SEISTAILFLOW: TAIL-AWARE CONDITIONAL FLOW MODELING FOR RELIABLE SEISMIC PHASE PICKING

Anonymous authors
Paper under double-blind review

ABSTRACT

Seismic phase picking is fundamental to automated earthquake monitoring, yet existing deep learning pickers still exhibit heavy-tailed error distributions, where a small fraction of large arrival-time errors can compromise the reliability of downstream phase association and event localization. We propose **SeisTailFlow**, a tail-aware conditional flow model for reliable phase picking. Rather than treating large prediction errors merely as residual uncertainty, SeisTailFlow uses out-of-sample residuals to identify waveform conditions associated with elevated error risk and explicitly models their conditional residual structure alongside dominant arrival-time patterns. Technically, SeisTailFlow builds parallel Main and Tail Flows and integrates residual-based tail supervision, background-referenced spectral enhancement, extreme-value regularization, and adaptive vector-field routing to jointly capture dominant conditional structure and high-error-related residual behavior. This design preserves accurate modeling of dominant arrival-time patterns while providing additional capacity for scarce high-error regimes and large arrival-time deviations. Experiments on synthetic data and the STEAD dataset show that SeisTailFlow improves overall picking accuracy and probabilistic prediction quality while substantially reducing tail errors and large-error events, with particularly robust performance under low-SNR conditions.

1 INTRODUCTION

Seismic phase picking is a fundamental component of automated earthquake monitoring. Reliable estimation of P- and S-wave arrival times is critical for downstream event association, source localization, and event characterization, while large arrival-time errors can degrade the reliability of earthquake association and source-parameter estimation (Di Stefano et al., 2006; Si et al., 2024). Recent deep learning methods have progressively improved the use of seismic waveform information through end-to-end waveform representation (Zhu & Beroza, 2019), long-range temporal context modeling (Mousavi et al., 2020), and multi-task or multi-station information integration (Zhu et al., 2022), bringing substantial gains in overall phase-picking accuracy (Ross et al., 2018; Soto & Schurr, 2021).

However, high overall accuracy does not necessarily imply reliable phase picking. Existing methods typically produce arrival-time errors concentrated near zero, while a small fraction of large errors persist and form relatively heavy prediction-error tails (Bornstein et al., 2024; Niksejel & Zhang, 2024). Such errors occur more frequently when phase onsets are weak or obscured by background noise, particularly under low signal-to-noise ratio (SNR) conditions (Münchmeyer et al., 2022). In these cases, arrival-time signatures become less distinguishable from surrounding waveform fluctuations, increasing the risk of large picking errors that can compromise downstream phase association and source localization despite favorable average performance (Husen & Hardebeck, 2010; Di Giacomo et al., 2025).

Recent studies have incorporated distributional information into seismic phase detection and arrival-time modeling, enabling probabilistic predictions (Armstrong et al., 2023; Park et al., 2025b). However, these approaches primarily focus on uncertainty estimation and calibration, while waveform-dependent structure associated with large prediction errors is typically not modeled explicitly. Moreover, although such large errors are relatively rare, they are not necessarily purely random (Husen & Hardebeck, 2010; Di Giacomo et al., 2025). Their occurrence may depend on waveform condi-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

tions that are insufficiently captured by dominant prediction patterns (Sun et al., 2023; Niksejel & Zhang, 2024). This suggests that prediction-error tails may contain structured waveform-dependent information that can be exploited to improve phase-picking reliability.

Learning from such high-error regimes is nevertheless challenging. First, observations associated with large prediction errors are inherently scarce, providing limited supervision for high-error-aware modeling. Second, large arrival-time errors frequently occur under weak or noisy waveform conditions, where informative phase-related structures are more difficult to extract. Third, it remains challenging to jointly preserve accurate modeling of dominant arrival-time patterns while allocating sufficient modeling capacity to waveform conditions associated with elevated error risk.

We propose **SeisTailFlow**, a tail-aware conditional flow model for reliable seismic phase picking. Rather than treating large arrival-time errors as irreducible random perturbations, SeisTailFlow uses prediction residuals to identify waveform conditions associated with elevated error risk and explicitly models their conditional residual structure alongside dominant arrival-time patterns. The model is built upon parallel Main Flow and Tail Flow modules: the Main Flow learns dominant phase-arrival patterns from regular waveforms, while the Tail Flow provides additional modeling capacity for waveform conditions associated with large deviations.

Within the Tail Flow, we develop a **Seismic Tail Information Screening (Seis TIS)** module to identify informative high-error cases from noisy waveforms. Seis TIS constructs targeted tail supervision from the upper tail of the out-of-sample prediction-error distribution, thereby identifying waveform conditions associated with elevated large-error risk. To improve modeling under low-SNR conditions, we further design a spectral-enhancement network that uses the background-noise spectrum as a reference to emphasize phase-related frequency structures, providing complementary information that may be difficult to capture from time-domain representations alone. Moreover, to alleviate the scarcity of high-error observations, we impose a generalized Pareto distribution (GPD) constraint on high-threshold residual exceedances during optimization (Balkema & de Haan, 1974; Pickands, 1975), enabling more effective learning from limited extreme-error samples.

To integrate dominant and high-error-related conditional structure, we introduce a state- and risk-aware adaptive routing mechanism that dynamically fuses the Main and Tail conditional vector fields during continuous transport. Unlike prediction-level ensembles that combine already formed predictions or predictive distributions (Park et al., 2025a), our router operates directly on the transport dynamics, adjusting the relative contributions of the two flows according to the evolving transport state and waveform-conditioned large-error risk. This allows the Main Flow to dominate under regular conditions while progressively increasing the contribution of the Tail Flow when the inferred error risk is high, enabling both pathways to jointly shape a unified conditional arrival-time distribution. In this way, SeisTailFlow preserves accurate modeling of dominant arrival-time patterns while more effectively reducing large arrival-time errors. The main contributions of this work are summarized as follows:

- We propose **SeisTailFlow**, a tail-aware conditional flow framework that jointly models dominant arrival-time patterns and high-error-related residual structure to improve picking accuracy and reliability.
- We develop a high-error-aware learning strategy combining **Seis TIS** for residual-based tail supervision, background-referenced spectral enhancement for low-SNR waveforms, and GPD regularization for high-threshold residual exceedances.
- We evaluate SeisTailFlow on synthetic data and the STEAD dataset (Mousavi et al., 2019), showing consistent improvements in overall accuracy, tail errors, low-SNR robustness, and probabilistic prediction quality, with ablations confirming the contributions of the proposed components.

2 METHODOLOGY

We propose **SeisTailFlow**, a tail-aware conditional flow framework for reliable seismic phase picking. As illustrated in Figure 1, SeisTailFlow consists of three modules. Module I extracts a shared waveform representation through global normalization, multi-scale temporal encoding, and attention-based aggregation. Module II constructs parallel Main Flow and Tail Flow pathways: the Main Flow learns dominant phase-arrival patterns, while the Tail Flow combines spectrally en-

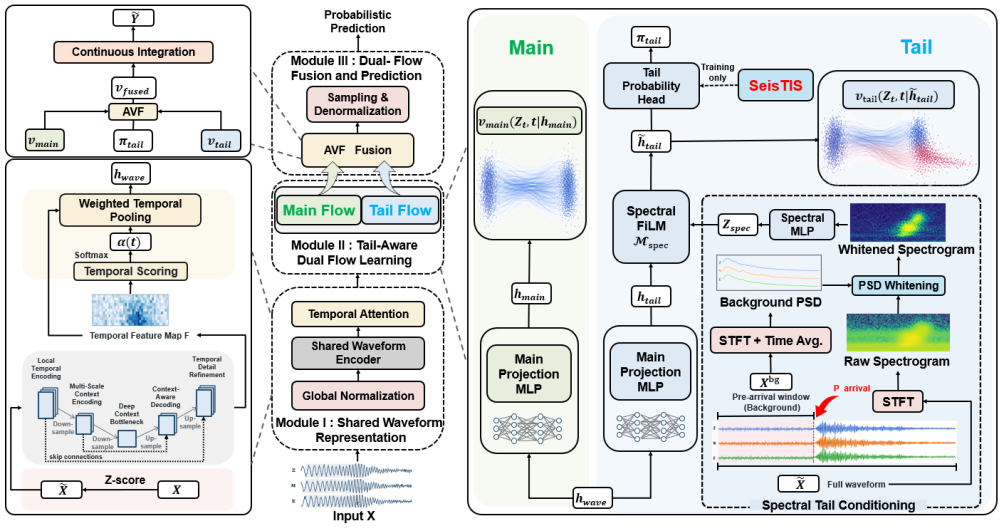


Figure 1: Overview of **SeisTailFlow**.

hanced conditioning with the proposed **Seismic Tail Information Screening (Seis TIS)** module to model conditional residual structure associated with waveform conditions of elevated large-error risk. Module III dynamically fuses the two conditional vector fields according to the current transport state and waveform-conditioned large-error risk, yielding the final conditional phase-arrival distribution through continuous integration and sampling. We review the relevant literature in Appendix A.

2.1 Module I: Shared Waveform Representation

Given a multi-component seismic waveform $\mathbf{X} \in \mathbb{R}^{C\times T}$, SeisTailFlow first constructs a shared waveform representation that serves as a common conditioning basis for the subsequent Main and Tail pathways. Further architectural details are provided in Appendix B.1.

Global Waveform Normalization. To maintain a consistent input scale across waveforms, we standardize each channel using statistics estimated from the training set: $\widetilde{\mathbf{X}}_c = (\mathbf{X}_c - \mu_c)/\sigma_c,\ c = 1,\dots,C$, where μ_c and σ_c denote the global mean and standard deviation of the c-th channel estimated from the training data, respectively, and $\widetilde{\mathbf{X}}$ denotes the normalized waveform.

Shared Waveform Encoding. Inspired by U-Net-style encoder–decoder architectures and their application to seismic phase picking (Ronneberger et al., 2015; Zhu & Beroza, 2019), SeisTailFlow employs a shared waveform encoder E_ψ to extract common phase-related structures from the normalized multi-component waveform $\widetilde{\mathbf{X}}$ and avoid redundant feature extraction across the Main and Tail pathways. The encoder progressively aggregates contextual information across temporal scales through downsampling and restores temporal resolution through upsampling and skip connections, producing a temporal feature map aligned with the input time axis:

$$\mathbf{F} = E_\psi(\widetilde{\mathbf{X}}) = [\mathbf{f}_1,\dots,\mathbf{f}_T], \qquad \mathbf{f}_t \in \mathbb{R}^D, \tag{1}$$

where E_ψ denotes the shared waveform encoder parameterized by ψ, $\widetilde{\mathbf{X}} \in \mathbb{R}^{C\times T}$ is the normalized multi-component input waveform, and C and T denote the number of waveform components and temporal samples, respectively. $\mathbf{F} \in \mathbb{R}^{D\times T}$ is the resulting temporal feature map, where D denotes the feature dimension and $\mathbf{f}_t \in \mathbb{R}^D$ is the local waveform feature at temporal position t.

Temporal Attention Readout. To convert the time-aligned feature map $\mathbf{F}$ into a fixed-dimensional representation, we apply temporal attention to adaptively weight different temporal positions, yielding the global waveform feature $\bar{\mathbf{f}} = \sum_{t=1}^{T} \alpha_t \mathbf{f}_t$, where α_t denotes the normalized attention weight assigned to temporal position t. To retain coarse temporal information from the attention response, we further compute its normalized temporal center ℓ and dispersion d. The resulting shared wave-

form representation, $\mathbf{h}_{\text{wave}} = [\bar{\mathbf{f}}; \ell; d] \in \mathbb{R}^{D+2}$, is used as the common conditioning representation for the subsequent Main and Tail pathways.

2.2 Module II: Tail-Aware Dual Flow Learning

Dual Conditional Pathways. From the shared waveform representation $\mathbf{h}_{\text{wave}}$, SeisTailFlow constructs parallel Main and Tail conditioning pathways. The Main Projection produces $\mathbf{h}_{\text{main}}$ to represent dominant arrival-time patterns, while the Tail Projection produces $\mathbf{h}_{\text{tail}}$ to provide specialized conditioning for waveform regimes associated with large arrival-time deviations. Implementation details are provided in Appendix B.2.

2.2.1 Main Flow Learning

The Main Flow learns conditional transport dynamics associated with dominant phase-arrival patterns. Given $\mathbf{h}_{\text{main}}$, the Main Flow starts from $z_0^{\text{main}} \sim \mathcal{N}(0,1)$ and follows a probability path to z_1^{main} under flow matching. Conditioned on the current state z_t^{main}, flow time t, and $\mathbf{h}_{\text{main}}$, the Main Flow parameterizes $v_{\theta_{\text{m}}}(z_t^{\text{main}}, t \mid \mathbf{h}_{\text{main}})$ and is trained with conditional flow matching (Tong et al., 2024) to regress the target path velocity u_t^{main}:

$$\mathcal{L}_{\text{FM}}^{\text{main}} = \mathbb{E}_{z_0^{\text{main}}, z_1^{\text{main}}, t}\left[\left\| v_{\theta_{\text{m}}}\left(z_t^{\text{main}}, t \mid \mathbf{h}_{\text{main}}\right) - u_t^{\text{main}} \right\|_2^2\right]. \tag{2}$$

Minimizing this objective enables the Main Flow to learn conditional transport dynamics corresponding to the dominant phase-arrival patterns.

2.2.2 Tail Flow Learning

The Tail Flow provides complementary modeling capacity for waveform regimes associated with large arrival-time deviations. It combines spectral enhancement, Seismic Tail Information Screening (Seis TIS), and extreme-value regularization to strengthen representation and learning under noisy, high-error conditions.

Spectral-Enhanced Tail Conditioning. Under low-SNR conditions, noise can obscure informative phase features in the time domain, while time–frequency representations provide complementary spectral structure (Tang & Chen, 2026). We therefore augment the Tail representation $\mathbf{h}_{\text{tail}}$ with spectral information measured relative to the pre-arrival background noise. Let $\widetilde{\mathbf{X}}_c$ denote the c-th channel of the normalized waveform and $\widetilde{\mathbf{X}}_c^{\text{bg}}$ its pre-arrival background segment. We estimate the channel-wise background power spectral density as

$$P_c(f) = \text{stopgrad}\left[\frac{1}{|\mathcal{T}_{\text{bg}}|}\sum_{\tau \in \mathcal{T}_{\text{bg}}}\left|\text{STFT}(\widetilde{\mathbf{X}}_c^{\text{bg}})_{f,\tau}\right|^2\right], \tag{3}$$

where $\mathcal{T}_{\text{bg}}$ indexes the STFT frames within the pre-arrival background window, and $\text{stopgrad}[\cdot]$ blocks gradient propagation through the background-spectrum estimate.

Following spectral-amplitude normalization (Naskar et al., 2025), we normalize the full-waveform time–frequency representation by

$$A_c(f,\tau) = \frac{\left|\text{STFT}(\widetilde{\mathbf{X}}_c)_{f,\tau}\right|}{\sqrt{P_c(f) + \epsilon}}, \tag{4}$$

where $\epsilon > 0$ ensures numerical stability. This normalization emphasizes time–frequency responses that are strong relative to the background noise. A learnable spectral encoder $S_\omega(\cdot)$ then maps the normalized representation $\mathbf{A} = \{A_c(f,\tau)\}_{c,f,\tau}$ to a compact spectral feature $\mathbf{z}_{\text{spec}} = S_\omega(\mathbf{A})$. The spectral feature $\mathbf{z}_{\text{spec}}$ modulates the Tail condition:

$$\widetilde{\mathbf{h}}_{\text{tail}} = \mathcal{M}_{\text{spec}}\left(\mathbf{h}_{\text{tail}}, \mathbf{z}_{\text{spec}}\right), \tag{5}$$

where $\mathcal{M}_{\text{spec}}$ is a learnable spectral-conditioning module. The resulting $\widetilde{\mathbf{h}}_{\text{tail}}$ combines the shared time-domain representation with complementary background-relative spectral information.

Tail Conditional Flow. Given the spectrally enhanced condition $\widetilde{\mathbf{h}}_{\text{tail}}$, the Tail Flow learns an independent conditional vector field specialized for high-error-related waveform regimes. Unlike the Gaussian base used by the Main Flow, it starts from a variance-standardized Student-t distribution $z_0^{\text{tail}} \sim t_\nu^{\text{std}}$, where ν denotes the degrees of freedom. Conditioned on the current state z_t^{tail}, flow time t, and $\widetilde{\mathbf{h}}_{\text{tail}}$, the Tail Flow parameterizes $v_{\theta_t}(z_t^{\text{tail}}, t \mid \widetilde{\mathbf{h}}_{\text{tail}})$ and is trained with conditional flow matching (Tong et al., 2024):

$$\mathcal{L}_{\text{FM}}^{\text{tail}} = \mathbb{E}_{z_0^{\text{tail}}, z_1^{\text{tail}}, t}\left[\left\| v_{\theta_t}\left(z_t^{\text{tail}}, t \mid \widetilde{\mathbf{h}}_{\text{tail}}\right) - u_t^{\text{tail}} \right\|_2^2\right]. \tag{6}$$

This objective enables the Tail Flow to learn conditional transport dynamics specialized for waveform conditions associated with elevated large-error risk.

