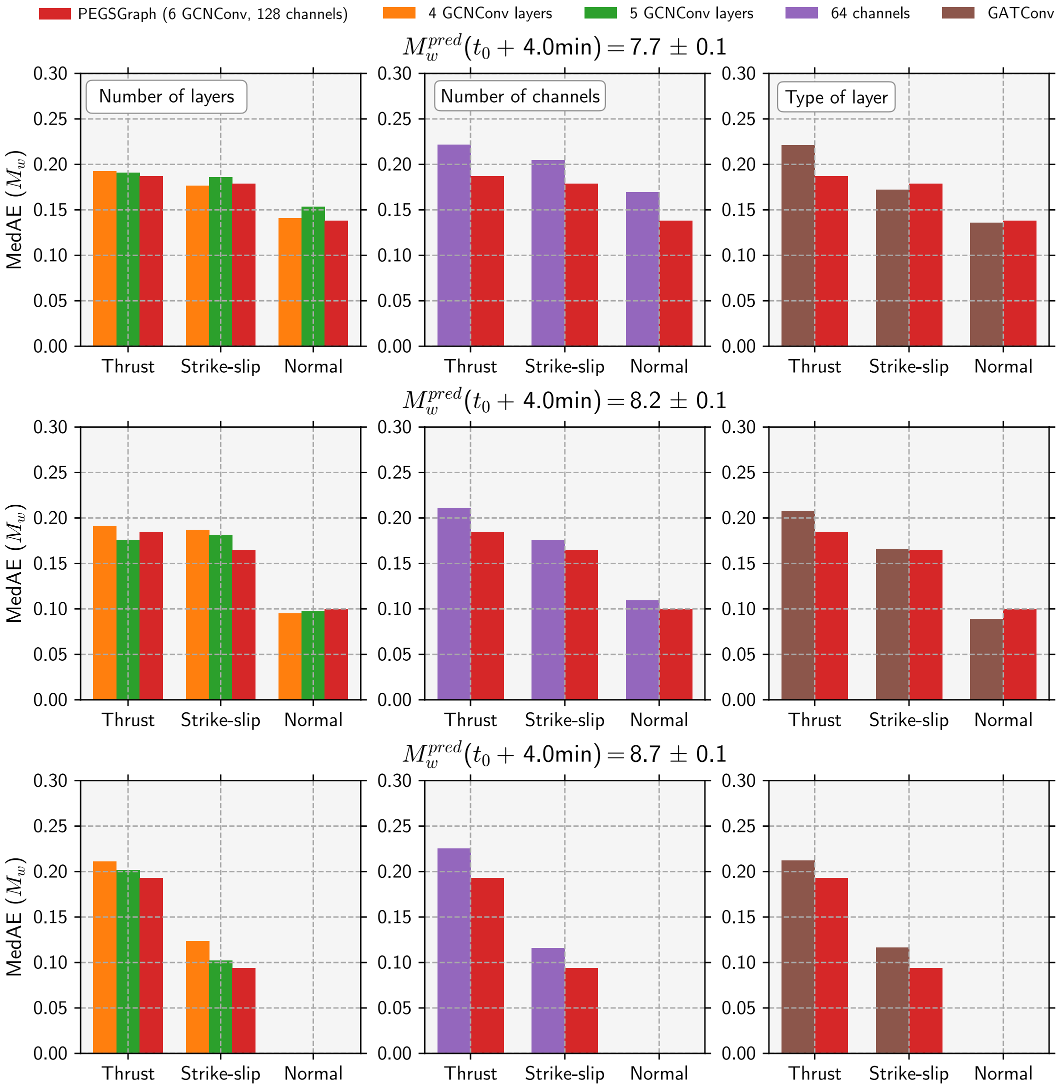

Figure 10. Median Absolute Error (MedAE) of the magnitude obtained with different GNN architectures. Results are shown for the different event types and predicted magnitudes, measured 4 min after the earthquake onset. A minimum of 10 samples is required to calculate the MedAE.

sensitivity of the model to these structural changes. Overall, we observed minimal differences in performance between the different architectures; however, the specific architecture of PEGSGraph seems to provide the best results.

While GCNs provide an intuitive approach to modeling spatial relationships between seismological stations, we also explore more flexible architectures such as Graph Attention Networks (GATs). GATs incorporate attention mechanisms directly into the graph structure, allowing the model to dynamically weight the importance of connections between stations, rather than relying solely on fixed graph relationships. This ability to assign dynamic edge weights makes GATs an intermediate approach between GCNs and transformers, offering more flexibility in how stations communicate within the graph.

Transformers, in contrast, remove the need for a predefined graph structure altogether, learning the importance of connections between stations automatically. This makes transformers potentially interesting for applications like earthquake early warning systems (Münchmeyer et al., 2021). By determining how to combine features from different stations dynamically, transformers could address challenges faced by traditional CNN-based models.

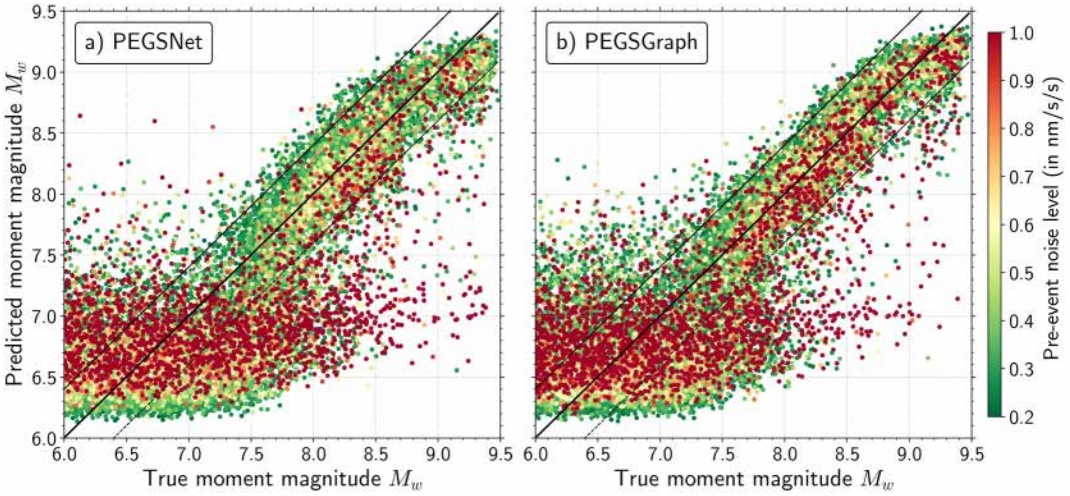

Figure 11. Impact of the pre-event noise level (in nm/s^2) on the performance of the algorithms. This analysis only includes test samples with STF inefficiency <1 (indicating fast onsets) to focus solely on the impact of the noise level. The black lines indicate the ground truth (solid) and the ±0.4 M_w unit range (dashed), that is the range of magnitudes wherein a M_w estimate is considered successful. (a) PEGSNet M_w estimates, compared with the ground truth 4 min after onset time, color-coded by the pre-event noise level. (b) Same as (a) for the PEGSGraph predictions.

However, for this study, we prioritized the use of GCNs and GATs due to the explicit spatial knowledge provided by the seismic station layout, which is central to understanding seismic wave propagation patterns.

Despite testing several architectures using GAT layers instead of GCNConv, we did not observe a significant improvement in performance (an example of performance with GAT layers is shown in Figure 10). However, the adaptability of transformers remains a compelling direction for future work, particularly in regions where seismic networks are evolving rapidly.

4.2. Impact of the Pre-Event Noise Level

We define the noise level as the standard deviation of each waveform recorded during the 5 min-long time-window preceding a real or synthetic earthquake. We compute the median of these noise levels obtained for each station, to get an estimate of the noise level across the entire network of sensors. We restrict the analysis to test samples with STF inefficiency <1 (indicating fast starts) to focus on the impact of the noise level. STF inefficiency is defined as the ratio between the time required to release half of the final moment and its functional form (Meier et al., 2017) (see Figure S9 in Supporting Information S1 for an illustration of the STF inefficiency). For PEGSNet, events characterized by a noise level exceeding 1 nm/s^2 (highlighted by red dots, indicating a strong background seismic noise, or surface waves generated by regional or teleseismic events) most often exhibit inaccuracies in magnitude prediction (Figure 11a). This limitation suggests that higher noise levels adversely affect PEGSNet's ability to provide precise estimates. In contrast, PEGSGraph demonstrates enhanced performance under similar noisy conditions. This improvement is noticeable in Figure 11b, where a higher number of test samples with high noise levels falls within the expected magnitude range. This highlights PEGSGraph's robustness in dealing with noisy data and its ability to achieve more accurate magnitude estimations.

4.3. Impact of Earthquake Location

To assess the effect of the earthquake's location relative to the station network on prediction accuracy, we consider six locations (denoted T1 to T6 in Figure 12a) along the 30 km isodepth of the subducting slab, according to the Slab2.0 model (Hayes et al., 2018). We then calculate the corresponding synthetic PEGS considering the strike and dip angles consistent with the Slab2.0 model at the six considered locations, a rake angle of 90°, and a magnitude of 7.8 (which corresponds to PEGSNet's sensitivity limit). For each synthetic PEGS, we add 1,000 time-windows of empirical noise extracted from the test set. Gathering the magnitude estimations for each of these newly generated test samples allows us to estimate the uncertainty for each of these synthetic thrust earthquakes (Figure 12b).