Seismic Tail Information Screening. We introduce *Seismic Tail Information Screening* (Seis TIS), which converts out-of-sample prediction errors into targeted supervision for high-error-aware learning. Specifically, Seis TIS first uses a nonlinear *Seis Main predictor*, combining dense timing estimation with residual transport, to obtain arrival-time predictions $\hat{y}_i^{\text{Seis}}$. To reduce bias from in-sample fitting, each prediction is generated via cross-fitting by a predictor that excludes the corresponding sample during training (Wolpert, 1992; Chernozhukov et al., 2018). We then compute the absolute prediction residual, determine a high-quantile threshold from the out-of-sample training residuals, and construct tail-specific supervision:

$$r_i^{\text{Seis}} = \left|y_i - \hat{y}_i^{\text{Seis}}\right|, \qquad u^{\text{Seis}} = Q_\gamma\left(\{r_j^{\text{Seis}}\}_{j=1}^N\right), \qquad q_i^{\text{Seis}} = \mathbb{I}\left(r_i^{\text{Seis}} > u^{\text{Seis}}\right), \tag{7}$$

where $Q_\gamma(\cdot)$ denotes the empirical γ-quantile. Samples with $q_i^{\text{Seis}} = 1$ belong to the upper tail of the out-of-sample prediction-error distribution and define high-error cases used for tail-specific supervision. In this way, Seis TIS identifies waveform conditions associated with elevated prediction-error risk without requiring predefined categories of challenging waveforms. Details of the Seis Main predictor, cross-fitting procedure, and tail-specific supervision are provided in Appendix B.3.

Residual-Driven Tail Learning. Given the tail-specific supervision, the Tail pathway learns a continuous waveform-conditioned large-error risk that generalizes the discrete high-error labels observed during training to unseen waveforms. Specifically, we introduce a tail probability head H_{tail} on the spectrally enhanced Tail representation $\widetilde{\mathbf{h}}_{\text{tail}}$:

$$\pi_{\text{tail}} = \sigma\Big(H_{\text{tail}}(\widetilde{\mathbf{h}}_{\text{tail}})\Big), \qquad \mathcal{L}_{\text{BCE}} = -\mathbb{E}\left[q^{\text{Seis}} \log \pi_{\text{tail}} + (1 - q^{\text{Seis}}) \log(1 - \pi_{\text{tail}})\right]. \tag{8}$$

Here, $\pi_{\text{tail}} \in (0, 1)$ denotes the predicted risk that the current waveform condition is associated with a large arrival-time error, q^{Seis} is the discrete high-error supervision produced by Seis TIS, and $\mathcal{L}_{\text{BCE}}$ is the corresponding binary cross-entropy loss. This converts the discrete training-time labels into a continuous waveform-dependent risk signal for subsequent Main–Tail adaptive fusion.

We further model the severity of high-error deviations using a generalized Pareto distribution (GPD). Motivated by extreme-value theory, we regularize exceedances above a high prediction-error threshold using a GPD (Balkema & de Haan, 1974; Pickands, 1975). Given the Seis residual r^{Seis} and threshold u^{Seis}, we define

$$e^{\text{Seis}} = r^{\text{Seis}} - u^{\text{Seis}}, \qquad r^{\text{Seis}} > u^{\text{Seis}}. \tag{9}$$

The Tail pathway predicts the GPD shape and scale parameters, ξ and σ, from the enhanced waveform condition and constrains the residual-exceedance structure through

$$\mathcal{L}_{\text{GPD}} = -\mathbb{E}_{r^{\text{Seis}} > u^{\text{Seis}}}\left[\log p_{\text{GPD}}\left(e^{\text{Seis}} \mid \xi, \sigma\right)\right]. \tag{10}$$

Thus, $\mathcal{L}_{\text{BCE}}$ and $\mathcal{L}_{\text{GPD}}$ respectively supervise large-error occurrence and residual-exceedance severity.

2.3 Module III: Dual-Flow Fusion and Probabilistic Prediction

Rather than combining independently formed predictions, SeisTailFlow directly fuses the Main and Tail conditional vector fields during continuous transport, allowing dominant and high-error-aware conditional information to jointly shape the final arrival-time distribution.

Adaptive Vector-Field Fusion. During fused transport, SeisTailFlow evaluates $v_{\text{main}}(z_t, t \mid \mathbf{h}_{\text{main}})$ and $v_{\text{tail}}(z_t, t \mid \widetilde{\mathbf{h}}_{\text{tail}})$ at the same evolving state z_t and adaptively combines them using the waveform-conditioned large-error risk π_{tail} learned from Seis-TIS supervision. Thus, the two pathways interact directly at the vector-field level throughout continuous transport. We define the adaptive routing weight as

$$\rho_t = \text{sigmoid}\left(\frac{u_t - |z_t|}{\kappa u_{\text{base}}}\right)(1 - \pi_{\text{tail}}), \quad u_t = u_{\text{base}}\left[1 + 0.5\tanh\left(f_u(\widetilde{\mathbf{h}}_{\text{tail}}, t)\right)\right], \tag{11}$$

where ρ_t weights the Main Flow and $1 - \rho_t$ weights the Tail Flow. The routing therefore depends jointly on the evolving transport state and the waveform-conditioned large-error risk: the Main Flow receives greater weight near central transport states with low predicted error risk, whereas the Tail Flow contribution increases as $|z_t|$ grows or the predicted large-error risk becomes higher.

The resulting fused conditional vector field is

$$v_{\text{fused}}(z_t, t \mid \mathbf{X}) = \rho_t\, v_{\text{main}}(z_t, t \mid \mathbf{h}_{\text{main}}) + (1 - \rho_t)\, v_{\text{tail}}(z_t, t \mid \widetilde{\mathbf{h}}_{\text{tail}}). \tag{12}$$

Here, $v_{\text{fused}}(z_t, t \mid \mathbf{X})$ denotes the waveform-conditioned fused vector field at state z_t and flow time t.

Probabilistic Phase-Arrival Prediction. Given the fused vector field, SeisTailFlow transports random initial states to the conditional residual space through continuous integration. We adopt a location–residual decomposition that separates the waveform-dependent arrival-time center from its stochastic residual:

$$Y = \mu(\mathbf{X}) + Z, \qquad \mu(\mathbf{X}) = L_\eta\left(\mathbf{h}_{\text{main}}\right), \tag{13}$$

where $L_\eta(\cdot)$ is a learnable location mapping, $\mu(\mathbf{X})$ represents the conditional arrival-time center, and Z denotes the stochastic residual around this center. Details of the location parameterization and supervision are provided in Appendix B.4.

Given an input waveform $\mathbf{X}$, we independently draw M initial states $z_0^{(m)} \sim \mathcal{N}(0, 1)$, $m = 1, \ldots, M$, and transport each sample from $t = 0$ to $t = 1$ through the same fused vector field:

$$\frac{\mathrm{d}z_t^{(m)}}{\mathrm{d}t} = v_{\text{fused}}\left(z_t^{(m)}, t \mid \mathbf{X}\right), \qquad z_1^{(m)} = \mathcal{T}_{v_{\text{fused}}}\left(z_0^{(m)} \mid \mathbf{X}\right), \quad m = 1, \ldots, M, \tag{14}$$

where $\mathcal{T}_{v_{\text{fused}}}(\cdot \mid \mathbf{X})$ denotes the conditional transport map induced by continuous integration. All terminal residual states are generated through the same adaptive Main–Tail transport rather than by post-hoc mixing of independently generated predictions. The resulting phase-arrival samples are

$$\hat{y}^{(m)} = \mu(\mathbf{X}) + z_1^{(m)}, \qquad m = 1, \ldots, M, \tag{15}$$

where $\{\hat{y}^{(m)}\}_{m=1}^M$ provides a Monte Carlo approximation to the conditional predictive distribution $p(Y \mid \mathbf{X})$. We use the predictive mean $\hat{y} = M^{-1}\sum_{m=1}^M \hat{y}^{(m)}$ as the point estimate, while the empirical sample distribution provides predictive intervals and probabilistic uncertainty.

2.4 Joint Optimization

To provide robust supervision for the waveform-dependent location term $\mu(\mathbf{X})$, we impose a Student-t negative log-likelihood:

$$\mathcal{L}_{\text{loc}} = -\frac{1}{N}\sum_{i=1}^N \log p_{\text{Student-}t}\left(y_i \mid \mu(\mathbf{X}_i), s_{\text{res},i}, \nu\right), \tag{16}$$

where $s_{\text{res},i} > 0$ is a sample-dependent residual scale and ν denotes the degrees of freedom.

To further align the generated arrival-time samples with the empirical target distribution, we adopt an empirical characteristic function (ECF) matching objective (Ansari et al., 2020):

$$\mathcal{L}_{\text{ECF}} = \frac{1}{|\Omega|}\sum_{\omega\in\Omega}\left|\frac{1}{M}\sum_{m=1}^M \exp\left(\mathrm{i}\omega\hat{y}^{(m)}\right) - \frac{1}{N}\sum_{i=1}^N \exp(\mathrm{i}\omega y_i)\right|^2. \tag{17}$$

The final objective is

$$\mathcal{L} = \mathcal{L}_{\text{FM}} + \lambda_{\text{loc}}\mathcal{L}_{\text{loc}} + \lambda_{\text{tail}}\mathcal{L}_{\text{BCE}} + \lambda_{\text{GPD}}\mathcal{L}_{\text{GPD}} + \lambda_{\text{ECF}}\mathcal{L}_{\text{ECF}}, \tag{18}$$

where $\mathcal{L}_{\text{FM}} = \mathcal{L}_{\text{FM}}^{\text{main}} + \mathcal{L}_{\text{FM}}^{\text{tail}}$, and the λ coefficients balance the corresponding training objectives. Additional optimization and implementation details are provided in Appendix C.

3 EXPERIMENTS

3.1 SYNTHETIC EXPERIMENTS

Synthetic Data Generation. To study the formation and modeling of large phase-arrival errors under controlled conditions, we construct a three-component synthetic waveform dataset following prior approaches that emulate realistic phase morphology and observational noise characteristics (Zhou et al., 2024; Song et al., 2026). Each sample consists of vertical (Z), north–south (N), and east–west (E) components, which represent orthogonal projections of the same ground motion rather than different wave types; the P-wave arrival is therefore jointly expressed across the three components with component-dependent amplitudes. The true P-wave arrival and key waveform factors governing picking difficulty are explicitly controllable.

$$X_c(t) = S_c(t; \tau_P, \theta_s) + N_c(t; \theta_n) + C_c(t; \theta_c), \tag{19}$$

where S_c denotes the P-wave signal with randomized polarization and spectral characteristics, N_c denotes colored background noise with controllable SNR, and C_c denotes a structured transient preceding the true P-wave arrival. We vary the SNR, confounder strength, and confounder–P-wave spectral similarity to generate waveforms ranging from clear arrivals to low-SNR and strongly ambiguous cases (Tang & Chen, 2026), while preserving waveform-level randomness and conditional heterogeneity. The dataset contains 20,000 training, 4,000 validation, and 4,000 test samples. As shown in Figure 2, we present three representative three-component waveforms with different SNRs and ambiguity levels, where ambiguity reflects the similarity between the preceding confounder and the true P wave. All waveforms are aligned to the true P-wave arrival at $T = 0$. As SNR decreases and confounder similarity increases, phase information becomes harder to identify. More generation details are provided in Appendix D.1.

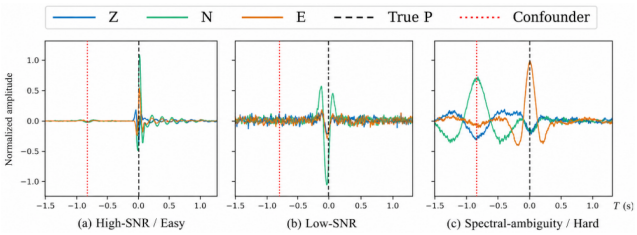


Figure 2: **Representative three-component synthetic waveforms.** Black dashed and red dotted lines denote the true P arrival and the preceding confounder.

Baselines. We compare SeisTailFlow with representative methods spanning deterministic phase picking, uncertainty quantification, and conditional generative modeling. For deterministic picking, we include PhaseNet (Zhu & Beroza, 2019) and the attention-based EQTransformer (Mousavi et al., 2020). For probabilistic prediction, we adopt the Bayesian uncertainty method of Armstrong et al. (2023) to evaluate predictive calibration and sharpness. For conditional generative modeling, we construct a Vanilla CFM baseline based on Conditional Flow Matching (Lipman et al., 2023) and include CW-Flow (Yang et al., 2026) under the conditionally whitened generative modeling framework.

Table 1: **Quantitative results on the Synthetic experiments.** Detailed definitions of the evaluation metrics are provided in Appendix D.2.

	Overall	Tail Error			Low-SNR	Extreme Error		Sharp.	Calibration	
Method	MAE↓	P95↓	P99↓	Tail MAE↓	Low MAE↓	ES@99↓	Exc.@0.3↓	W@95↓	CRPS↓	CE@95↓
PhaseNet	0.0082±.001	0.0185±.003	0.0234±.007	0.0321±.005	0.0085±.003	0.0392±.018	0.18±.03%	–	–	–
EQTransformer	0.0093±.002	0.0224±.005	0.0307±.010	0.0348±.008	0.0097±.004	0.0417±.016	0.24±.03%	–	–	–
Armstrong	0.0101±.001	0.0248±.004	0.0336±.009	0.0367±.007	0.0104±.004	0.0449±.015	0.27±.02%	0.0348±.009	0.0076±.002	0.0610±.003
Vanilla CFM	0.0621±.001	0.1134±.004	0.1430±.008	0.1286±.006	0.0590±.004	0.1755±.014	0.82±.02%	0.5193±.008	0.0467±.001	0.0495±.003
CW-Flow	0.0081±.001	0.0182±.006	0.0228±.011	0.0314±.009	0.0082±.005	0.0384±.013	0.16±.02%	0.0317±.007	0.0063±.001	0.0528±.002
SeisTailFlow	**0.0069**±.001	**0.0160**±.003	**0.0197**±.006	**0.0268**±.004	**0.0071**±.003	**0.0228**±.010	**0.09**±.01%	**0.0293**±.006	**0.0049**±.001	**0.0453**±.001

Main Results. Table 1 shows that SeisTailFlow consistently improves both overall accuracy and tail-error performance, with more pronounced gains under extreme-error and low-SNR conditions. Importantly, these improvements are accompanied by better probabilistic quality, indicating that tail-error reduction is achieved without sacrificing overall accuracy or predictive sharpness.Additional results are provided in Appendices D.3–D.6, while the complete ablation study is presented in Appendix E.

3.2 Empirical Experiments

STEAD Dataset. We conduct real-data experiments on the Stanford Earthquake Dataset (STEAD) (Mousavi et al., 2019), focusing on P-wave arrival-time prediction. Each record contains Z-, N-, and E-component waveforms sampled at 100 Hz with an input length of $T = 6000$ (60 s).We split the data at the earthquake-event level to prevent correlated traces from the same event from appearing across training and evaluation sets.Specifically, STEAD chunk2 is used for training, calibration, and development, with the training split containing 101,997 traces from 43,549 events, while the independent chunk3 serves as the frozen final holdout with 147,499 traces from 97,914 events. Full details of the data splits, preprocessing, and evaluation protocol are provided in Appendix F.1.

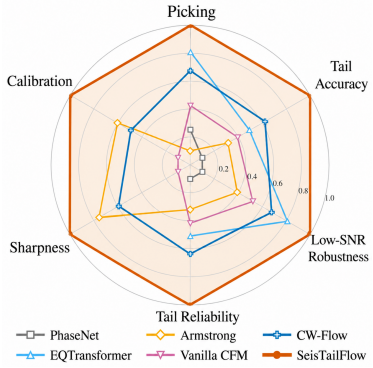


Figure 3: **Six-metric comparison on STEAD.**

Main Results. Table 2 shows that SeisTailFlow improves overall picking accuracy, with larger gains in high-error and low-SNR regimes. Improvements in tail-error, sharpness, and calibration metrics further indicate that these gains are achieved without simply widening predictive uncertainty. Figure 3 summarizes the results across six dimensions, where each dimension is represented by a composite score obtained by normalizing and aggregating the corresponding metrics in The consistently strong six-dimensional profile shows that the improvement is not confined to a single metric, but extends from overall picking to tail-error robustness and probabilistic quality. Additional STEAD results on tail error distribution and extreme-error metrics, calibration and coverage-matched sharpness, conditional performance across SNR levels, and robustness analyses are provided in Appendices F.2–F.5.

Table 2: **Quantitative results on the STEAD dataset.**

	Overall	Tail Error			Low-SNR	Extreme Error		Sharp.	Calibration	
Method	MAE↓	P95↓	P99↓	Tail MAE↓	Low MAE↓	ES@99↓	Exc.@0.3↓	W@95↓	CRPS↓	CE@95↓
PhaseNet	0.438±.011	1.214±.032	1.906±.057	1.521±.045	0.642±.019	2.184±.069	42.6±.9%	–	–	–
EQTransformer	0.409±.009	1.036±.025	1.611±.046	1.263±.036	0.501±.014	1.873±.055	35.8±.8%	–	–	–
Armstrong	0.446±.010	1.127±.028	1.741±.050	1.371±.039	0.584±.017	2.008±.060	39.1±.8%	1.103±.021	0.276±.006	0.0018±.0004
Vanilla CFM	0.429±.010	1.091±.026	1.682±.047	1.318±.037	0.558±.016	1.938±.057	37.4±.8%	1.188±.023	0.289±.006	0.0035±.0006
CW-Flow	0.416±.009	0.982±.023	1.518±.042	1.176±.033	0.527±.015	1.746±.051	34.2±.7%	1.124±.022	0.268±.005	0.0029±.0005
SeisTailFlow	**0.399**±.008	**0.821**±.019	**1.216**±.034	**0.931**±.027	**0.463**±.013	**1.398**±.041	**29.4**±.6%	**1.071**±.020	**0.255**±.005	**0.0012**±.0003

Ablation Study. Table 3 evaluates SeisTailFlow from four aspects: tail supervision, spectral conditioning, high-error modeling, and Main–Tail coordination. The full model consistently performs best across all metrics. Replacing out-of-sample Seis TIS supervision or removing the Tail Flow causes the most evident degradation in high-quantile and tail errors, confirming the importance of reliable high-error identification and dedicated high-error-aware modeling. Spectral conditioning and the GPD constraint provide complementary gains, while neither the Main-Flow-only nor Tail-Flow-only variant matches the full model, highlighting the benefit of coordinating the two flows.