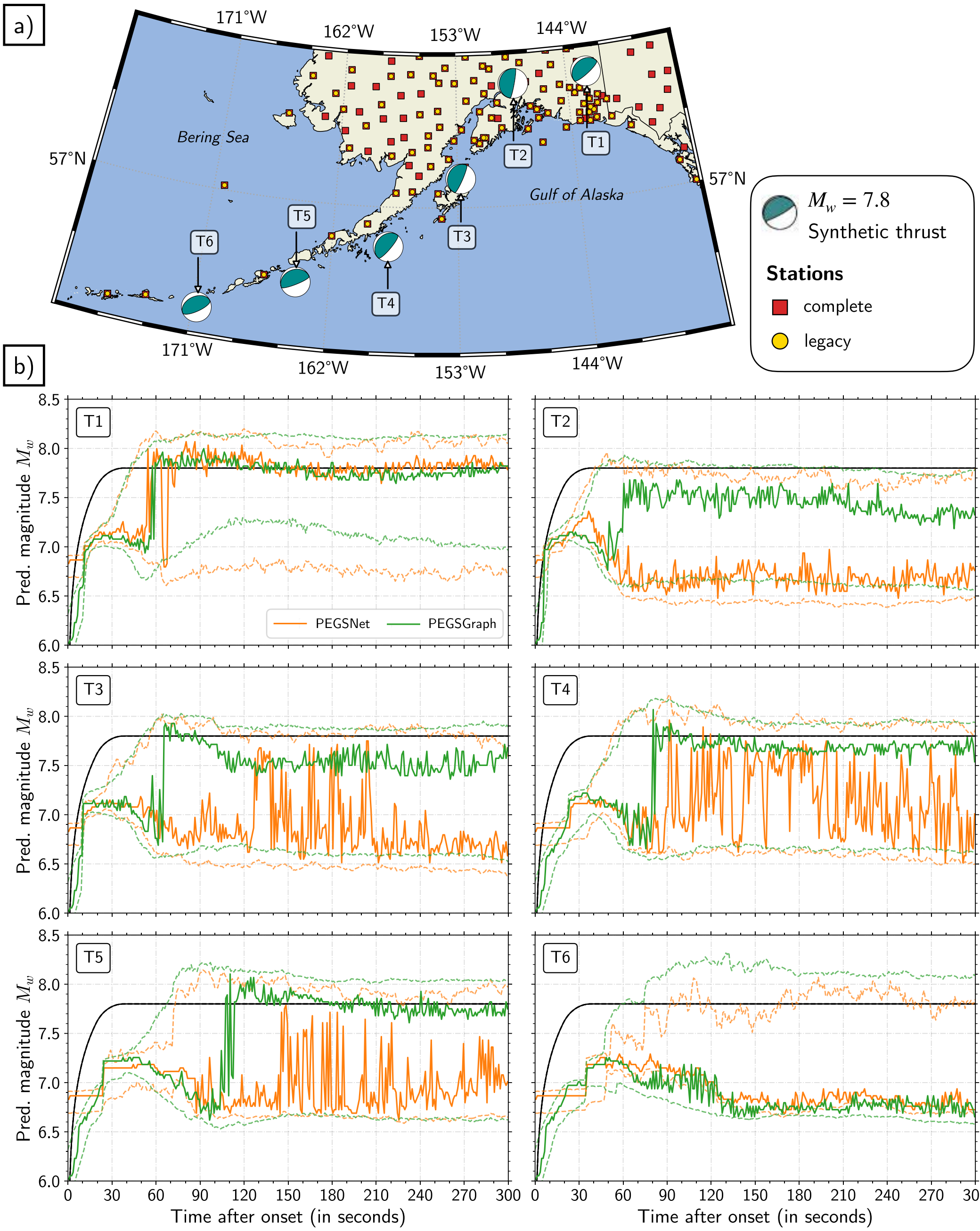

Figure 12. Magnitude estimates for $M_w = 7.8$ synthetic thrust earthquakes, based on the Slab2.0 model (Hayes et al., 2018). (a) Location of the six synthetic sources, along the 30 km isodepth of the subducting slab. (b) Magnitude estimates from synthetic waveforms (synthetic PEGS based on Slab2.0 parameters, augmented with 1,000 noise time-windows from the test set). The orange lines indicate the distribution mode (solid) and the 5th and 95th percentiles (dashed) for PEGSNet, while the green lines represent the mode (solid) and the 5th and 95th percentiles (dashed) for PEGSGraph. The ground truth, moment-rate source time function is a triangular-shaped function with typical duration $T = \left(M_0/10^{16}\ \text{N.m}\right)^{1/3}$ (Houston, 2001), with M_0 the seismic moment. The corresponding accumulated moment magnitude is provided in black as a reference for each event.

For location T1, both models show similar results, with solutions converging to the ground truth 60 s after the earthquake onset. However, as we move further east (hence farther away from the station network), the accuracy of PEGSNet's solutions rapidly decreases. The lower accuracy of PEGSNet's solutions for locations T2 and T3 may be attributed to low dip angles (8.6° and 12.0°, respectively) compared to the normal distribution used in the training database ($\mathcal{N}(\mu = 25°, \sigma = 7°)$, Figure S2 in Supporting Information S1). In contrast, PEGSGraph maintains a more stable distribution mode, providing a solution around $M_w \sim 7.5$ for events T2 and T3. We observe an oscillation in PEGSNet distribution modes between the target value and $M_w \sim 6.7$ for events T3, T4, and T5. For events T4 and T5, PEGSGraph offers a robust and stable solution, with the predicted magnitude converging around the target value at approximately 80 and 110 s, respectively. Finally, for the easternmost location T6, neither of the two algorithms provides a solution close to the target magnitude. The T6 radiation pattern (and associated optimal locations for PEGS detection) is indeed directed toward the Bering Sea, well away from the network of sensors.

We show the corresponding "legacy" performances in Figure S10 in Supporting Information S1. These results, for both configurations, highlight the critical importance of a well-distributed and dense station network on the reliability and accuracy of the magnitude estimations. We note that in regions far from the existing station network, PEGSGraph performs better than PEGSNet, providing more consistent and stable magnitude estimates despite the sparse station coverage.

4.4. Impact of the Source Time Function Onset

The source of the Sand Point event not only presents a significantly lower final seismic moment than the other events, but also a slower onset. Fifteen seconds after the initial onset, the event had only reached $\mathrm{M_w} = 7.0$ (see the SCARDEC reference curves in Figure 7d). This slow evolution hampers the effectiveness of PEGS generation, complicating the precise extraction of source parameters. Nevertheless, the solution produced by PEGSGraph shows a stable solution close to the target magnitude around 150 s after onset time.

Figure 13 further illustrates the impact of the STF efficiency on the predictive capabilities of PEGSNet and PEGSGraph. Both models encounter difficulties in accurately characterizing the source when the STF exhibits a very slow onset. In contrast, medium-to-fast onsets lead to precise M_w estimates for test samples that are above the sensitivity threshold. Compared to PEGSNet, PEGSGraph demonstrates improved performance by providing more samples within the expected magnitude range, as indicated by the red lines. Solutions for sources with a very slow STF onset (inefficiency around 2) appear to be more accurate (Figure 13d). Moreover, a noticeable reduction in the sensitivity threshold, from 7.8 to 7.7, is observed with PEGSGraph (Figure 13c), highlighting its enhanced ability to detect and accurately estimate the magnitude of smaller events.

4.5. PEGSGraph Generalization

One of the key strengths of our GNN-based approach with PEGSGraph is its flexibility in accommodating changes to the sensor network. Unlike CNNs, which require complete retraining when new stations are added, PEGSGraph can incorporate additional sensors—whether permanent or temporary—without the need for extensive model retraining. This adaptability is particularly advantageous for evolving seismic networks, where new stations may be deployed to enhance coverage.

Adapting PEGSGraph to other regions requires adjustments due to potential differences in instrumentation (e.g., sensors density, spatial distribution and sensitivity) and in recorded signals (e.g., seismic noise levels and tectonic settings). Some regions may have sparse or dense networks, which affects graph construction, and varying installation qualities (e.g., underground vaults vs. surface installations) that influence the signal observability. Additionally, regional tectonic settings (such as subduction zones only vs. areas with mixed tectonic activity) impact the PEGS patterns the model must recognize. Retraining is thus essential to ensure accuracy, but starting from scratch is not necessary. Using transfer learning, we can fine-tune PEGSGraph with pre-trained weights, reducing data requirements and expediting adaptation for new regions.

5. Conclusion

We designed PEGSGraph, a graph neural network, to enhance rapid characterization of large earthquakes based on PEGS. We compared its performance to PEGSNet's—a convolutional neural network previously designed for the same task—on a dense seismic network deployed in Alaska and Western Canada. PEGSGraph is able to estimate the magnitude of synthetic earthquakes down to $\mathrm{M_w}$7.6–7.7 (compared to $\mathrm{M_w}$ 7.8 for PEGSNet), and to provide their focal mechanisms (thrust fault, strike-slip fault, or normal fault)—that is, all the necessary ingredients to forecast the amplitude of a potential tsunami wave—within 70 s after the event initiation. Tested on

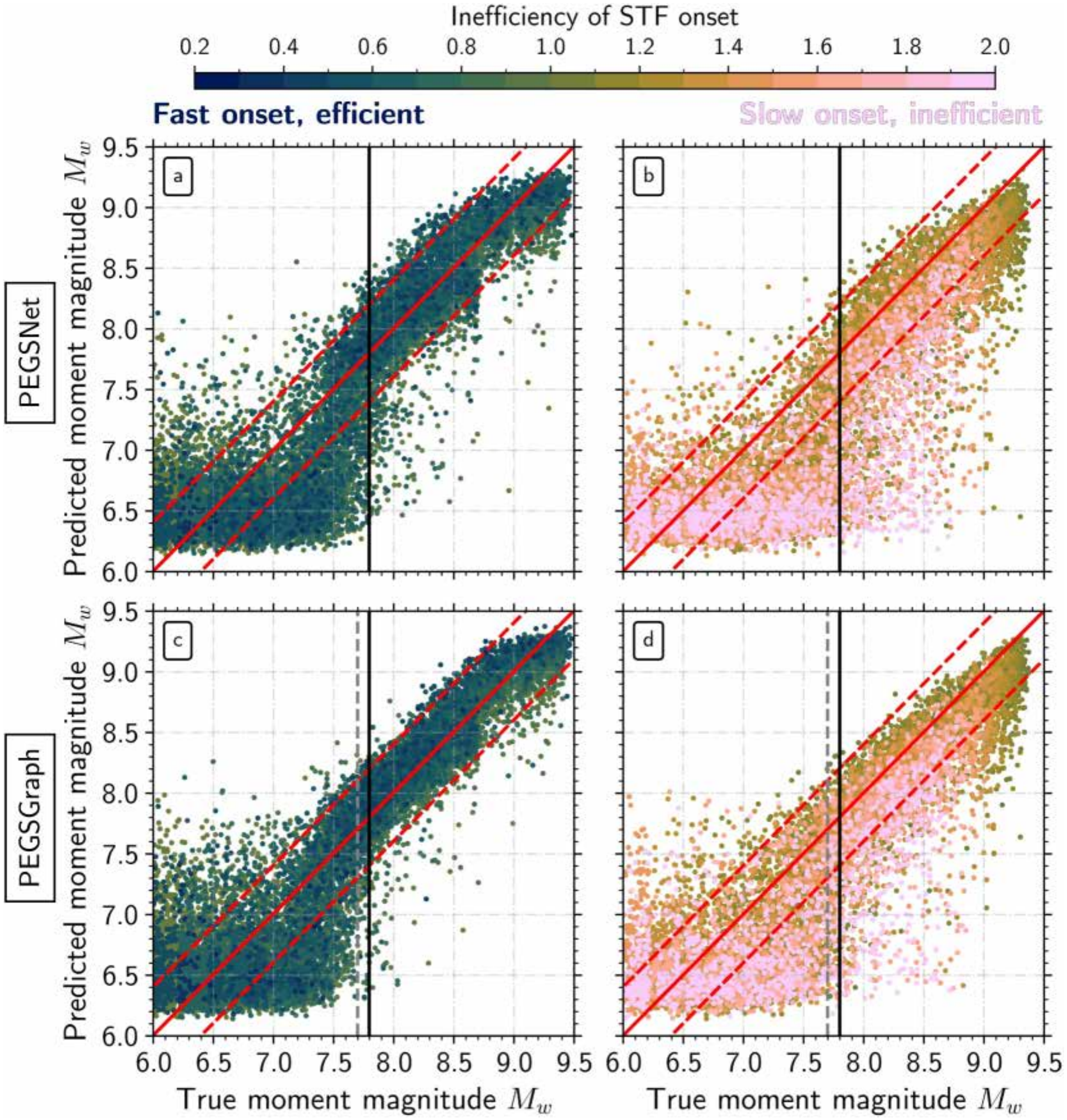

Figure 13. Impact of the STF onset on the performance of the algorithms. We exclude test samples with very high noise (>1 nm/s^2) to focus solely on the impact of STF efficiency. The red lines indicate the ground truth (solid) and the ± 0.4 M_w unit range (dashed), that is the range of magnitudes wherein a M_w estimate is considered successful. The vertical lines represent PEGSNet (solid) and PEGSGraph (dashed) sensitivity thresholds. (a) M_w estimates from PEGSNet, compared to the ground truth 4 min after onset time, for events with STF inefficiency <1 (indicating a fast onset). (b) PEGSNet M_w estimates for events with STF inefficiency >1 (indicating a slow onset). (c) PEGSGraph M_w estimates for events with STF inefficiency <1 (fast onset). (d) PEGSGraph M_w estimates for events with STF inefficiency >1 (slow onset).

real data, PEGSGraph proved capable of providing robust estimates of magnitude and focal mechanism for three $7.8 \leq M_w \leq 8.2$ earthquakes less than 2 min after the event initiations. For the one M_w 7.6 event, PEGSGraph converges toward a solution close to the target magnitude 2 min after onset, while PEGSNet does not converge. Additionally, PEGSGraph's architecture makes it capable to handle non-stationary seismic networks (resulting from addition/removal of sensors). Overall, PEGSGraph appears as a viable, flexible solution for tsunami warning, with performances above state-of-the-art early warning systems.