Table 3: **Ablation study of SeisTailFlow on STEAD across four key components.**

	Tail Supervision		Spectral Conditioning		High-Error Modeling		Main–Tail Coordination		Full
Metric	In-sample residual	w/o Seis TIS supervision	Time-domain only	w/o spectral filtering	w/o Tail Flow	w/o GPD constraint	Main Flow only	Tail Flow only	**SeisTailFlow**
MAE↓	0.425±.009	0.432±.010	0.429±.009	0.421±.009	0.424±.009	0.407±.008	0.404±.008	0.497±.012	**0.399**±.008
P95↓	0.934±.024	0.958±.026	0.941±.025	0.911±.023	0.949±.026	0.856±.021	0.879±.022	1.081±.029	**0.821**±.019
P99↓	1.402±.041	1.451±.045	1.423±.043	1.371±.040	1.469±.047	1.289±.036	1.336±.039	1.668±.052	**1.216**±.034
Tail MAE↓	1.084±.031	1.139±.034	1.112±.033	1.071±.030	1.158±.035	0.997±.028	1.046±.030	1.301±.039	**0.931**±.027
CRPS↓	0.272±.006	0.279±.006	0.276±.006	0.271±.005	0.281±.006	0.264±.005	0.268±.005	0.324±.007	**0.255**±.005

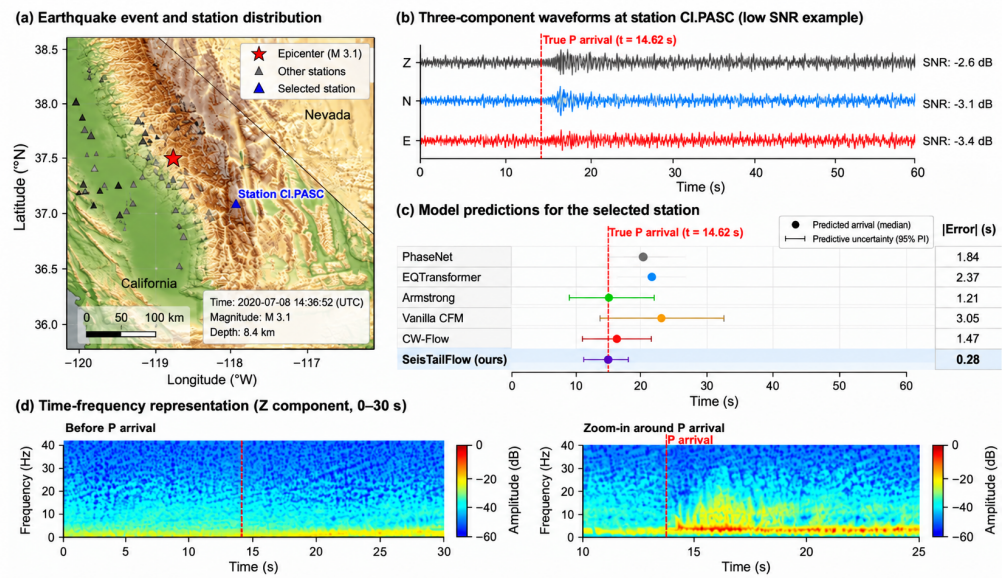


Figure 4: **Representative low-SNR case from STEAD.** (a) Event and station locations; (b) three-component waveform; (c) arrival-time predictions and errors; (d) Z-component time–frequency representations.

Case Study under Low-SNR Conditions. Figure 4 presents a representative low-SNR case from STEAD and illustrates the practical value of tail-aware phase picking under challenging waveform conditions. Figure 4(a) shows the earthquake event and surrounding stations in the California–Nevada region, with station CI.PASC selected for detailed analysis. As shown in Figure 4(b), all three components exhibit very low SNRs, and the true P-wave arrival at 14.62,s is only weakly distinguishable from the surrounding waveform fluctuations, making accurate picking from the time-domain signal particularly difficult. Figure 4(c) compares the predicted arrival times of all methods and the 95% predictive intervals of probabilistic methods.Several baselines exhibit noticeable deviations from the manual pick, whereas SeisTailFlow achieves the smallest absolute error (0.28,s) and produces a relatively concentrated predictive interval around the true arrival. Figure 4(d) provides further insight from the Z-component time–frequency representation. Although the P-wave onset is difficult to isolate in the raw waveform, phase-related energy changes become more distinguishable relative to the background spectrum around the arrival time.

This case highlights the potential practical value for regional earthquake monitoring in the California–Nevada area. The event has a relatively small magnitude (M 3.1) and is recorded by multiple nearby stations, yet the selected CI.PASC waveform exhibits strongly negative SNRs across all three components. In such settings, large errors at a few noisy stations can propagate to phase association and event localization. By producing a P-wave estimate much closer to the manual pick in this difficult case, SeisTailFlow can provide more reliable arrival-time inputs for multi-station association and localization. This example clearly illustrates how reducing large-tailed errors at noisy stations can improve the robustness of automated monitoring for small and difficult-to-detect events. Additional waveform examples under different noise and difficulty levels are provided in Appendix F.6.

4 Conclusion

We proposed SeisTailFlow for high-error-aware conditional modeling in seismic phase picking. By addressing scarce high-error observations, condition-dependent ambiguity, and Main–Tail coordination, SeisTailFlow exploits waveform-dependent structure associated with elevated prediction-error risk, improving overall performance, especially in high-error and low-SNR regimes. These results support more reliable phase picking through high-error-aware probabilistic learning and suggest broader applicability to scientific problems with rare but consequential heavy-tailed errors.

AI USE STATEMENT

Generative AI tools were used for limited language polishing and presentation assistance, including grammar and wording suggestions for parts of the manuscript. The authors reviewed the assisted edits and take responsibility for the final content. No generative AI tools were used to generate research ideas, methods, algorithms, proofs, datasets, experiments, quantitative results, or conclusions.

REPRODUCIBILITY STATEMENT

We provide detailed descriptions of the model architecture, training procedure, and experimental setup in the main text and appendices. The appendices further document the synthetic data generation process, STEAD data processing and splitting protocol, OOF tail construction, evaluation metrics, and additional experimental settings. We also provide anonymized source code and configuration files as supplementary material to facilitate reproduction of the reported results.

REFERENCES

Abdul Fatir Ansari, Jonathan Scarlett, and Harold Soh. A characteristic function approach to deep implicit generative modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7478–7487, 2020.

Alysha D. Armstrong, Zachary Claerhout, Ben Baker, and Keith D. Koper. A deep-learning phase picker with calibrated bayesian-derived uncertainties for earthquakes in the yellowstone volcanic region. *Bulletin of the Seismological Society of America*, 113(6):2323–2344, 2023.

August A. Balkema and Laurens de Haan. Residual life time at great age. *The Annals of Probability*, 2(5):792–804, 1974.

Thomas Bornstein, Dietrich Lange, Jannes Münchmeyer, Jack Woollam, Andreas Rietbrock, Grace Barcheck, Ingo Grevemeyer, and Frederik Tilmann. Pickblue: Seismic phase picking for ocean bottom seismometers with deep learning. *Earth and Space Science*, 11(1):e2023EA003332, 2024.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David K. Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, volume 31, pp. 6571–6583, 2018.

Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins. Double/debiased machine learning for treatment and structural parameters. *The Econometrics Journal*, 21(1):C1–C68, 2018.

Domenico Di Giacomo, Tom Garth, István Bondár, Natalia Poiata, Ryan Gallacher, James Harris, and Dmitry A. Storchak. On the location uncertainty of early-instrumental earthquakes. *Seismica*, 4(1), 2025.

R. Di Stefano, F. Aldersons, E. Kissling, P. Baccheschi, C. Chiarabba, and D. Giardini. Automatic seismic phase picking and consistent observation error assessment: Application to the italian seismicity. *Geophysical Journal International*, 165(1):121–134, 2006.

Tian Feng, Saeed Mohanna, and Lingsen Meng. Edgephase: A deep learning model for multi-station seismic phase picking. *Geochemistry, Geophysics, Geosystems*, 23(11):e2022GC010453, 2022.

Will Grathwohl, Ricky T. Q. Chen, Jesse Bettencourt, Ilya Sutskever, and David Duvenaud. FFJORD: Free-form continuous dynamics for scalable reversible generative models. In *International Conference on Learning Representations*, 2019.

Stephan Husen and Jeanne Hardebeck. Earthquake location accuracy. Community Online Resource for Statistical Seismicity Analysis (CORSSA), 2010.

Marcel Kollovieh, Marten Lienen, David Lüdke, Leo Schwinn, and Stephan Günnemann. Flow matching with gaussian process priors for probabilistic time series forecasting. In *International Conference on Learning Representations*, 2025.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

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.

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*, 11:3952, 2020.

Jannes Münchmeyer, Jack Woollam, Andreas Rietbrock, Frederik Tilmann, Dietrich Lange, Thomas Bornstein, Tobias Diehl, Carlo Giunchi, Florian Haslinger, Dario Jozinović, Alberto Michelini, Joachim Saul, and Hugo Soto. Which picker fits my data? a quantitative evaluation of deep learning based seismic pickers. *Journal of Geophysical Research: Solid Earth*, 127(1): e2021JB023499, 2022.

Tarun Naskar, Mrinal Bhaumik, Sayan Mukherjee, and Sai Vivek Adari. Spectral whitening based seismic data preprocessing technique to improve the quality of surface wave's velocity spectra. *Computers & Geosciences*, 195:105784, 2025.

Alireza Niksejel and Miao Zhang. Obstransformer: A deep-learning seismic phase picker for obs data using automated labelling and transfer learning. *Geophysical Journal International*, 237(1): 485–505, 2024.

Jonggeon Park, Giung Nam, Hyunsu Kim, Jongmin Yoon, and Juho Lee. Ensemble distribution distillation via flow matching. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 48170–48191, 2025a.

Yongsoo Park, Alysha D. Armstrong, William L. Yeck, David R. Shelly, and Gregory C. Beroza. Divide and conquer: Separating the two probabilities in seismic phase picking. *Geophysical Journal International*, 243(2):ggaf333, 2025b.

James Pickands, III. Statistical inference using extreme order statistics. *The Annals of Statistics*, 3 (1):119–131, 1975.

Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1530–1538, 2015.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pp. 234–241, 2015.

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.

Xu Si, Xinming Wu, Zefeng Li, Shenghou Wang, and Jun Zhu. An all-in-one seismic phase picking, location, and association network for multi-task multi-station earthquake monitoring. *Communications Earth & Environment*, 5(1):22, 2024.

Xiaohan Song, Men-Andrin Meier, William L. Ellsworth, and Gregory C. Beroza. FocoNet: Transformer-based focal-mechanism determination. *Journal of Geophysical Research: Machine Learning and Computation*, 3(2):e2025JH000879, 2026.

Hugo Soto and Bernd Schurr. Deepphasepick: A method for detecting and picking seismic phases from local earthquakes based on highly optimized convolutional and recurrent deep neural networks. *Geophysical Journal International*, 227(2):1268–1294, 2021.

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.

Kuan-Wei Tang and Kuan-Yu Chen. SeismoDual: A dual-domain deep learning framework for robust seismic phase picking. *Computers & Geosciences*, 207:106080, 2026.

Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856.

David H. Wolpert. Stacked generalization. *Neural Networks*, 5(2):241–259, 1992.

Yixing Wu, Shiou-Ya Wang, Dingyi Nie, Sanket Kumbhar, Yun-Tung Hsieh, Yun-Cheng Wang, Po-Chyi Su, and C.-C. Jay Kuo. GreenPhase: A green learning approach for earthquake phase picking. *arXiv preprint arXiv:2603.03344*, 2026.

Yanfeng Yang, Siwei Chen, Pingping Hu, Zhaotong Shen, Yingjie Zhang, Zhuoran Sun, Shuai Li, Ziqi Chen, and Kenji Fukumizu. Conditionally whitened generative models for probabilistic time series forecasting. In *International Conference on Learning Representations*, 2026.

Yi-Lun Zhang, Zhi-Chao Yu, and Chuan He. Uncertainty-aware neural networks with manual quality control for hydraulic fracturing downhole microseismic monitoring: From automated phase detection to robust source location. *Petroleum Science*, 22(11):4520–4537, 2025.

Jiarun Zhou, Thanh-Son Phm, and Hrvoje Tkalčić. Deep-learning phase-onset picker for deep earth seismology: PKIKP waves. *Journal of Geophysical Research: Solid Earth*, 129(9): e2024JB029360, 2024.

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.

Weiqiang Zhu, Kai Sheng Tai, S. Mostafa Mousavi, Peter Bailis, and Gregory C. Beroza. 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, 2022.

A Related Works

A.1 Deep Learning for Seismic Phase Picking

Deep learning-based seismic phase picking typically aims to identify P- and S-wave arrivals directly from continuous waveform recordings and estimate their corresponding arrival times. PhaseNet formulates this task as a point-wise classification problem and employs a convolutional neural network to produce probability sequences for P waves, S waves, and noise from three-component seismic waveforms, achieving substantially improved automatic picking performance over traditional approaches (Zhu & Beroza, 2019). EQTransformer subsequently combines convolutional and recurrent architectures with attention mechanisms to jointly perform earthquake detection and P/S phase picking within a unified model, thereby improving phase identification under complex continuous-waveform and noisy-background conditions (Mousavi et al., 2020). Both methods learn phase-relevant representations directly from raw waveform data and produce detection and arrival-time information in an end-to-end manner.

Subsequent studies have further improved seismic phase identification by exploiting dependencies across tasks and stations. EQNet integrates feature extraction, phase picking, and event detection and association within a unified framework, allowing individual phase observations to contribute directly to the complete earthquake detection pipeline (Zhu et al., 2022). For multi-station settings, EdgePhase leverages spatial dependencies among neighboring stations through cross-station information interaction (Feng et al., 2022), while PhaseNO jointly models multi-station waveforms using neural operators to capture broader phase-related correlations (Sun et al., 2023). Meanwhile, recent methods such as GreenPhase have begun to consider model efficiency and interpretability in addition to high picking accuracy, using multi-resolution feed-forward feature learning to reduce computational cost (Wu et al., 2026). Together, these studies extend the capabilities of deep seismic phase picking from complementary perspectives, including task coordination, spatial information utilization, and computational efficiency.

Despite these advances, the performance of existing phase pickers can still vary substantially with data distributions and waveform conditions. Cross-dataset benchmark studies have shown that both generalization performance and large arrival-time errors are strongly data dependent (Münchmeyer et al., 2022), with degradation becoming particularly evident for low signal-to-noise ratio waveforms or records with ambiguous phase onsets. Existing methods typically treat such errors as part of overall picking performance, but rarely use high-error cases from the upper tail of the prediction-error distribution as explicit supervision for waveform-conditioned modeling.

A.2 Probabilistic Phase Picking and Uncertainty Quantification

Early studies on automatic phase picking already combined arrival-time estimation with observational error assessment, using waveform characteristics and signal-to-noise conditions to quantify picking quality (Di Stefano et al., 2006). With the development of deep learning, uncertainty estimation has been further incorporated into seismic phase detection and arrival-time prediction. EQTransformer retains dropout during inference and performs multiple stochastic forward passes to approximate Bayesian inference, using the variability of repeated predictions to characterize uncertainty in event detection and phase picking (Mousavi et al., 2020). DeepPhasePick similarly applies Monte Carlo dropout to P/S arrival-time estimation, using stochastic predictive samples to quantify uncertainty in phase-onset timing (Soto & Schurr, 2021). Subsequent work has placed greater emphasis on probabilistic calibration of predictive uncertainty. Armstrong et al. construct Bayesian-approximate arrival-time predictive distributions using SWAG/MultiSWAG and calibrate the resulting uncertainty estimates to obtain phase-arrival credible intervals with target coverage levels (Armstrong et al., 2023).

Beyond Bayesian predictive uncertainty, recent studies have further examined the statistical interpretation of the probabilities produced by seismic phase pickers. Conventional deep-learning pickers typically construct probabilistic target sequences around manually annotated arrival times. Although these outputs can be used to localize phase arrivals, their amplitudes are not inherently equivalent to calibrated detection probabilities or arrival-time probabilities. Park et al. explicitly distinguish the probability that a phase is present from the probability distribution of its arrival time conditional on phase presence. They construct a discrete probability mass function over arrival times

from classification probabilities and further evaluate its probabilistic calibration properties (Park et al., 2025b). Uncertainty-aware phase picking has also been introduced into practical applications such as low-SNR microseismic monitoring, where arrival-time predictions and their associated uncertainty estimates are jointly used for downstream quality control and source localization (Zhang et al., 2025).

Overall, existing studies have progressed from quantifying predictive uncertainty toward calibrating credible intervals and constructing arrival-time probabilities with clearer statistical interpretations. However, these approaches primarily focus on model uncertainty, arrival-time uncertainty, and probabilistic calibration. High-error cases in the upper tail of the prediction-error distribution, whose occurrence and severity vary with waveform conditions, have not yet been a primary modeling target of these methods.