Data Availability Statement

PEGSGraph (Juhel, Hourcade, & Bletery, 2024) is built and trained using PyTorch (Paszke et al., 2019) and PyTorch Geometric (Fey & Lenssen, 2019). The Python scripts used to generate the input PEGS database and train the graph neural networks described within this paper are available at the following GitLab repository (https://gitlab.com/kjuhel/pegsgraph). The broadband seismic data from the following networks was used in this study (alphabetic order): the Alaska Geophysical Network (AK, Alaska Earthquake Center, Univ. of Alaska Fairbanks, 1987), the National Tsunami Warning Center Alaska Seismic Network (AT, NOAA National Oceanic and Atmospheric Administration (USA), 1967), the Alaska Volcano Observatory (AV, Alaska Volcano Observatory/USGS, 1988), the Canadian National Seismograph Network (CN, Natural Resources Canada, 1975), the Global Seismograph Network (II, Scripps Institution of Oceanography, 1986), the Global Seismograph Network (IU, Albuquerque Seismological Laboratory/USGS, 2014), the USArray Transportable Array (TA, IRIS Transportable Array, 2003). Waveforms were retrieved from IRIS using the Python toolbox ObspyDMT (Hosseini & Sigloch, 2017), and processed with the Python toolbox ObsPy (Beyreuther et al., 2010). SCARDEC source time functions were accessed on http://scardec.projects.sismo.ipgp.fr. Cartopy (https://scitools.org.uk/cartopy), Matplotlib (Hunter, 2007) and perceptually uniform scientific color maps (Crameri, 2023) were used for plotting purposes.

Acknowledgments
We thank Rongjiang Wang for his open-source QSSP algorithm used to compute PEGS synthetics. This work has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (Grant Agreement 949221). This work was granted access to the HPC resources of IDRIS under the allocations 2020-AD011012142, 2021-AP011012536 and 2021-A0101012314 made by GENCI.

References

Alaska Earthquake Center, Univ. of Alaska Fairbanks. (1987). Alaska Geophysical Network [Dataset]. *International Federation of Digital Seismograph Networks*. https://doi.org/10.7914/SN/AK

Alaska Volcano Observatory/USGS. (1988). Alaska Volcano Observatory [Dataset]. *International Federation of Digital Seismograph Networks*. https://doi.org/10.7914/SN/AV

Albuquerque Seismological Laboratory/USGS. (2014). Global Seismograph Network (GSN—IRIS/USGS) [Dataset]. *International Federation of Digital Seismograph Networks*. https://doi.org/10.7914/SN/IU

Allen, R. M., & Kanamori, H. (2003). The potential for earthquake early warning in southern California. *Science*, *300*(5620), 786–789. https://doi.org/10.1126/science.1080912

Allen, R. M., & Melgar, D. (2019). Earthquake early warning: Advances, scientific challenges, and societal needs. *Annual Review of Earth and Planetary Sciences*, *47*(1), 361–388. https://doi.org/10.1146/annurev-earth-053018-060457

Arias, G., Bletery, Q., Licciardi, A., Juhel, K., Ampuero, J. P., & Rouet-Leduc, B. (2023). Rapid source characterization of the Maule earthquake using prompt elasto-gravity signals. *Journal of Geophysical Research: Solid Earth*, *128*(9). https://doi.org/10.1029/2023JB026588

Atz, K., Grisoni, F., & Schneider, G. (2021). Geometric deep learning on molecular representations. *Nature Machine Intelligence*, *3*(12), 1023–1032. https://doi.org/10.1038/s42256-021-00418-8

Bergen, K. J., Chen, T., & Li, Z. (2019). Preface to the focus section on machine learning in seismology. *Seismological Research Letters*, *90*(2A), 477–480. https://doi.org/10.1785/0220190018

Beyreuther, M., Barsch, R., Krischer, L., Megies, T., Behr, Y., & Wassermann, J. (2010). ObsPy: A Python toolbox for seismology. *Seismological Research Letters*, *81*(3), 530–533. https://doi.org/10.1785/gssrl.81.3.530

Bronstein, M. M., Bruna, J., Cohen, T., & Veličković, P. (2021). Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *arXiv preprint arXiv:2104.13478*.

Bronstein, M. M., Bruna, J., LeCun, Y., Szlam, A., & Vandergheynst, P. (2017). Geometric deep learning: Going beyond Euclidean data. *IEEE Signal Processing Magazine*, *34*(4), 18–42. https://doi.org/10.1109/msp.2017.2693418

Busby, R. W., & Aderhold, K. (2020). The Alaska Transportable Array: As built. *Seismological Research Letters*, *91*(6), 3017–3027. https://doi.org/10.1785/0220200154

Chami, I., Abu-El-Haija, S., Perozzi, B., Ré, C., & Murphy, K. (2022). Machine learning on graphs: A model and comprehensive taxonomy. *Journal of Machine Learning Research*, *23*(89), 1–64.

Crameri, F. (2023). Scientific colour maps (8.0.1) [Software]. https://doi.org/10.5281/zenodo.8409685

Crowell, B. W., & Melgar, D. (2020). Slipping the Shumagin gap: A kinematic coseismic and early afterslip model of the Mw 7.8 Simeonof Island, Alaska, earthquake. *Geophysical Research Letters*, *47*(19). https://doi.org/10.1029/2020gl090308

Crowell, B. W., Schmidt, D. A., Bodin, P., Vidale, J. E., Gomberg, J., Renate Hartog, J., et al. (2016). Demonstration of the Cascadia G-FAST geodetic earthquake early warning system for the Nisqually, Washington, earthquake. *Seismological Research Letters*, *87*(4), 930–943. https://doi.org/10.1785/0220150255

Duputel, Z., Rivera, L., Kanamori, H., & Hayes, G. (2012). W phase source inversion for moderate to large earthquakes (1990–2010). *Geophysical Journal International*, *189*(2), 1125–1147. https://doi.org/10.1111/j.1365-246x.2012.05419.x

Duputel, Z., Rivera, L., Kanamori, H., Hayes, G., Hirsorn, B., & Weinstein, S. (2011). Real-time W phase inversion during the 2011 Tohoku-Oki earthquake. *Earth Planets and Space*, *63*(7), 535–539. https://doi.org/10.5047/eps.2011.05.032

Ekström, G., Nettles, M., & Dziewoński, A. (2012). The global CMT project 2004–2010: Centroid-moment tensors for 13,017 earthquakes. *Physics of the Earth and Planetary Interiors*, *200*, 1–9. https://doi.org/10.1016/j.pepi.2012.04.002

Fan, W., Ma, Y., Li, Q., He, Y., Zhao, E., Tang, J., & Yin, D. (2019). Graph neural networks for social recommendation. In *The World Wide Web Conference* (pp. 417–426).

Fey, M., & Lenssen, J. E. (2019). Fast graph representation learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*. https://doi.org/10.48550/arXiv.1903.02428

Grapenthin, R., Johanson, I. A., & Allen, R. M. (2014). Operational real-time GPS-enhanced earthquake early warning. *Journal of Geophysical Research: Solid Earth*, *119*(10), 7944–7965. https://doi.org/10.1002/2014jb011400

Harms, J., Ampuero, J.-P., Barsuglia, M., Chassande-Mottin, E., Montagner, J.-P., Somala, S., & Whiting, B. (2015). Transient gravity perturbations induced by earthquake rupture. *Geophysical Journal International*, *201*(3), 1416–1425. https://doi.org/10.1093/gji/ggv090

Hayes, G. P., Moore, G. L., Portner, D. E., Hearne, M., Flamme, H., Furtney, M., & Smoczyk, G. M. (2018). Slab2, a comprehensive subduction zone geometry model. *Science*, *362*(6410), 58–61. https://doi.org/10.1126/science.aat4723

Heaton, T. H. (2017). Correspondence: Response of a gravimeter to an instantaneous step in gravity. *Nature Communications*, *8*(1), 966. https://doi.org/10.1038/s41467-017-01348-z

Herman, M. W., & Furlong, K. P. (2021). Triggering an unexpected earthquake in an uncoupled subduction zone. *Science Advances*, *7*(13). https://doi.org/10.1126/sciadv.abf7590

Hosseini, K., & Sigloch, K. (2017). ObspyDMT: A Python toolbox for retrieving and processing large seismological data sets. *Solid Earth*, *8*(5), 1047–1070. https://doi.org/10.5194/se-8-1047-2017

Houston, H. (2001). Influence of depth, focal mechanism, and tectonic setting on the shape and duration of earthquake source time functions. *Journal of Geophysical Research*, *106*(B6), 11137–11150. https://doi.org/10.1029/2000jb900468

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, *9*(03), 90–95. https://doi.org/10.1109/MCSE.2007.55

IRIS Transportable Array. (2003). USArray Transportable Array [Dataset]. *International Federation of Digital Seismograph Networks*. https://doi.org/10.7914/SN/TA

Joshi, A., Raman, B., Mohan, C. K., & Cenkeramaddi, L. R. (2024). Application of a new machine learning model to improve earthquake ground motion predictions. *Natural Hazards*, *120*(1), 729–753. https://doi.org/10.1007/s11069-023-06230-4

Jost, M. U., & Herrmann, R. (1989). A student's guide to and review of moment tensors. *Seismological Research Letters*, *60*(2), 37–57. https://doi.org/10.1785/gssrl.60.2.37

Juhel, K., Bletery, Q., Licciardi, A., Vallée, M., Hourcade, C., & Michel, T. (2024). Fast and full characterization of large earthquakes from prompt elastogravity signals. *Communications Earth & Environment*, *5*(1), 561. https://doi.org/10.1038/s43247-024-01725-9

Juhel, K., Duputel, Z., Rivera, L., & Vallée, M. (2023). Early source characterization of large earthquakes using W phase and prompt elastogravity signals. *Seismological Research Letters*, *95*(3), 1558–1568. https://doi.org/10.1785/0220230195

Juhel, K., Hourcade, C., & Bletery, Q. (2024). PEGSGraph: GNN-based rapid earthquake characterization using prompt elastogravity signals [Software]. *Zenodo*. https://doi.org/10.5281/zenodo.12751347

Juhel, K., Licciardi, A., & Bletery, Q. (2024). PEGSNet: CNN-Based rapid earthquake characterization using prompt elastogravity signals [Software]. *Zenodo*. https://doi.org/10.5281/zenodo.10495626

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., et al. (2021). Highly accurate protein structure prediction with alphafold. *Nature*, *596*(7873), 583–589. https://doi.org/10.1038/s41586-021-03819-2

Kanamori, H. (1993). W phase. *Geophysical Research Letters*, *20*(16), 1691–1694. https://doi.org/10.1029/93gl01883