A.3 Conditional Flow Matching for Probabilistic Prediction

To reduce the restrictions imposed by predefined parametric distributions on predictive uncertainty modeling, generative probabilistic models provide a direct means of learning complex distributions. Among them, Normalizing Flows transform a simple base distribution into a more complex target distribution through a sequence of invertible mappings, providing flexible distributional expressiveness while retaining tractable probability-density evaluation (Rezende & Mohamed, 2015). When conditioned on input information, this framework can represent input-dependent distributional characteristics such as heteroscedasticity, skewness, and multimodality. Continuous Normalizing Flows (CNFs) further generalize finite sequences of invertible transformations to continuous-time transport processes governed by ordinary differential equations, allowing the transformation from a base distribution to a target distribution to be parameterized by a neural vector field (Chen et al., 2018; Grathwohl et al., 2019).

Flow Matching (FM) provides a simulation-free training strategy for continuous flows by directly regressing the vector field associated with a prescribed probability path (Lipman et al., 2023). The framework accommodates different forms of probability paths, including paths related to diffusion processes and displacement interpolation based on optimal transport. Tong et al. further formulate Conditional Flow Matching (CFM) as a more general simulation-free framework for training CNFs and introduce Optimal Transport Conditional Flow Matching (OT-CFM), which constructs probability paths using optimal-transport couplings between source and target distributions (Tong et al., 2024). This formulation allows conditioning information to directly participate in the continuous transport process, providing a flexible generative mechanism for learning input-dependent predictive distributions.

Such conditional generative capabilities have also begun to be explored for probabilistic time-series forecasting. TSFlow combines CFM with Gaussian-process priors and optimal-transport paths, using temporal structure to construct data-dependent generative priors (Kollovieh et al., 2025). Conditioned Whitened Generative Models instead construct conditional priors from conditional mean and covariance information, and further develop CW-Flow for probabilistic forecasting (Yang et al., 2026). These studies demonstrate that flow-matching-based generative models can incorporate observational information directly into conditional distribution learning, providing a unified framework for representing predictive uncertainty that varies with the input state. For seismic phase picking, this capability is particularly suitable for modeling waveform-dependent heterogeneity and asymmetry in arrival-time uncertainty.

A.4 Problem Formulation and Further Insights into SeisTailFlow

Given a multi-component seismic waveform $\mathbf{X} \in \mathbb{R}^{C\times T}$, where C denotes the number of waveform components and T the number of temporal samples, let the corresponding ground-truth phase-arrival time be $Y \in \mathbb{R}$. A conventional deterministic phase picker can be abstractly expressed as

$$Y = f_1(\mathbf{X}) + \varepsilon_1, \tag{20}$$

where $f_1(\mathbf{X})$ represents the dominant predictive structure learned from the input waveform, and ε_1 denotes the residual error not explained by this prediction rule. Importantly, ε_1 may still contain latent waveform-dependent structure that has not been sufficiently captured by f_1. We therefore

express it conceptually as

$$\varepsilon_1 = f_2(\mathbf{X}) + \varepsilon_2, \qquad Y = f_1(\mathbf{X}) + f_2(\mathbf{X}) + \varepsilon_2, \tag{21}$$

where $f_2(\mathbf{X})$ represents learnable waveform-dependent structure remaining in the residual of the dominant predictor, and ε_2 denotes the remaining random perturbation after this additional structure is accounted for.

To formalize the prediction-error tail considered in this work, for the i-th sample $(\mathbf{x}_i, y_i)$, let the dominant prediction be $\hat{y}_i = f_1(\mathbf{x}_i)$. We define the corresponding signed prediction residual as $\varepsilon_i = y_i - \hat{y}_i$, and the absolute prediction error as $r_i = |\varepsilon_i| = |y_i - \hat{y}_i|$. Given a high threshold u, observations satisfying $r_i > u$ are defined as belonging to the upper tail of the prediction-error distribution.

It is important to emphasize that $f_1(\mathbf{X})$ and $f_2(\mathbf{X})$ in Equation equation 21 constitute a conceptual decomposition of the dominant predictive structure and the waveform-dependent learnable structure remaining in its residuals. They do not imply that SeisTailFlow independently fits these two terms deterministically using the Main Flow and Tail Flow, respectively. Instead, we denote their combined learnable conditional structure as $g(\mathbf{X}) = f_1(\mathbf{X}) + f_2(\mathbf{X})$ and model the resulting predictive uncertainty probabilistically through coordinated Main and Tail pathways with complementary modeling roles. The resulting framework therefore aims to learn the full conditional predictive distribution $p(Y \mid \mathbf{X})$ rather than two separate deterministic predictors.

B Supplementary Details of SeisTailFlow

B.1 Supplementary Details of Module I

Waveform Encoder. Given the globally normalized multi-component waveform $\widetilde{\mathbf{X}} \in \mathbb{R}^{C\times T}$, the shared waveform encoder E_ψ adopts a one-dimensional U-Net-style encoder–decoder architecture to extract multi-scale waveform features while preserving temporal alignment with the input.

The encoder first maps the input waveform into a shallow feature space of dimension D_1 through a convolutional block. It then applies two successive downsampling stages to progressively enlarge the temporal receptive field, while expanding the feature dimensions to D_2 and D_3, respectively. In our implementation, $(D_1, D_2, D_3) = (32, 64, 128)$. Each convolutional block consists of a one-dimensional convolution, batch normalization, and a nonlinear activation function, enabling the network to extract local waveform patterns at the corresponding temporal scale. The downsampling operations further aggregate information from neighboring temporal positions, allowing deeper features to encode contextual structure over a broader temporal range. This multi-scale design enables the encoder to preserve local waveform variations associated with phase onsets while simultaneously incorporating background and contextual information over longer time spans.

The decoder employs two symmetric upsampling stages to progressively recover the temporal resolution. At each stage, the upsampled deep features are fused with encoder features at the corresponding temporal scale through skip connections, thereby combining high-level contextual information with local waveform details at higher temporal resolution. After the two decoding stages, the feature sequence is restored to the original temporal length T and mapped to the final temporal feature representation

$$\mathbf{F} = E_\psi(\widetilde{\mathbf{X}}) = [\mathbf{f}_1, \ldots, \mathbf{f}_T] \in \mathbb{R}^{D\times T},$$

where $D = 32$ in our implementation and $\mathbf{f}_t \in \mathbb{R}^D$ denotes the local waveform feature associated with the t-th temporal position. Accordingly, $\mathbf{F}$ remains temporally aligned with the input waveform and is subsequently used as the input to the temporal attention readout.

Temporal Attention Readout. To convert the temporally aligned feature sequence $\mathbf{F}$ into a fixed-dimensional shared waveform representation, we further apply an attention-based readout over all temporal positions. For each local feature $\mathbf{f}_t$, an attention scoring function $q_\phi : \mathbb{R}^D \to \mathbb{R}$ first produces a scalar score, which is then normalized across the temporal dimension using a softmax:

$$\alpha_t = \frac{\exp(q_\phi(\mathbf{f}_t))}{\sum_{\tau=1}^{T} \exp(q_\phi(\mathbf{f}_\tau))}, \qquad \sum_{t=1}^{T} \alpha_t = 1.$$

Using these weights, the temporal features are aggregated as $\bar{\mathbf{f}} = \sum_{t=1}^{T} \alpha_t \mathbf{f}_t \in \mathbb{R}^D$. Unlike simple global average pooling, this operation allows different temporal positions to contribute unequally to the final representation according to their waveform characteristics.

In addition to the aggregated waveform content, we retain coarse positional information from the attention response along the temporal axis. Let $p_t \in [-1, 1]$ denote the normalized temporal coordinate associated with the t-th sample. In our implementation, it is defined as

$$p_t = -1 + \frac{2(t-1)}{T-1}, \qquad t = 1, \ldots, T,$$

such that the beginning and end of the waveform correspond to -1 and 1, respectively. We then compute the first-order positional statistic of the attention distribution as

$$\ell = \sum_{t=1}^{T} \alpha_t p_t$$

and its weighted temporal dispersion around this center as

$$d = \sqrt{\sum_{t=1}^{T} \alpha_t (p_t - \ell)^2}.$$

Here, ℓ describes the temporal location around which the attention response is primarily concentrated, while d characterizes how concentrated or dispersed the response is along the temporal axis. These two statistics therefore complement $\bar{\mathbf{f}}$ with coarse information about the temporal location and spread of the attended waveform features.

Finally, we concatenate the attention-aggregated feature with the temporal location statistics to obtain

$$\mathbf{h}_{\text{wave}} = [\bar{\mathbf{f}}; \ell; d] \in \mathbb{R}^{D+2}.$$

With $D = 32$ in our implementation, this yields $\mathbf{h}_{\text{wave}} \in \mathbb{R}^{34}$, which serves as the shared conditioning representation for the subsequent Main and Tail pathways.

B.2 Implementation Details of Main–Tail Conditional Flows

This section provides additional implementation details for the Main–Tail dual conditional pathways and their associated conditional flows introduced in the main text. Specifically, we first describe the network architectures of the Main Projection and Tail Projection, then specify the probability paths used to train the two conditional flows, and finally present the parameterization of the Main and Tail conditional vector fields.

Dual Conditional Projections. The Main Projection and Tail Projection introduced in the main text share the same network architecture but use separate learnable parameters. Given the shared waveform representation $\mathbf{h}_{\text{wave}} \in \mathbb{R}^{D+2}$, both projection networks consist of Layer Normalization, two linear transformations, and an intermediate GELU activation:

$$\mathbf{h}_* = W_{*,2}^{\text{proj}} \,\text{GELU}\left(W_{*,1}^{\text{proj}} \,\text{LN}(\mathbf{h}_{\text{wave}})\right), \qquad * \in \{\text{main}, \text{tail}\}. \tag{22}$$

Here, $\text{LN}(\cdot)$ denotes Layer Normalization, while $W_{*,1}^{\text{proj}}$ and $W_{*,2}^{\text{proj}}$ denote the first- and second-layer projection parameters for pathway $*$, respectively. The Main and Tail projection parameters are separate, allowing the two pathways to derive distinct conditioning representations from the same shared waveform representation.

In our implementation, the first linear transformation maps the $(D+2)$-dimensional shared waveform representation to a 64-dimensional conditional space, while the second linear layer preserves this dimensionality. This yields $\mathbf{h}_{\text{main}}, \mathbf{h}_{\text{tail}} \in \mathbb{R}^{64}$. The representation $\mathbf{h}_{\text{main}}$ is used directly as the conditioning representation for the Main Flow. In contrast, $\mathbf{h}_{\text{tail}}$ is further processed by the spectral-conditioning enhancement described in the main text, producing $\widetilde{\mathbf{h}}_{\text{tail}} \in \mathbb{R}^{64}$, which serves as the final conditioning representation for the Tail Flow.

Probability-Path Construction. The main text provides only a high-level description of the probability paths used by the Main and Tail Flows. Here, we specify their construction in detail. For the Main Flow, the initial state is sampled from a standard Gaussian base distribution, $z_0^{\text{main}} \sim \mathcal{N}(0,1)$. For the Tail Flow, the initial state is sampled from a variance-standardized Student-t base distribution, $z_0^{\text{tail}} \sim t_\nu^{\text{std}}$, where ν denotes the degrees of freedom.

For $t \sim \mathcal{U}(0,1)$, both conditional flows employ a linear probability path of the same form:

$$z_t^* = (1-t)z_0^* + tz_1^*, \qquad u_t^* = \frac{\partial z_t^*}{\partial t} = z_1^* - z_0^*, \qquad * \in \{\text{main}, \text{tail}\}. \tag{23}$$

Here, z_0^*, z_t^*, and z_1^* denote the initial, intermediate, and target states of pathway $*$, respectively, while u_t^* denotes the target velocity of the corresponding probability path at flow time t. Thus, the Main and Tail Flows share the same path-construction form but differ in their base distributions and conditioning representations, yielding separately parameterized conditional vector fields.

Conditional Vector-Field Parameterization. The Main Flow and Tail Flow use separately parameterized networks to represent their respective conditional vector fields. During training, the Main Flow concatenates its probability-path state z_t^{main}, flow time t, and Main conditioning representation $\mathbf{h}_{\text{main}}$, while the Tail Flow analogously uses z_t^{tail}, t, and the spectrally enhanced Tail representation $\widetilde{\mathbf{h}}_{\text{tail}}$:

$$\mathbf{c}_{\text{main},t} = [z_t^{\text{main}}; t; \mathbf{h}_{\text{main}}], \qquad \mathbf{c}_{\text{tail},t} = [z_t^{\text{tail}}; t; \widetilde{\mathbf{h}}_{\text{tail}}], \tag{24}$$

where $[\cdot\,;\cdot]$ denotes feature concatenation, and $\mathbf{c}_{\text{main},t}$ and $\mathbf{c}_{\text{tail},t}$ are the inputs to the Main and Tail conditional vector-field networks, respectively.

The two pathways then pass their inputs through separate multilayer perceptrons to obtain hidden representations:

$$\mathbf{g}_{*,t} = \text{SiLU}\left(W_{*,2}^{\text{vf}} \,\text{SiLU}\left(W_{*,1}^{\text{vf}} \mathbf{c}_{*,t}\right)\right), \qquad * \in \{\text{main}, \text{tail}\}, \tag{25}$$

followed by pathway-specific velocity heads that produce scalar conditional velocities:

$$v_* = W_{*,v}^{\text{vf}} \mathbf{g}_{*,t}, \qquad * \in \{\text{main}, \text{tail}\}. \tag{26}$$

Here, $W_{*,1}^{\text{vf}}$, $W_{*,2}^{\text{vf}}$, and $W_{*,v}^{\text{vf}}$ denote the vector-field network parameters for pathway $*$. The Main and Tail vector fields use separate parameter sets.

In our implementation, $\mathbf{h}_{\text{main}}$ and $\widetilde{\mathbf{h}}_{\text{tail}}$ are both 64-dimensional. After concatenating the scalar transport state and flow time, $\mathbf{c}_{\text{main},t}$ and $\mathbf{c}_{\text{tail},t}$ are therefore both 66-dimensional. The hidden dimension of both vector-field networks is set to 128. Consequently, the Main and Tail conditional flows use the same underlying network architecture but learn complementary transport behavior through separate parameters, distinct conditioning representations, and different base distributions. The two conditional vector fields are subsequently combined through the adaptive vector-field fusion mechanism described in the main text.

B.3 Implementation Details of Seismic Tail Information Screening

This section provides the implementation details of the Seismic Tail Information Screening (Seis TIS) module introduced in the main text, including the Seis Main predictor, cross-fitting prediction, and the construction of prediction-error-tail supervision.

Seis Main Predictor. The Seis Main predictor adopts a two-stage nonlinear prediction structure consisting of *dense timing estimation + conditional residual flow*. Given a normalized multi-component waveform $\widetilde{\mathbf{X}} \in \mathbb{R}^{C\times T}$, a U-Net-like dense timing network first extracts temporally aligned multi-scale waveform features. The encoder consists of convolutional blocks with progressively increasing channel dimensions and temporal downsampling. After the bottleneck, a symmetric upsampling path restores the temporal resolution, while skip connections fuse temporal features across different scales. Each convolutional block consists of a one-dimensional convolution, batch normalization, and a nonlinear activation, producing the final temporal feature map $\mathbf{F}^{\text{Seis}} \in \mathbb{R}^{D\times T}$.

For the dense timing readout, the network further maps $\mathbf{F}^{\mathrm{Seis}}$ to a sequence of temporal logits $\mathbf{a} = (a_1, \ldots, a_T)$, which are normalized along the temporal dimension:

$$p_t = \frac{\exp(a_t)}{\sum_{\tau=1}^{T} \exp(a_\tau)}. \tag{27}$$

Let τ_t denote the physical time corresponding to the t-th sampling position. The dense arrival-time center is then computed as

$$\mu(\mathbf{X}) = \sum_{t=1}^{T} p_t \tau_t. \tag{28}$$

Using the same temporal weights, the dense feature map is simultaneously aggregated into a waveform-conditioned representation:

$$\bar{\mathbf{f}}^{\mathrm{Seis}} = \sum_{t=1}^{T} p_t \mathbf{f}_t^{\mathrm{Seis}}, \qquad \mathbf{h}^{\mathrm{Seis}} = P^{\mathrm{Seis}}\left(\bar{\mathbf{f}}^{\mathrm{Seis}}\right), \tag{29}$$

where $\mathbf{f}_t^{\mathrm{Seis}}$ denotes the local feature of $\mathbf{F}^{\mathrm{Seis}}$ at temporal position t, and P^{Seis} is a nonlinear projection network used to construct the conditioning representation.

Based on this representation, the conditional residual flow further models waveform-dependent residual structure that is not explained by the dense timing center. For sample i, we first define the signed residual relative to the dense timing center as $\varepsilon_i^{\mathrm{Seis}} = y_i - \mu(\mathbf{X}_i)$, and standardize it according to the residual statistics estimated from the corresponding training split:

$$\tilde{\varepsilon}_i^{\mathrm{Seis}} = \frac{\varepsilon_i^{\mathrm{Seis}} - \bar{\varepsilon}}{s_\varepsilon}, \tag{30}$$

where $\bar{\varepsilon}$ and s_ε denote the center and scale statistics of the signed residuals, respectively.

Using a Gaussian variable $z_0 \sim \mathcal{N}(0, 1)$ as the source distribution, the conditional residual flow learns a continuous transport from z_0 to $\tilde{\varepsilon}_i^{\mathrm{Seis}}$, conditioned on $\mathbf{h}^{\mathrm{Seis}}$. During inference, the learned conditional vector field is integrated from the zero initial state to obtain the terminal residual correction $z_\theta^{(L)}(0)$. The final Seis prediction is then given by

$$\hat{y}_i^{\mathrm{Seis}} = \mu(\mathbf{X}_i) + \bar{\varepsilon} + s_\varepsilon z_\theta^{(L)}(0), \tag{31}$$

where L denotes the number of numerical integration steps.

Therefore, the Seis Main predictor first uses the dense timing network to estimate the dominant arrival-time location and then employs a conditional residual flow to model and correct waveform-dependent residual structure not captured by the dense timing estimate.

Cross-Fitting Prediction. To ensure that the errors used for tail screening reflect out-of-sample predictive behavior as closely as possible, we adopt K-fold cross-fitting (Wolpert, 1992; Chernozhukov et al., 2018). The training set is partitioned into $\{\mathcal{D}_1, \ldots, \mathcal{D}_K\}$. For the k-th fold, an independent Seis Main predictor $f_{\theta^{(-k)}}$ is trained on the complementary subset $\mathcal{D}_{-k} = \bigcup_{j \neq k} \mathcal{D}_j$, and predictions are generated only for the held-out subset $\mathcal{D}_k$:

$$\hat{y}_i^{\mathrm{Seis}} = f_{\theta^{(-k)}}(\mathbf{X}_i), \qquad i \in \mathcal{D}_k. \tag{32}$$

After iterating over all K folds, every training sample receives exactly one Seis prediction from a predictor that was fitted without using that sample. The held-out predictions from all folds are then restored to their original sample order and aggregated to form the complete set of out-of-fold Seis predictions over the training data.

Prediction-Error-Tail Supervision Construction. For each training sample, we compute the absolute prediction error between the ground-truth phase-arrival time and its corresponding out-of-fold Seis prediction:

$$r_i^{\mathrm{Seis}} = \left| y_i - \hat{y}_i^{\mathrm{Seis}} \right|. \tag{33}$$

The screening threshold is determined exclusively from the out-of-fold Seis prediction-error distribution on the training set:

$$u^{\mathrm{Seis}} = Q_\gamma\left(\{r_i^{\mathrm{Seis}}\}_{i=1}^N\right), \tag{34}$$

where $Q_\gamma(\cdot)$ denotes the empirical γ-quantile.

Seis TIS then defines the binary high-error indicator

$$q_i^{\mathrm{Seis}} = \mathbb{I}\left(r_i^{\mathrm{Seis}} > u^{\mathrm{Seis}}\right), \tag{35}$$

and forms the corresponding high-error subset

$$\mathcal{D}_{\mathrm{tail}} = \left\{(\mathbf{X}_i, y_i) : q_i^{\mathrm{Seis}} = 1\right\}. \tag{36}$$

Samples satisfying $q_i^{\mathrm{Seis}} = 1$ therefore belong to the upper tail of the out-of-fold Seis prediction-error distribution and provide tail-specific supervision for high-error-aware learning. Importantly, the threshold u^{Seis}, the training-time labels q_i^{Seis}, and the ground-truth arrival times y_i are not provided as additional inputs during inference.

B.4 Location Prediction and Residual Parameterization

Conditional Location Prediction. SeisTailFlow models the conditional phase-arrival time as the sum of an input-dependent location term and a stochastic residual term:

$$Y = \mu(\mathbf{X}) + Z. \tag{37}$$

Here, $\mu(\mathbf{X})$ represents the waveform-conditioned arrival-time center, while Z describes the conditional stochastic deviation around this center. This parameterization separates prediction of the arrival-time center from probabilistic modeling of the conditional residual.

Because the Main pathway primarily captures dominant phase-arrival patterns, we estimate the location term from the Main conditioning representation $\mathbf{h}_{\mathrm{main}}$:

$$\mu(\mathbf{X}) = L_\eta\left(\mathbf{h}_{\mathrm{main}}\right), \tag{38}$$

where $L_\eta : \mathbb{R}^{D_h} \to \mathbb{R}$ denotes a learnable MLP-based location mapping parameterized by η, D_h is the dimensionality of the Main conditioning representation, and $\mu(\mathbf{X}) \in \mathbb{R}$ is the conditional arrival-time center for the input waveform. With this decomposition, the fused flow does not need to generate the complete absolute phase-arrival time directly from the base distribution, but instead models the conditional stochastic residual relative to $\mu(\mathbf{X})$.

Location Supervision. To ensure that $\mu(\mathbf{X})$ stably represents the arrival-time center, we impose explicit probabilistic supervision on the location prediction. Specifically, using $\mu(\mathbf{X})$ as the location parameter, we model deviations around the location prediction with a Student-t distribution:

$$p_{\mathrm{loc}}\left(Y \mid \mathbf{X}\right) = p_{\text{Student-}t}\left(Y \mid \mu(\mathbf{X}), s_{\mathrm{res}}(\mathbf{X}), \nu\right), \tag{39}$$

where $s_{\mathrm{res}}(\mathbf{X}) > 0$ denotes an input-dependent residual scale and ν is the degrees-of-freedom parameter of the Student-t distribution. The corresponding location-supervision loss is

$$\mathcal{L}_{\mathrm{loc}} = -\mathbb{E}_{(\mathbf{X},Y)}\left[\log p_{\text{Student-}t}\left(Y \mid \mu(\mathbf{X}), s_{\mathrm{res}}(\mathbf{X}), \nu\right)\right]. \tag{40}$$

Compared with direct squared-error supervision, the Student-t likelihood is more tolerant of large arrival-time residuals, thereby reducing the tendency of a small number of extreme deviations to dominate the learning of the location term.

Residual Transport and Reconstruction. After obtaining the location term $\mu(\mathbf{X})$, the fused Main–Tail flow models the remaining conditional stochastic residual. Given an initial random state z_0, continuous transport under the fused vector field v_{fused} produces the terminal state

$$z_1 = \mathcal{T}_{v_{\mathrm{fused}}}\left(z_0 \mid \mathbf{X}\right). \tag{41}$$

The terminal state is then added to the conditional location term to reconstruct the phase-arrival time:

$$\hat{Y} = \mu(\mathbf{X}) + z_1. \tag{42}$$

Thus, $\mu(\mathbf{X})$ determines the waveform-conditioned arrival-time center, whereas z_1 represents the conditional stochastic residual learned by the fused flow. Together, they define the final conditional phase-arrival prediction.

C TRAINING ALGORITHM OF SEISTAILFLOW

Algorithm 1 Training Procedure of SeisTailFlow

Input: Training set $\mathcal{D}_{\mathrm{tr}} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$; event-wise folds K; tail quantile q; loss weights $\lambda_{\mathrm{loc}}, \lambda_{\mathrm{tail}}, \lambda_{\mathrm{GPD}}, \lambda_{\mathrm{ECF}}$
Output: Trained parameters Θ

Initialize SeisTailFlow parameters Θ

Module I: Shared Waveform Representation

For each mini-batch $\mathcal{B}$, encode $\mathbf{X}_{\mathcal{B}}$ as $\mathbf{h}_{\mathrm{wave}} \leftarrow \mathrm{WaveEnc}(\mathbf{X}_{\mathcal{B}})$

Module II: Dual Conditional Flow Learning

Seismic Tail Information Screening (Seis TIS)

Partition $\mathcal{D}_{\mathrm{tr}}$ into K event-disjoint folds $\{\mathcal{D}^{(k)}\}_{k=1}^{K}$
for $k = 1, \ldots, K$ **do**
  Train the OOF predictor on $\mathcal{D}_{\mathrm{tr}} \setminus \mathcal{D}^{(k)}$ and compute $r_i^{\mathrm{OOF}} = |y_i - \hat{y}_i^{(-k)}|$, $i \in \mathcal{D}^{(k)}$
end for
$u \leftarrow Q_q(\{r_i^{\mathrm{OOF}}\}_{i=1}^{N})$; $\tau_i \leftarrow \mathbb{I}(r_i^{\mathrm{OOF}} > u)$; $e_i \leftarrow r_i^{\mathrm{OOF}} - u$ for $\tau_i = 1$

Joint Main–Tail Flow Learning

for each training epoch and mini-batch $\mathcal{B}$ **do**
  $\mathbf{h}_{\mathrm{wave}} \leftarrow \mathrm{WaveEnc}(\mathbf{X}_{\mathcal{B}})$; $\mathbf{h}_{\mathrm{main}} \leftarrow P_{\mathrm{main}}(\mathbf{h}_{\mathrm{wave}})$; $\mathbf{h}_{\mathrm{tail}} \leftarrow P_{\mathrm{tail}}(\mathbf{h}_{\mathrm{wave}})$
  $\mathbf{z}_{\mathrm{spec}} \leftarrow \mathrm{SpecEnc}(\mathbf{X}_{\mathcal{B}}, \mathbf{X}_{\mathcal{B}}^{\mathrm{bg}})$; $\widetilde{\mathbf{h}}_{\mathrm{tail}} \leftarrow \mathcal{M}_{\mathrm{spec}}(\mathbf{h}_{\mathrm{tail}}, \mathbf{z}_{\mathrm{spec}})$
  Sample $t \sim \mathcal{U}(0,1)$, $z_0^{\mathrm{main}} \sim \mathcal{N}(0,1)$, and $z_0^{\mathrm{tail}} \sim t_\nu^{\mathrm{std}}$
  Construct the Main and Tail probability paths and compute $\mathcal{L}_{\mathrm{FM}} = \mathcal{L}_{\mathrm{FM}}^{\mathrm{main}} + \mathcal{L}_{\mathrm{FM}}^{\mathrm{tail}}$
  $(\boldsymbol{\pi}_{\mathrm{tail}}, \boldsymbol{\xi}, \boldsymbol{\sigma}) \leftarrow H_{\mathrm{tail}}(\widetilde{\mathbf{h}}_{\mathrm{tail}})$
  Compute $\mathcal{L}_{\mathrm{BCE}}$ from $\{\pi_{\mathrm{tail},i}, \tau_i\}_{i \in \mathcal{B}}$ and $\mathcal{L}_{\mathrm{GPD}}$ from $\{e_i : \tau_i = 1,\ i \in \mathcal{B}\}$

Module III: Adaptive Vector-Field Fusion

  Evaluate $v_{\mathrm{main}}(z_t, t \mid \mathbf{h}_{\mathrm{main}})$ and $v_{\mathrm{tail}}(z_t, t \mid \widetilde{\mathbf{h}}_{\mathrm{tail}})$ on the shared transport state z_t
  $\rho_t \leftarrow \mathrm{Router}(z_t, t, \boldsymbol{\pi}_{\mathrm{tail}})$
  $v_{\mathrm{STF}}(z_t, t) \leftarrow \rho_t v_{\mathrm{main}}(z_t, t \mid \mathbf{h}_{\mathrm{main}}) + (1 - \rho_t) v_{\mathrm{tail}}(z_t, t \mid \widetilde{\mathbf{h}}_{\mathrm{tail}})$
  $\mu \leftarrow L_\eta(\mathbf{h}_{\mathrm{main}})$; obtain conditional samples $\{\hat{y}^{(m)}\}_{m=1}^{M}$ by integrating v_{STF}

Joint Objective

  $\mathcal{L}_{\mathrm{loc}} \leftarrow -\frac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} \log p_{\text{Student-}t}(y_i \mid \mu_i, s_{\mathrm{res},i}, \nu)$
  $\mathcal{L}_{\mathrm{BCE}} \leftarrow -\frac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} [\tau_i \log \pi_{\mathrm{tail},i} + (1 - \tau_i) \log(1 - \pi_{\mathrm{tail},i})]$
  $\mathcal{L}_{\mathrm{GPD}} \leftarrow -\frac{1}{|\mathcal{B}_{\mathrm{tail}}|} \sum_{i \in \mathcal{B}_{\mathrm{tail}}} \log p_{\mathrm{GPD}}(e_i; \xi_i, \sigma_i)$
  $\mathcal{L}_{\mathrm{ECF}} \leftarrow \mathrm{ECFMatch}\left(\{\hat{y}^{(m)}\}_{m=1}^{M}, \{y_i\}_{i \in \mathcal{B}}\right)$
  $\mathcal{L} \leftarrow \mathcal{L}_{\mathrm{FM}} + \lambda_{\mathrm{loc}}\mathcal{L}_{\mathrm{loc}} + \lambda_{\mathrm{tail}}\mathcal{L}_{\mathrm{BCE}} + \lambda_{\mathrm{GPD}}\mathcal{L}_{\mathrm{GPD}} + \lambda_{\mathrm{ECF}}\mathcal{L}_{\mathrm{ECF}}$
  Update Θ by gradient descent on $\mathcal{L}$
end for
return Θ

D SUPPLEMENTARY DETAILS OF SYNTHETIC EXPERIMENTS

D.1 SYNTHETIC DATA GENERATION DETAILS

By jointly varying the SNR, the relative strength of the confounding transient, and its spectral similarity to the target P wave, we construct a continuous range of picking difficulty from clear phase onsets to low-SNR, strong-confounder, and highly spectrally ambiguous conditions. This design allows the major factors affecting phase-picking difficulty to be explicitly controlled while preserving waveform-level randomness and conditional heterogeneity across samples.

Importantly, we do not create large-error samples by perturbing the ground-truth P-wave arrival labels or by injecting additional heavy-tailed label noise. All samples use the generative arrival time τ_P as the ground-truth target. Consequently, large prediction errors arise from the model's difficulty in identifying the true phase under challenging waveform conditions such as low SNR, strong confounding transients, and high spectral ambiguity, rather than from artificially corrupted prediction targets.

Table 4 summarizes the main parameters used in the synthetic data-generation process. In addition to the basic waveform configuration, we explicitly control the SNR, the strength of the preceding confounder, the temporal separation between the confounder and the target P wave, and their spectral similarity, thereby generating a continuum of waveform conditions ranging from clear arrivals to low-SNR, strongly confounded, and spectrally ambiguous cases. All generation parameters are fixed before model training and kept identical across all compared methods.

Table 4: Configuration and parameter distributions used for synthetic waveform generation. Difficulty-controlling variables jointly determine the SNR, strength, temporal proximity, and spectral ambiguity of the pre-P confounder.

Group	Parameter	Symbol	Distribution / Range	Role
Waveform	Sampling rate	f_s	100 Hz	Temporal resolution
	Window length	T	60 s	Observation window
	Components	–	Z, N, E	Three-component waveform
Target P wave	P-wave arrival	τ_P	$\mathcal{U}(10, 50)$ s	Ground-truth arrival
	Characteristic frequency	f_P	$\mathcal{U}(2, 12)$ Hz	Spectral content
	Relative amplitude	A_P	1.0	Reference amplitude
	Polarization	$\mathbf{p}_P$	Random unit vector	Component variability
Background noise	Signal-to-noise ratio	SNR	$\mathcal{U}(-5, 20)$ dB	**Difficulty control**
	Noise spectrum	θ_N	Colored, sample-dependent	Background variability
Pre-P confounder	Amplitude ratio	A_C/A_P	$\mathcal{U}(0.05, 0.90)$	**Difficulty control**
	Arrival offset	$\Delta\tau_C$	$\mathcal{U}(0.20, 1.50)$ s	**Temporal ambiguity**
	Confounder arrival	τ_C	$\tau_P - \Delta\tau_C$	Pre-P transient
	Spectral similarity	ρ_{spec}	$\mathcal{U}(0, 0.95)$	**Difficulty control**
	Characteristic frequency	f_C	$(1 - \rho_{\text{spec}})\tilde{f} + \rho_{\text{spec}} f_P$	Spectral ambiguity

The final synthetic dataset contains $20{,}000$ training samples, $4{,}000$ validation samples, and $4{,}000$ test samples, which are generated using independent random seeds. All compared methods use exactly the same data splits, with no overlap in samples or latent generative instances across the training, validation, and test sets. All data-dependent statistics, tail supervision signals, and corresponding thresholds are estimated exclusively from the training set. The validation set is used only for model selection, while the test set is reserved for final performance evaluation.

Before conducting the formal experiments, we freeze the data-generating process and all associated parameter configurations. We further perform integrity checks on the generated waveforms, ground-truth P-wave arrival times, and conditioning variables including SNR, confounder strength, and spectral similarity. These checks ensure the absence of non-finite values, invalid arrival times, and duplicated samples across different data splits.

D.2 Evaluation Protocol and Metrics

We evaluate all methods from six complementary perspectives: overall picking accuracy (**Overall Picking**), tail-error behavior (**Tail Error**), low-SNR robustness (**Low-SNR Robustness**), extreme-error behavior (**Extreme Error**), predictive sharpness (**Sharpness**), and probabilistic calibration (**Calibration**). These six dimensions correspond directly to the metric groups reported in the main experimental tables. The first four dimensions primarily characterize point-prediction accuracy and the resulting error distribution, whereas the latter two evaluate the quality of the conditional predictive distribution.

Let the test set contain N samples. For the i-th sample, let the ground-truth P-wave arrival time be τ_i and the corresponding point prediction be $\hat{\tau}_i$. For probabilistic methods, $\hat{\tau}_i$ denotes the point estimate derived from the predictive distribution; for SeisTailFlow, this is the predictive mean defined in Eq. 15 of the main text. We define the absolute arrival-time error as

$$e_i = |\hat{\tau}_i - \tau_i|, \qquad i = 1, \dots, N. \tag{43}$$

All arrival-time errors are measured in seconds (s). For probabilistic methods, we further denote the conditional predictive distribution by $p(\tau \mid \mathbf{x}_i)$ and its cumulative distribution function by $F_i(\tau)$.