Kanamori, H., & Rivera, L. (2008). Source inversion of W phase: Speeding up seismic tsunami warning. *Geophysical Journal International*, *175*(1), 222–238. https://doi.org/10.1111/j.1365-246x.2008.03887.x

Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Krabbenhoeft, A., von Huene, R., Miller, J. J., Lange, D., & Vera, F. (2018). Strike-slip 23 January 2018 M_W 7.9 Gulf of Alaska rare intraplate earthquake: Complex rupture of a fracture zone system. *Scientific Reports*, *8*(1), 13706. https://doi.org/10.1038/s41598-018-32071-4

Lara, P., Bletery, Q., Ampuero, J.-P., Inza, A., & Tavera, H. (2023). 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. https://doi.org/10.1029/2023jb026575

Licciardi, A., Bletery, Q., Rouet-Leduc, B., Ampuero, J.-P., & Juhel, K. (2022). Instantaneous tracking of earthquake growth with elastogravity signals. *Nature*, *606*(7913), 1–6. https://doi.org/10.1038/s41586-022-04672-7

Liu, C., Lay, T., & Xiong, X. (2022). The 29 July 2021 MW 8.2 Chignik, Alaska peninsula earthquake rupture inferred from seismic and geodetic observations: Re-rupture of the western 2/3 of the 1938 rupture zone. *Geophysical Research Letters*, *49*(4). https://doi.org/10.1029/2021gl096004

Liu, C., Lay, T., Xiong, X., & Wen, Y. (2020). Rupture of the 2020 M_W 7.8 earthquake in the Shumagin gap inferred from seismic and geodetic observations. *Geophysical Research Letters*, *47*(22). https://doi.org/10.1029/2020gl090806

Liu, Y., Zhao, Q., & Wang, Y. (2024). Peak ground acceleration prediction for on-site earthquake early warning with deep learning. *Scientific Reports*, *14*(1), 5485. https://doi.org/10.1038/s41598-024-56004-6

McBrearty, I. W., & Beroza, G. C. (2022). Earthquake location and magnitude estimation with graph neural networks. In *2022 IEEE International Conference on Image Processing (ICIP)* (pp. 3858–3862). IEEE. https://doi.org/10.1109/ICIP46576.2022.9897468

McBrearty, I. W., & Beroza, G. C. (2023). Earthquake phase association with graph neural networks. *Bulletin of the Seismological Society of America*, *113*(2), 524–547. ([physics]). https://doi.org/10.1785/0120220182

Meier, M.-A., Ampuero, J., & Heaton, T. H. (2017). The hidden simplicity of subduction megathrust earthquakes. *Science*, *357*(6357), 1277–1281. https://doi.org/10.1126/science.aan5643

Melgar, D., Allen, R. M., Riquelme, S., Geng, J., Bravo, F., Baez, J. C., et al. (2016). Local tsunami warnings: Perspectives from recent large events. *Geophysical Research Letters*, *43*(3), 1109–1117. https://doi.org/10.1002/2015gl067100

Minson, S. E., Meier, M.-A., Baltay, A. S., Hanks, T. C., & Cochran, E. S. (2018). The limits of earthquake early warning: Timeliness of ground motion estimates. *Science Advances*, *4*(3), eaaq0504. https://doi.org/10.1126/sciadv.aaq0504

Minson, S. E., Murray, J. R., Langbein, J. O., & Gomberg, J. S. (2014). Real-time inversions for finite fault slip models and rupture geometry based on high-rate GPS data. *Journal of Geophysical Research: Solid Earth*, *119*(4), 3201–3231. https://doi.org/10.1002/2013jb010622

Montagner, J.-P., Juhel, K., Barsuglia, M., Ampuero, J. P., Chassande-Mottin, E., Harms, J., et al. (2016). Prompt gravity signal induced by the 2011 Tohoku-Oki earthquake. *Nature Communications*, *7*(1), 1–7. https://doi.org/10.1038/ncomms13349

Mori, N., Satake, K., Cox, D., Goda, K., Catalan, P. A., Ho, T.-C., et al. (2022). Giant tsunami monitoring, early warning and hazard assessment. *Nature Reviews Earth and Environment*, *3*(9), 557–572. https://doi.org/10.1038/s43017-022-00327-3

Mousavi, S. M., & Beroza, G. C. (2022). Deep-learning seismology. *Science*, *377*(6607), eabm4470. https://doi.org/10.1126/science.abm4470

Mousavi, S. M., & Beroza, G. C. (2023). Machine learning in earthquake seismology. *Annual Review of Earth and Planetary Sciences*, *51*(1), 105–129. https://doi.org/10.1146/annurev-earth-071822-100323

Münchmeyer, J., Bindi, D., Leser, U., & Tilmann, F. (2021). The transformer earthquake alerting model: A new versatile approach to earthquake early warning. *Geophysical Journal International*, *225*(1), 646–656. https://doi.org/10.1093/gji/ggaa609

Natural Resources Canada. (1975). Canadian National Seismograph Network [Dataset]. *International Federation of Digital Seismograph Networks*. https://doi.org/10.7914/SN/CN

Nicolsky, D. J., Suleimani, E., & Koehler, R. D. (2017). *Tsunami inundation maps for the city of sand point, Alaska*. State of Alaska, Department of Natural Resources, Division of Geological.

NOAA National Oceanic and Atmospheric Administration (USA). (1967). National Tsunami Warning Center Alaska Seismic Network [Dataset]. *International Federation of Digital Seismograph Networks*. https://doi.org/10.7914/SN/AT

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., et al. (2019). Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, *32*. https://doi.org/10.48550/arXiv.1912.01703

Renou, J., Vallée, M., & Dublanchet, P. (2019). How does seismic rupture accelerate? Observational insights from earthquake source time functions. *Journal of Geophysical Research: Solid Earth*, *124*(8), 8942–8952. https://doi.org/10.1029/2019JB018045

Rivera, L., & Kanamori, H. (2014). Diagnosing source geometrical complexity of large earthquakes. *Pure and Applied Geophysics*, *171*(10), 2819–2840. https://doi.org/10.1007/s00024-013-0769-4

Ruppert, N. A., Rollins, C., Zhang, A., Meng, L., Holtkamp, S. G., West, M. E., & Freymueller, J. T. (2018). Complex faulting and triggered rupture during the 2018 M_W 7.9 offshore Kodiak, Alaska, earthquake. *Geophysical Research Letters*, *45*(15), 7533–7541. https://doi.org/10.1029/2018gl078931

Ruppert, N. A., & West, M. E. (2020). The impact of USArray on earthquake monitoring in Alaska. *Seismological Research Letters*, *91*(2A), 601–610. https://doi.org/10.1785/0220190227

Saad, O. M., Helmy, I., Mohammed, M., Savvaidis, A., Chatterjee, A., & Chen, Y. (2024). Deep learning peak ground acceleration prediction using single-station waveforms. *IEEE Transactions on Geoscience and Remote Sensing*, *62*, 1–13. https://doi.org/10.1109/tgrs.2024.3367725

Santellanes, S. R., Melgar, D., Crowell, B. W., & Lin, J.-T. (2022). Potential megathrust co-seismic slip during the 2020 Sand Point, Alaska strike-slip earthquake. *ESS Open Archive*. https://doi.org/10.1002/essoar.10508002.2

Scripps Institution of Oceanography. (1986). Global Seismograph Network—IRIS/IDA [Dataset]. *International Federation of Digital Seismograph Networks*. https://doi.org/10.7914/SN/II

Shang, L., Zhang, Y., Zhang, D., & Wang, D. (2020). Fauxward: A graph neural network approach to fauxtography detection using social media comments. *Social Network Analysis and Mining*, *10*(1), 76. https://doi.org/10.1007/s13278-020-00689-w

Vallée, M., Ampuero, J. P., Juhel, K., Bernard, P., Montagner, J.-P., & Barsuglia, M. (2017). Observations and modeling of the elastogravity signals preceding direct seismic waves. *Science*, *358*(6367), 1164–1168. https://doi.org/10.1126/science.aao0746

Vallée, M., & Douet, V. (2016). A new database of source time functions (STFs) extracted from the SCARDEC method. *Physics of the Earth and Planetary Interiors*, *257*, 149–157. https://doi.org/10.1016/j.pepi.2016.05.012

Vallée, M., & Juhel, K. (2019). Multiple observations of the prompt elastogravity signals heralding direct seismic waves. *Journal of Geophysical Research: Solid Earth*, *124*(3), 2970–2989. https://doi.org/10.1029/2018jb017130

Van Den Ende, M. P. A., & Ampuero, J. (2020). Automated seismic source characterization using deep graph neural networks. *Geophysical Research Letters*, *47*(17), e2020GL088690. https://doi.org/10.1029/2020GL088690

Wang, R., Heimann, S., Zhang, Y., Wang, H., & Dahm, T. (2017). Complete synthetic seismograms based on a spherical self-gravitating Earth model with an atmosphere–ocean–mantle–core structure. *Geophysical Journal International*, *210*(3), 1739–1764. https://doi.org/10.1093/gji/ggx259

Whitmore, P., Benz, H., Bolton, M., Crawford, G., Dengler, L., Fryer, G., et al. (2008). NOAA/West coast and Alaska tsunami warning center Pacific Ocean response criteria. *Science of Tsunami Hazards*, *27*(2), 1–19.

Wirth, E. A., Sahakian, V. J., Wallace, L. M., & Melnick, D. (2022). The occurrence and hazards of great subduction zone earthquakes. *Nature Reviews Earth and Environment*, *3*(2), 125–140. https://doi.org/10.1038/s43017-021-00245-w

Wu, Y.-M., & Zhao, L. (2006). Magnitude estimation using the first three seconds p-wave amplitude in earthquake early warning. *Geophysical Research Letters*, *33*(16). https://doi.org/10.1029/2006gl026871

Yano, K., Shiina, T., Kurata, S., Kato, A., Komaki, F., Sakai, S., & Hirata, N. (2021). Graph-partitioning based convolutional neural network for earthquake detection using a seismic array. *Journal of Geophysical Research: Solid Earth*, *126*(5), e2020JB020269. https://doi.org/10.1029/2020JB020269

Ye, L., Bai, Y., Si, D., Lay, T., Cheung, K. F., & Kanamori, H. (2022). Rupture model for the 29 July 2021 M_W 8.2 Chignik, Alaska earthquake constrained by seismic, geodetic, and tsunami observations. *Journal of Geophysical Research: Solid Earth*, *127*(7). https://doi.org/10.1029/2021jb023676

Zhang, S., Wang, R., Dahm, T., Zhou, S., & Heimann, S. (2020). Prompt elasto-gravity signals (PEGS) and their potential use in modern seismology. *Earth and Planetary Science Letters*, *536*, 116150. https://doi.org/10.1016/j.epsl.2020.116150

Zhang, X., Reichard-Flynn, W., Zhang, M., Hirn, M., & Lin, Y. (2022). Spatiotemporal graph convolutional networks for earthquake source characterization. *Journal of Geophysical Research: Solid Earth*, *127*(11), e2022JB024401. https://doi.org/10.1029/2022JB024401

Zhu, W., Tai, K. S., Mousavi, S. M., Bailis, P., & Beroza, G. C. (2022). An end-to-end earthquake detection method for Joint phase Picking and association using deep learning. *Journal of Geophysical Research: Solid Earth*, *127*(3), e2021JB023283. https://doi.org/10.1029/2021JB023283