Overall Picking Accuracy. The first evaluation dimension measures average seismic phase-picking accuracy over the complete test distribution. We use the mean absolute error (MAE),

$$\mathrm{MAE} = \frac{1}{N} \sum_{i=1}^{N} e_i. \tag{44}$$

MAE measures the average absolute deviation between the predicted and ground-truth P-wave arrival times, with lower values indicating more accurate overall phase picking. This metric is reported as **MAE** under the **Overall** column of the main experimental tables.

Tail Error. The second dimension evaluates prediction errors in the high-error region of the arrival-time error distribution. We first report high quantiles of the absolute-error distribution. For a quantile level $\alpha \in (0, 1)$,

$$P_\alpha = Q_\alpha\left(\{e_i\}_{i=1}^{N}\right), \tag{45}$$

where $Q_\alpha(\cdot)$ denotes the empirical α-quantile. In particular, the main experiments report

$$P95 = P_{0.95}, \qquad P99 = P_{0.99}. \tag{46}$$

P95 and P99 characterize increasingly extreme regions of the prediction-error distribution, with P99 placing greater emphasis on rare but severe arrival-time deviations.

We additionally evaluate the average error among samples belonging to the designated high-error region. Let u denote an evaluation threshold fixed before test evaluation, and define

$$\mathcal{T}(u) = \{i : e_i > u\}. \tag{47}$$

The test set is never used to tune or select u. Over the resulting model-specific exceedance subset, Tail MAE is defined as

$$\mathrm{TailMAE}(u) = \frac{1}{|\mathcal{T}(u)|} \sum_{i \in \mathcal{T}(u)} e_i. \tag{48}$$

Tail MAE therefore measures the model-conditional average magnitude of errors exceeding the predefined threshold.

The evaluation threshold used for Tail MAE is an evaluation quantity and should not be confused with the Seis TIS threshold u_{Seis} introduced in the training procedure. The latter is constructed from approximately out-of-sample Seis residuals to provide prediction-error-tail supervision for SeisTailFlow, whereas the present metric is used only for evaluating prediction errors.

Low-SNR Robustness. The third dimension evaluates robustness when the P-wave arrival is difficult to distinguish from background noise. For the synthetic dataset, SNR is an explicitly controlled variable in the data-generating process, allowing the test samples to be stratified independently of model predictions. For the real-data experiments, the same principle is applied using the pre-specified SNR stratification defined by the corresponding evaluation protocol.

Let the fixed SNR bins be $\{\mathcal{B}_k^{\mathrm{SNR}}\}_{k=1}^{K}$, where

$$\mathcal{B}_k^{\mathrm{SNR}} = \{i : b_{k-1} \le \mathrm{SNR}_i < b_k\}\,. \tag{49}$$

For the k-th SNR subgroup, we compute

$$\mathrm{MAE}_k^{\mathrm{SNR}} = \frac{1}{|\mathcal{B}_k^{\mathrm{SNR}}|} \sum_{i \in \mathcal{B}_k^{\mathrm{SNR}}} e_i. \tag{50}$$

We analogously evaluate $P95_k^{\mathrm{SNR}}$, $P99_k^{\mathrm{SNR}}$, and Tail MAE within individual SNR subgroups when conducting detailed stratified analyses.

The **Low MAE** reported in the main tables denotes the MAE evaluated on the pre-specified lowest-SNR subgroup. Thus, it directly measures phase-picking accuracy under the waveform conditions in which the signal is weakest relative to the background noise.

To quantify the relative improvement over a reference method under a given SNR condition, for an error metric M we additionally define

$$\mathrm{RER}_k(M) = \frac{M_k^{\mathrm{ref}} - M_k^{\mathrm{model}}}{M_k^{\mathrm{ref}}} \times 100\%. \tag{51}$$

A larger RER_k indicates a greater reduction in error relative to the reference method under the corresponding SNR condition.

For the synthetic experiments, we further apply the same stratified evaluation principle to the confounder-to-P amplitude ratio A_C/A_P and the confounder–P-wave spectral similarity ρ_{spec}. These analyses characterize how prediction errors change as the preceding transient becomes stronger or spectrally more similar to the true P wave. All subgroup boundaries are fixed before computing test performance and are applied consistently to all compared methods.

Extreme-Error Behavior. The fourth dimension evaluates the frequency and severity of large arrival-time errors. Unlike the Tail Error dimension, which focuses on high-error quantiles and model-conditional error magnitude, this dimension characterizes how often large errors occur and how severe the most extreme errors become.

We first use expected shortfall (ES) to quantify the mean severity of errors beyond a high quantile. For a probability level α, let

$$q_\alpha = Q_\alpha\left(\{e_i\}_{i=1}^{N}\right). \tag{52}$$

The expected shortfall is then

$$\mathrm{ES}_\alpha = \frac{\sum_{i=1}^{N} e_i \mathbb{I}(e_i > q_\alpha)}{\sum_{i=1}^{N} \mathbb{I}(e_i > q_\alpha)}, \tag{53}$$

where $\mathbb{I}(\cdot)$ denotes the indicator function. Whereas P_α specifies the boundary of a high-error region, ES_α measures the average error magnitude beyond that boundary. The main experimental tables report $\mathrm{ES@99} = \mathrm{ES}_{0.99}$.

We also report a fixed-threshold exceedance rate. For an error tolerance δ,

$$\widehat{P}_{\mathrm{exc}}(\delta) = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}(e_i > \delta). \tag{54}$$

This quantity measures the proportion of test samples whose arrival-time error exceeds a fixed evaluation tolerance. In the main experimental tables,

$$\mathrm{Exc.@0.3} = \widehat{P}_{\mathrm{exc}}(0.3\,\mathrm{s}), \tag{55}$$

i.e., the percentage of test samples with an absolute arrival-time error greater than 0.3 s. Lower values indicate fewer large-error events.

For more detailed high-error-tail analysis, given a high threshold u, we define the exceedance magnitude as

$$\zeta_i = e_i - u, \qquad e_i > u, \tag{56}$$

and estimate the empirical survival function

$$\widehat{S}(\zeta) = \frac{1}{|\mathcal{T}(u)|} \sum_{i \in \mathcal{T}(u)} \mathbb{I}(\zeta_i > \zeta). \tag{57}$$

This representation allows direct comparison of how rapidly the high-error tail decays across different methods.

SeisTailFlow additionally incorporates GPD-based modeling of high-threshold exceedances during training. For an exceedance variable $\zeta > 0$, the GPD is written as

$$\zeta \sim \mathrm{GPD}(\xi, \sigma), \tag{58}$$

where ξ and σ denote the shape and scale parameters. For $\xi \neq 0$, its cumulative distribution function is

$$G(\zeta; \xi, \sigma) = 1 - \left(1 + \frac{\xi\zeta}{\sigma}\right)^{-1/\xi}, \qquad \zeta \geq 0, \tag{59}$$

subject to $1 + \xi\zeta/\sigma > 0$.

In SeisTailFlow, the GPD parameters used during optimization are predicted from the spectrally enhanced waveform condition and are therefore waveform dependent. When an aggregate GPD fit is shown in the evaluation, it is used only as a diagnostic for comparing empirical tail behavior and should not be confused with the conditional GPD parameterization used in model training.

Predictive Sharpness. The fifth dimension evaluates how concentrated the probabilistic predictive distribution remains while maintaining the required level of empirical coverage.

For a nominal coverage level $1 - \alpha$, let the central prediction interval produced by a model be

$$I_i^{(1-\alpha)} = \left[L_i^{(1-\alpha)}, U_i^{(1-\alpha)}\right], \tag{60}$$

where $L_i^{(1-\alpha)}$ and $U_i^{(1-\alpha)}$ denote the corresponding lower and upper predictive quantiles. The interval width is

$$W_i^{(1-\alpha)} = U_i^{(1-\alpha)} - L_i^{(1-\alpha)}, \tag{61}$$

and its mean over the test set is

$$\mathrm{Width}_{1-\alpha} = \frac{1}{N} \sum_{i=1}^{N} W_i^{(1-\alpha)}. \tag{62}$$

Narrower prediction intervals indicate greater sharpness only when the associated coverage is comparable. A model could otherwise obtain an artificially narrow interval simply by under-covering the observations. We therefore use *coverage-matched sharpness*, which compares interval widths at the same empirical coverage level.

Let the target empirical coverage be c. For model m, define

$$\mathrm{CMS}_m(c) = \mathrm{Width}_m \;\big|\; \widehat{C}_m \approx c, \tag{63}$$

where $\widehat{C}_m$ denotes the empirical coverage achieved by model m. At matched coverage, a smaller $\mathrm{CMS}_m(c)$ indicates that the same level of empirical coverage is achieved using a narrower predictive interval.

The **W@95** reported in the main tables corresponds to the coverage-matched mean interval width at an empirical coverage level of approximately 95%:

$$\mathrm{W@95} = \mathrm{CMS}(0.95). \tag{64}$$

Thus, W@95 should be interpreted jointly with coverage rather than as an unconstrained interval-width measure.

Probabilistic Calibration. The sixth dimension evaluates whether the uncertainty represented by the conditional predictive distribution is statistically consistent with the observed outcomes.

For a nominal coverage level $1-\alpha$, the empirical coverage is defined as

$$\widehat{C}_{1-\alpha} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}\left(\tau_i \in I_i^{(1-\alpha)}\right). \tag{65}$$

An ideally calibrated predictive distribution should satisfy

$$\widehat{C}_{1-\alpha} \approx 1-\alpha. \tag{66}$$

We therefore define the corresponding coverage error as

$$\mathrm{CE}_{1-\alpha} = \left|\widehat{C}_{1-\alpha} - (1-\alpha)\right|. \tag{67}$$

A smaller CE indicates that the empirical coverage is closer to the nominal coverage level. The main experimental tables report

$$\mathrm{CE@95} = \left|\widehat{C}_{0.95} - 0.95\right|. \tag{68}$$

To assess calibration over multiple coverage levels, we further use a reliability curve comparing nominal coverage levels $\mathcal{C} = \{c_1, \ldots, c_J\}$ with their corresponding empirical coverage values. The average calibration error is

$$\mathrm{ACE} = \frac{1}{J}\sum_{j=1}^{J} \left|\widehat{C}(c_j) - c_j\right|, \tag{69}$$

where an ideally calibrated model satisfies $\widehat{C}(c) = c$ across the evaluated coverage range.

We additionally use the continuous ranked probability score (CRPS) to assess the overall quality of the complete conditional predictive distribution:

$$\mathrm{CRPS}\left(F_i, \tau_i\right) = \int_{-\infty}^{+\infty} \left[F_i(z) - \mathbb{I}(\tau_i \le z)\right]^2 \mathrm{d}z, \tag{70}$$

with the test-set average

$$\overline{\mathrm{CRPS}} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{CRPS}(F_i, \tau_i). \tag{71}$$

CRPS jointly reflects the location and dispersion of the predictive distribution relative to the observation, with lower values indicating better probabilistic prediction quality. Although CRPS is grouped under the probabilistic-evaluation block in the main tables, it evaluates the predictive distribution as a whole rather than calibration alone.

Accordingly, probabilistic prediction is not evaluated using a single criterion. We jointly consider CRPS, coverage error, reliability, and coverage-matched sharpness: calibration measures whether reported predictive probabilities agree with their empirical frequencies; sharpness measures how concentrated the predictive distribution remains at comparable coverage; and CRPS provides an integrated assessment of the full conditional predictive distribution. For methods that provide only deterministic point predictions and do not produce a native conditional predictive distribution, we do not report CRPS, coverage, sharpness, or reliability metrics.

Six-Dimensional Summary. For clarity, the six evaluation dimensions reported in the main experiments are summarized as

$$\begin{aligned}
\text{Overall Picking}: &\quad \mathrm{MAE},\\
\text{Tail Error}: &\quad P95,\ P99,\ \mathrm{TailMAE},\\
\text{Low-SNR Robustness}: &\quad \mathrm{LowMAE},\\
\text{Extreme Error}: &\quad \mathrm{ES@99},\ \mathrm{Exc.@0.3},\\
\text{Sharpness}: &\quad \mathrm{W@95},\\
\text{Calibration}: &\quad \mathrm{CRPS},\ \mathrm{CE@95}.
\end{aligned} \tag{72}$$

All reported error, extreme-error, sharpness, and calibration metrics are lower-is-better. The six dimensions are designed to distinguish average picking accuracy from high-error behavior, conditional robustness, extreme-error severity and frequency, and the quality of probabilistic predictions.

D.3 Extended Tail and Probabilistic Evaluation

The main text reports the core metrics of the six-dimensional evaluation protocol, including P95, P99, Tail MAE, ES@99, Exc.@0.3, W@95, CRPS, and CE@95. Here, we extend this evaluation to additional high-error levels, fixed error thresholds, and probabilistic operating points in order to provide a more fine-grained characterization of high-error behavior and predictive uncertainty. Specifically, we additionally report P90 for tail-error characterization, ES@95 together with exceedance probabilities at error thresholds of 0.2, 0.3, and $0.5\,\mathrm{s}$ for extreme-error behavior, and calibration error and coverage-matched interval width at 90% and 95% coverage levels. These complementary operating points allow us to examine whether the conclusions observed in the main text remain consistent across progressively more stringent high-error and probabilistic criteria.

Table 5: **Extended tail-error and probabilistic evaluation on the synthetic dataset.** P90 complements the tail-error metrics in the main text; ES and fixed-threshold exceedance rates characterize extreme-error behavior; CE measures calibration error; and W denotes coverage-matched prediction-interval width. Lower is better for all metrics.

	Tail Error	Extreme Error					Calibration		Sharpness	
Method	P90↓	ES@95↓	ES@99↓	Exc.@0.2↓	Exc.@0.3↓	Exc.@0.5↓	CE@90↓	CE@95↓	W@90↓	W@95↓
PhaseNet	0.0146 ±0.0025	0.0267 ±0.009	0.0392 ±0.018	0.34 ±0.05%	0.18 ±0.03%	0.06 ±0.01%	–	–	–	–
EQTransformer	0.0178 ±0.0036	0.0318 ±0.010	0.0417 ±0.016	0.41 ±0.05%	0.24 ±0.03%	0.08 ±0.01%	–	–	–	–
Armstrong et al.	0.0197 ±0.0032	0.0345 ±0.009	0.0449 ±0.015	0.46 ±0.04%	0.27 ±0.02%	0.09 ±0.01%	0.0540 ±0.003	0.0610 ±0.003	0.0284 ±0.007	0.0348 ±0.009
Vanilla CFM	0.0875 ±0.0035	0.1458 ±0.010	0.1755 ±0.014	1.37 ±0.06%	0.82 ±0.02%	0.31 ±0.03%	0.0430 ±0.003	0.0495 ±0.003	0.4410 ±0.008	0.5193 ±0.008
CW-Flow	0.0142 ±0.0046	0.0260 ±0.008	0.0384 ±0.013	0.30 ±0.04%	0.16 ±0.02%	0.05 ±0.01%	0.0475 ±0.002	0.0528 ±0.002	0.0258 ±0.006	0.0317 ±0.007
SeisTailFlow	**0.0128** ±0.0024	**0.0196** ±0.006	**0.0228** ±0.010	**0.16** ±0.02%	**0.09** ±0.01%	**0.02** ±0.01%	**0.0382** ±0.001	**0.0453** ±0.001	**0.0235** ±0.005	**0.0293** ±0.006

Table 5 extends the main-text results to additional operating points of the error and predictive distributions. The same overall pattern remains visible when the evaluation is expanded from the core metrics reported in the main text to less and more stringent high-error criteria.

SeisTailFlow consistently reduces both high-error quantiles and expected shortfall relative to the baselines. This consistent improvement across complementary high-error metrics shows that the gain extends beyond individual error quantiles to the severity of errors in the extreme tail.

The fixed-threshold analysis provides a complementary view of extreme-error behavior. Across the evaluated thresholds, SeisTailFlow consistently exhibits lower exceedance rates than the baselines, with the reduction persisting from moderately large errors to rarer and more severe deviations. Together with the high-quantile and expected-shortfall results, this shows that the improvement extends across complementary aspects of high-error behavior, including the magnitude and frequency of large arrival-time errors.

For probabilistic prediction, calibration and sharpness are evaluated jointly rather than interpreting interval width in isolation. At matched empirical coverage, SeisTailFlow produces sharper predictive intervals while maintaining favorable calibration relative to the probabilistic baselines. The same pattern is observed across the evaluated coverage levels, indicating that the gain in sharpness is not achieved by sacrificing empirical coverage.

D.4 Tail-Error Distribution and Extreme-Error Analysis

To further examine whether the improvements of SeisTailFlow extend beyond average prediction accuracy, we analyze the full high-error behavior of the absolute arrival-time error distribution. Rather than relying on a single tail statistic, we jointly examine the empirical error distribution, the complementary cumulative distribution function (CCDF), high-error summary metrics, and fixed-threshold exceedance probabilities. The corresponding metric definitions follow Appendix D.2.

Figure 5 provides a distribution-level view of the large-error behavior of the compared methods. Figure 5(a) shows the absolute-error distribution, while Figure 5(b) presents the corresponding CCDF, $\Pr(e > r)$. The CCDF directly characterizes how rapidly the probability of observing an error larger than r decays as the error threshold increases. Compared with overall MAE, these two views place greater emphasis on the relatively small fraction of samples associated with large arrival-time deviations and therefore reveal differences that can be obscured by the dominant low-error region.

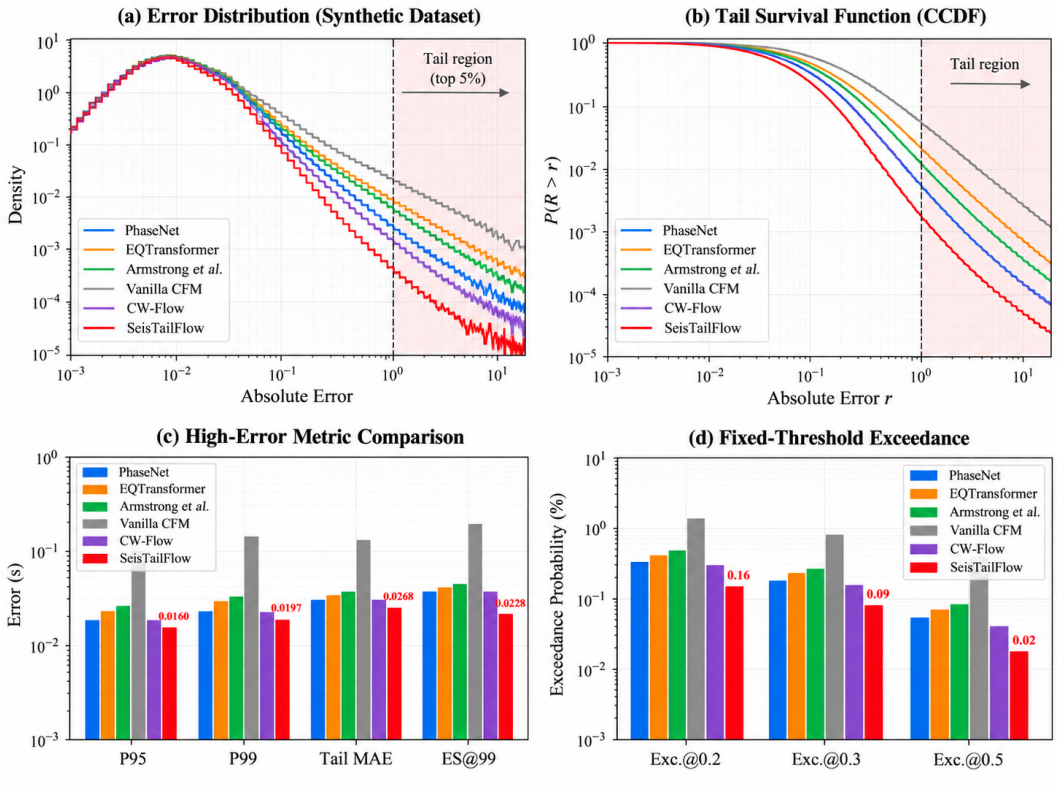


Figure 5: **Tail-error distribution and extreme-error analysis on the synthetic dataset.** (a) Distribution of absolute arrival-time errors, with the high-error region highlighted. (b) Complementary cumulative distribution function (CCDF), showing the probability of exceeding a given error magnitude. (c) Comparison of P95, P99, Tail MAE, and ES@99. (d) Fixed-threshold exceedance probabilities at 0.2, 0.3, and 0.5 s. Lower values indicate better high-error behavior in panels (c) and (d).

The quantitative comparison in Figure 5(c) further confirms this behavior across complementary high-error statistics. SeisTailFlow achieves lower P95, P99, Tail MAE, and ES@99 than the compared methods.

Their consistent improvement indicates that the gain is not tied to a single high-error statistic.

Figure 5(d) examines the same phenomenon from a fixed-threshold perspective. For a threshold δ, the exceedance probability $\Pr(e > \delta)$ measures how frequently arrival-time errors exceed a specified tolerance. Evaluating additional thresholds at 0.2 and 0.5 s extends this comparison from moderately large errors to increasingly rare and severe deviations.

The results show that the improvement of SeisTailFlow is not confined to the central part of the error distribution, but extends to the frequency and severity of large arrival-time errors. This distribution-level evidence complements the aggregate tail-error metrics reported in the main text and provides a more complete characterization of extreme prediction-error behavior.

D.5 Difficulty-Stratified Analysis

An important advantage of the synthetic dataset is that the waveform factors governing picking difficulty are explicitly controlled and therefore can be analyzed directly. Beyond the aggregate metrics reported in the main text, we further examine model performance from three complementary perspectives: signal-to-noise ratio (SNR), confounder strength, and spectral similarity. These analyses help determine whether the gains of SeisTailFlow are primarily concentrated in waveform condi-

tions where the true P-wave arrival becomes difficult to distinguish from noise or from structured preceding transients.

SNR-Stratified Performance. We first stratify the test samples according to pre-specified SNR intervals. Figure 6 reports representative error statistics across different SNR conditions.

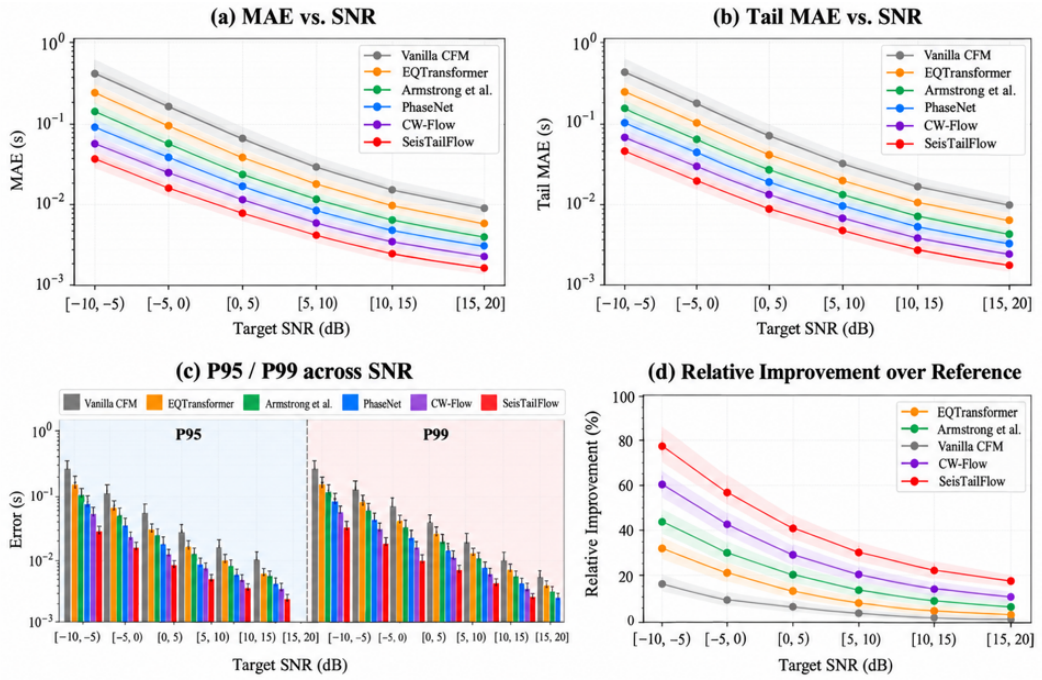


Figure 6: **SNR-stratified performance on the synthetic dataset.** The same SNR bins are applied to all methods. Representative panels compare overall and high-error metrics across SNR levels and summarize the corresponding relative improvement under increasingly noisy waveform conditions.

As SNR decreases, background noise progressively weakens the distinguishability of the true P-wave onset, making accurate arrival-time picking increasingly difficult. This stratified analysis is therefore used to determine whether performance differences among methods are concentrated in low-SNR waveform conditions, rather than being dominated by the large number of relatively easy samples. Consistent with the main-text Low MAE comparison, SeisTailFlow exhibits its largest advantage in the low-SNR regime, while still maintaining competitive performance at moderate and high SNR levels. This pattern indicates that the proposed high-error-aware design is particularly effective when the waveform evidence for the true arrival is weak or partially obscured by background noise.

In addition to overall subgroup MAE, we also examine stratified high-error statistics such as P95, P99, and Tail MAE. These metrics show that as the waveform becomes noisier, all methods incur larger errors, but the increase is slower for SeisTailFlow. This suggests that the observed improvement is not limited to average error reduction, but also extends to the control of high-error events under low-SNR conditions.

Confounder Strength. We further examine picking difficulty by stratifying samples according to the strength of the preceding confounding transient relative to the target P wave. A weak confounder leaves the target onset comparatively distinct, whereas a stronger confounder introduces a more competitive onset-like transient before the true arrival. This creates greater temporal ambiguity and makes accurate identification of the target P-wave onset more challenging.

Spectral Similarity. We further analyze model performance as the preceding confounder becomes spectrally more similar to the target P wave. Greater spectral similarity reduces the frequency-domain contrast between the two transients, making the confounder increasingly difficult to distin-

guish from the true arrival. This setting is particularly challenging when the time-domain waveform provides limited evidence for resolving the two candidate onsets.

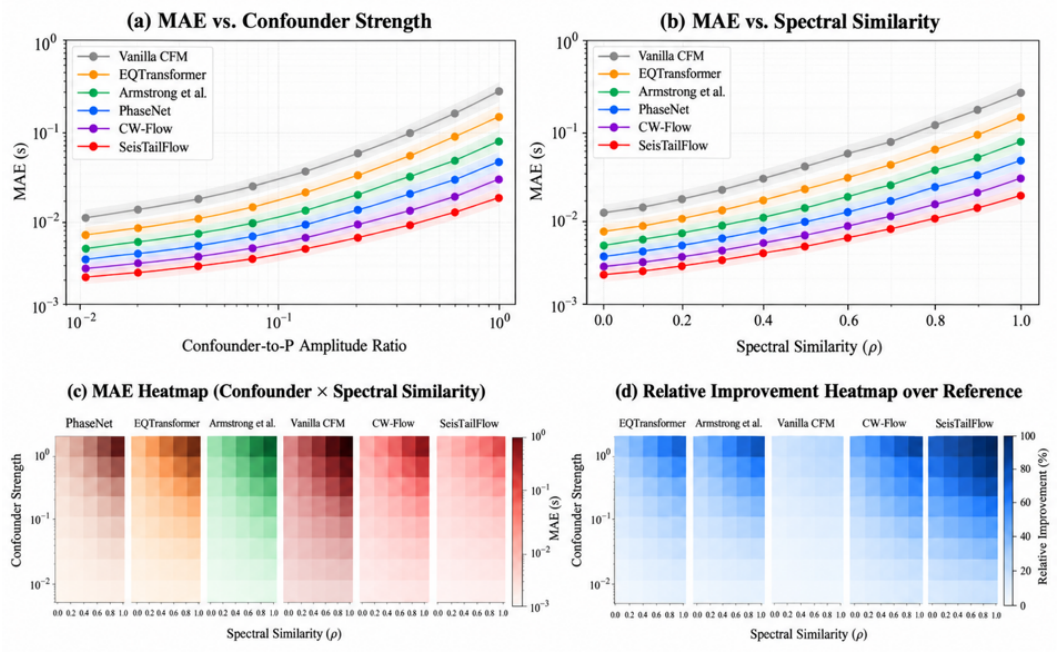


Figure 7: **Performance under controlled waveform difficulty.** Samples are analyzed according to confounder strength and spectral similarity. Representative panels show marginal performance trends along each factor as well as their joint effect on picking difficulty.

Figure 7 summarizes performance under these two additional difficulty factors. As either confounder strength or spectral similarity increases, arrival-time errors rise for all compared methods, reflecting the increasing ambiguity of the waveform. However, SeisTailFlow consistently shows smaller degradation, indicating that its advantage is not restricted to purely low-SNR conditions. Instead, the model remains more robust when the main challenge is the presence of a structured transient that competes with the true P-wave onset.

The joint analysis of confounder strength and spectral similarity provides a more complete view of controlled waveform difficulty. The most challenging cases arise when the confounder is both strong and spectrally similar to the target P wave, because the model must distinguish between competing transients that are similar in both amplitude structure and frequency content. Under these conditions, SeisTailFlow continues to achieve lower error than the compared baselines, supporting the claim that the method improves waveform-conditioned high-error behavior under multiple sources of waveform ambiguity.

Taken together, the SNR-, confounder-, and spectral-similarity-stratified results show that the gains of SeisTailFlow are concentrated in precisely the waveform regimes most likely to produce large arrival-time deviations. This difficulty-aware evidence complements the aggregate results in the main text by showing not only that SeisTailFlow improves overall accuracy, but also that it more effectively controls high-error behavior when waveform conditions become noisy, ambiguous, or structurally confounded.

D.6 Probabilistic Reliability and Sharpness

For probabilistic seismic phase picking, reliable uncertainty estimation requires both accurate calibration and informative predictive intervals. High empirical coverage alone is insufficient, as overly conservative distributions may achieve large coverage by producing unnecessarily wide intervals. Therefore, we jointly evaluate calibration reliability and coverage-matched sharpness.

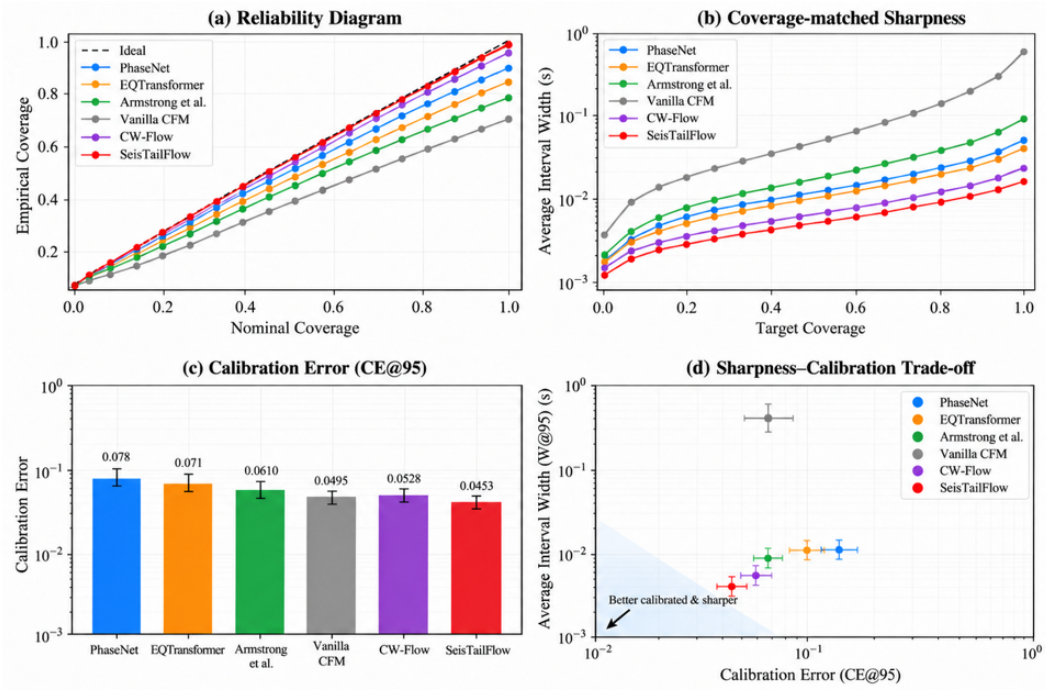


Figure 8: **Reliability, calibration, and coverage-matched sharpness on the synthetic dataset.** (a) Reliability diagrams comparing empirical and nominal coverage, where the diagonal represents ideal calibration. (b) Coverage-matched sharpness measured by the average prediction-interval width at matched coverage levels. (c) Calibration error CE@95 across methods. (d) Joint sharpness–calibration comparison, where the lower-left region indicates simultaneously better calibration and sharper predictive intervals.

Figure 8(a) compares nominal and empirical coverage across different confidence levels. A well-calibrated predictive distribution should remain close to the ideal diagonal, indicating consistency between predicted uncertainty and observed outcomes. Figure 8(b) further evaluates the sharpness of predictive intervals under matched coverage levels. Narrower intervals indicate more concentrated uncertainty estimates when coverage is comparable.

Figure 8(c) summarizes the quantitative calibration error at 95% coverage. SeisTailFlow achieves the lowest CE@95 among the compared methods (0.0453), indicating closer agreement between predicted and empirical coverage. Figure 8(d) provides a joint comparison of calibration and sharpness. The preferred region is the lower-left corner, where prediction intervals are both better calibrated and more concentrated. SeisTailFlow achieves a favorable trade-off by maintaining low calibration error while producing sharper predictive intervals.

Together with the extreme-error and difficulty-stratified analyses, these results demonstrate that SeisTailFlow improves not only point-prediction accuracy but also the reliability and informativeness of the associated predictive distributions.

E ABLATION STUDIES

This section provides a comprehensive analysis of the contribution of each component in SeisTailFlow. Due to the limited space in the main text, we move the complete ablation results to this appendix. We investigate the roles of prediction-error-tail supervision, spectral conditioning, Tail Flow modeling, GPD-based exceedance regularization, and adaptive routing.

Unless otherwise specified, all ablation variants use the same data splits, training budget, optimization settings, and evaluation protocol as the full model. Each experiment is repeated with three independent random seeds, and results are reported as mean $\pm$ standard deviation.

Table 6: **Comprehensive ablation study of SeisTailFlow.** Each column corresponds to a component-level modification of the full model. The upper header groups variants according to their functional roles: tail supervision, spectral conditioning, dual-flow modeling, high-error regularization, and vector-field fusion. Lower rows report performance across overall accuracy, tail-error behavior, extreme-error control, and probabilistic prediction quality. Lower values are better for all metrics.

	Tail Supervision		**Spectral**	**Dual-Flow Modeling**			**High-Error Reg.**	**Fusion**	**Full**
	In-sample Residual	**w/o Seis TIS**	**w/o Spectral Conditioning**	**w/o Tail Flow**	**Main Flow only**	**Tail Flow only**	**w/o GPD Constraint**	**Uniform Mixing**	**SeisTailFlow Full**
MAE↓	0.0075	0.0074	0.0075	0.0080	0.0072	0.0090	0.0073	0.0074	**0.0069**
P95↓	0.0185	0.0181	0.0187	0.0214	0.0205	0.0238	0.0177	0.0180	**0.0160**
P99↓	0.0238	0.0227	0.0241	0.0289	0.0276	0.0321	0.0225	0.0231	**0.0197**
Tail MAE↓	0.0178	0.0170	0.0181	0.0227	0.0213	0.0264	0.0169	0.0173	**0.0152**
ES@99↓	0.0287	0.0275	0.0292	0.0345	0.0331	0.0387	0.0268	0.0278	**0.0228**
Exc.@0.3↓	0.18%	0.15%	0.19%	0.26%	0.24%	0.31%	0.14%	0.16%	**0.09%**
CRPS↓	0.0053	0.0052	0.0054	0.0058	0.0054	0.0063	0.0051	0.0052	**0.0049**
CE@95↓	0.0521	0.0508	0.0534	0.0582	0.0527	0.0641	0.0502	0.0516	**0.0453**
W@95↓	0.0338	0.0329	0.0351	0.0417	0.0345	0.0475	0.0321	0.0334	**0.0293**

Table 6 shows that each component contributes to different aspects of SeisTailFlow. Removing the Tail Flow produces the largest degradation in high-error metrics, increasing P99, Tail MAE, and ES@99. This indicates that dedicated modeling capacity for waveform conditions associated with large prediction errors is important for controlling rare but severe arrival-time deviations.

Removing the GPD constraint mainly affects extreme-error metrics while having limited influence on average accuracy. This indicates that the extreme-value component primarily regularizes high-threshold prediction-error exceedances rather than average prediction behavior.

The spectral conditioning component provides additional robustness by incorporating frequency-domain information from the waveform. Without spectral conditioning, both overall accuracy and high-error metrics degrade, indicating that spectral cues are complementary to time-domain features, especially for ambiguous waveform conditions.

The ablation of Seis TIS supervision further shows the importance of reliable high-error labels derived from approximately out-of-sample residuals. Using in-sample residuals weakens high-error identification because the resulting error signals may be biased toward the training samples.

Finally, replacing adaptive routing with uniform mixing or removing the Main Flow pathway degrades the coordination between dominant-pattern modeling and high-error-aware correction. Unlike prediction-level fusion, adaptive routing coordinates the Main and Tail conditional vector fields during transport, allowing the model to preserve accurate modeling of dominant arrival-time patterns while dynamically increasing Tail-Flow contribution under elevated large-error risk.

Overall, the complete SeisTailFlow achieves the best balance across average accuracy, tail-error behavior, extreme-error control, and probabilistic prediction quality, confirming the complementary roles of the proposed components.

E.1 Mechanistic Analysis of Main–Tail Coordination

To further investigate whether the dual-flow architecture learns complementary functional roles, we separately analyze the behavior of the Main Flow, Tail Flow, and final routed prediction under different waveform conditions. Specifically, we examine their error patterns across SNR, confounder strength, spectral similarity, error distributions, and difficulty groups. These analyses aim to determine whether the Main and Tail pathways develop complementary behavior across regular and high-error-prone waveform conditions rather than learning redundant predictors.

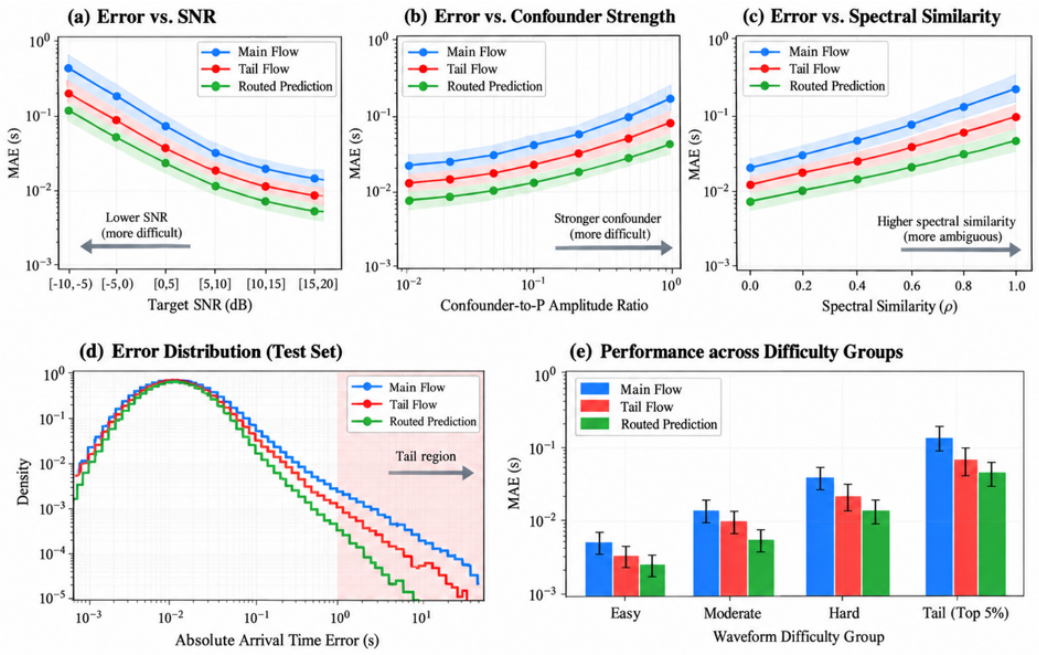


Figure 9: **Main–Tail coordination analysis of SeisTailFlow.** (a)–(c) Arrival-time errors of the Main Flow, Tail Flow, and final routed prediction under varying waveform conditions, including SNR, confounder strength, and spectral similarity. (d) Absolute error distributions of different prediction pathways. (e) Performance comparison across waveform difficulty groups. The results illustrate complementary functional roles of the Main and Tail pathways: the Main Flow remains stable under regular conditions, while the Tail Flow provides additional modeling capacity under ambiguous and high-error-prone conditions.

As shown in Figure 9(a)–(c), the Main Flow achieves stable performance under relatively regular waveform conditions, including high SNR, weak confounders, and low spectral ambiguity. This behavior is consistent with its role in capturing dominant phase-arrival patterns.

In contrast, the Tail Flow becomes increasingly beneficial as waveform conditions become more ambiguous. Its advantage becomes more evident under low-SNR, strong-confounder, and high-spectral-similarity conditions, where large prediction deviations are more likely to occur. The final routed prediction combines these complementary behaviors and maintains robust performance across different waveform regimes.

The error distributions in Figure 9(d) further reveal the complementary behavior of the two pathways. The Main Flow and Tail Flow differ most clearly in the high-error region, while the final routed prediction exhibits a faster decay of extreme errors. This indicates that the improvement of SeisTailFlow is primarily associated with better handling of rare large deviations rather than a uniform shift of the entire prediction-error distribution.

Figure 9(e) summarizes performance across increasing waveform difficulty levels. While the Main Flow performs competitively on easier samples, its degradation becomes more pronounced as waveform ambiguity increases. The Tail Flow provides complementary robustness in these challenging

groups, and the final routed prediction achieves the most stable performance across all difficulty levels.

These results demonstrate that the Main Flow and Tail Flow are not redundant predictors. Instead, they provide complementary modeling capacity across waveform conditions, with the Main Flow emphasizing dominant arrival-time patterns and the Tail Flow providing additional capacity for rare and difficult high-error cases. Their coordination forms the basis for reliable high-error-aware prediction in SeisTailFlow.

E.2 Adaptive Routing Behavior

To understand how adaptive routing coordinates the Main and Tail pathways, we analyze the learned routing weights under different waveform conditions. Unlike fixed-weight or prediction-level fusion, SeisTailFlow performs adaptive vector-field coordination during continuous transport. Therefore, the routing behavior provides direct evidence of how the model adjusts the relative influence of the Main and Tail conditional flows.

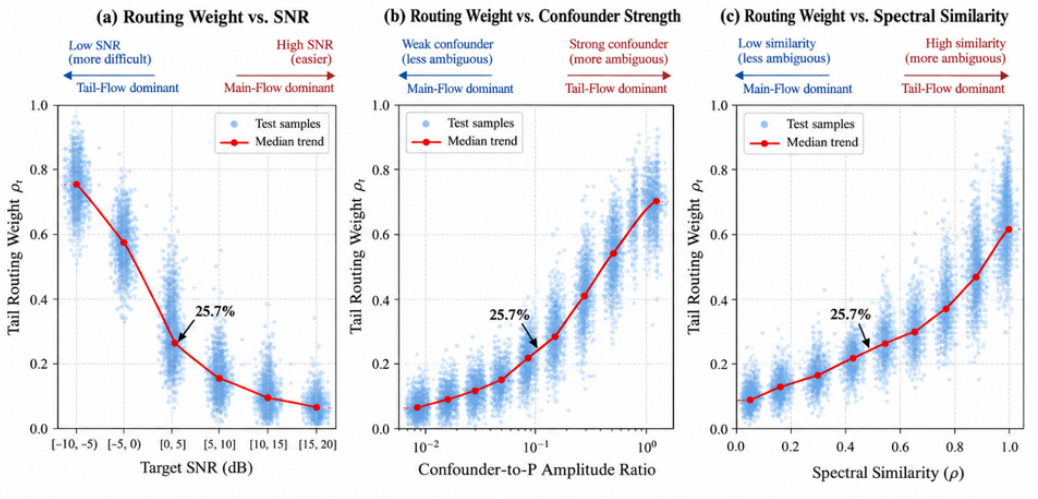


Figure 10: **Adaptive routing behavior of SeisTailFlow.** The learned routing weights are shown as functions of (a) SNR, (b) confounder strength, and (c) spectral similarity. Each point represents one test sample, while the solid curve denotes the conditional median trend. The results show that SeisTailFlow increases the contribution of the Tail pathway under more ambiguous waveform conditions, while relying more on the Main pathway for regular samples.

As shown in Figure 10, the routing weights exhibit clear dependence on waveform characteristics. For high-SNR samples with relatively reliable phase information, the model assigns a larger contribution to the Main Flow, which primarily captures dominant phase-arrival patterns.

In contrast, when SNR decreases or waveform ambiguity increases due to stronger confounders or higher spectral similarity, the routing weight shifts toward the Tail Flow. This behavior indicates that the model identifies waveform conditions associated with elevated large-error risk and dynamically increases the contribution of the high-error-aware pathway.

Such adaptive behavior cannot be achieved by static averaging of the two pathways, since their relative contributions vary across waveform conditions. By adjusting the conditional vector-field contributions during transport, the routing mechanism flexibly coordinates dominant-pattern modeling and high-error-aware correction.

E.3 Component-wise Mechanism Analysis

E.3.1 OOF-based Tail Supervision

A key challenge in high-error-aware learning is constructing reliable supervision signals for rare high-error samples. Using residuals computed on the same samples used for model fitting may lead to optimistic estimates of prediction errors and weaken high-error identification. Therefore, SeisTailFlow adopts out-of-fold (OOF) residuals to construct prediction-error-tail supervision.

Table 7: **Effect of OOF-based tail supervision.** We compare different strategies for constructing tail supervision signals. OOF residuals provide sample-independent error estimates, while in-sample residuals may underestimate generalization errors.

	In-sample Residual	**w/o OOF Supervision**	**OOF-based SeisTailFlow**
MAE↓	0.0075	0.0074	**0.0069**
P99↓	0.0238	0.0229	**0.0197**
Tail MAE↓	0.0178	0.0171	**0.0152**
ES@99↓	0.0287	0.0276	**0.0228**
Exc.@0.3↓	0.18%	0.15%	**0.09%**

Table 7 shows that the quality of tail supervision directly affects high-error prediction. Compared with in-sample residuals, OOF residuals provide more reliable estimates of unseen prediction errors, leading to improved identification of high-error samples.

Removing OOF-based supervision results in degraded high-error performance, particularly in P99, Tail MAE, and ES@99. This indicates that accurate prediction-error-tail assignment is important for learning waveform-dependent structure associated with large prediction errors, rather than merely fitting residual patterns observed during training.

Overall, the results validate the role of OOF residuals as a reliable supervision mechanism for high-error-aware learning, enabling SeisTailFlow to better handle rare but consequential arrival-time deviations.

E.3.2 GPD-based Exceedance Modeling

Extreme arrival-time errors are rare but can substantially affect downstream seismic analysis. To evaluate whether explicit modeling of high-threshold residual exceedances improves the characterization of such events, we analyze the effect of the GPD-based constraint. The comparison focuses on the high-error exceedance region rather than average prediction accuracy.

Figure 11(a) compares empirical exceedance distributions with the fitted GPD. The GPD-based model provides a closer approximation to the observed high-threshold exceedance behavior, indicating that the proposed constraint captures statistical characteristics of rare large prediction errors more accurately.

The survival curves in Figure 11(b) further demonstrate the difference in high-error behavior. Without GPD regularization, the modeled exceedance behavior deviates more strongly from the empirical high-error region, especially for large residual values. In contrast, the GPD-constrained model better preserves the observed decay pattern of extreme prediction errors.

Figure 11(c) summarizes the impact on extreme-error metrics. Removing the GPD constraint causes limited changes in average accuracy but leads to consistent degradation in P99, Tail MAE, and expected shortfall. This indicates that the primary role of GPD regularization is to constrain high-threshold prediction-error exceedances rather than average prediction behavior.

Overall, these results show that the GPD-based component provides explicit regularization of high-threshold residual exceedances, complementing the Main and Tail Flow architecture by improving reliability in rare but consequential large-error cases.

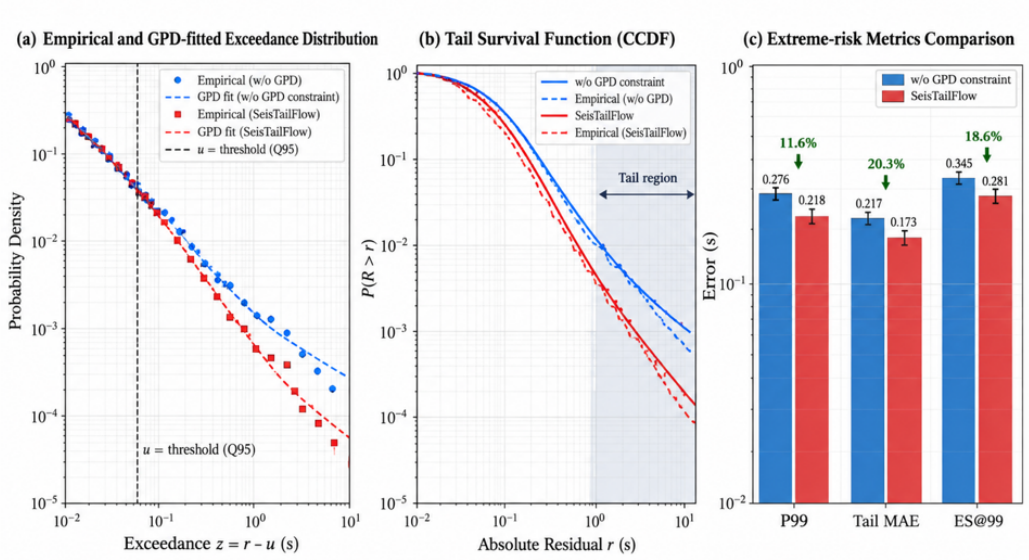


Figure 11: **Effect of GPD-based exceedance modeling.** (a) Empirical high-threshold exceedance distributions and corresponding GPD fits. (b) Survival functions (CCDF) comparing models with and without the GPD constraint. (c) Comparison of extreme-error metrics, including P99, Tail MAE, and expected shortfall. The results show that GPD regularization improves the characterization of rare large prediction errors.

E.3.3 Spectral Conditioning

The time-domain waveform representation may become insufficient when the target phase arrival is obscured by noise or spectrally similar transient signals. To evaluate the contribution of spectral conditioning, we compare the full model with variants that remove frequency-domain information. The analysis focuses on whether spectral cues provide complementary information under challenging waveform conditions.

E.4 Hyperparameter Sensitivity

To examine the robustness of SeisTailFlow with respect to important hyperparameters, we conduct sensitivity analyses on three parameters related to high-error supervision, GPD regularization, and adaptive routing. For each experiment, only one parameter is varied while keeping all other training configurations fixed. We evaluate both overall accuracy and high-error performance to assess whether the model maintains a stable balance between dominant-pattern modeling and high-error control.

Figure 12(a) shows the effect of the prediction-error tail-threshold quantile q. Very low quantile thresholds include too many non-extreme samples in the high-error supervision set, while overly high thresholds reduce the amount of available tail supervision. The model achieves stable performance in an intermediate range, indicating a reasonable trade-off between the amount and specificity of high-error supervision.

Figure 12(b) evaluates the sensitivity to the GPD loss weight. When the weight is too small, the exceedance constraint provides insufficient regularization. Conversely, an excessively large weight may overemphasize extreme errors and disturb the balance with overall prediction accuracy. The stable region indicates that GPD regularization improves high-error behavior without dominating the overall prediction objective.

Figure 12(c) analyzes the routing-related parameter. The performance remains stable within a broad intermediate range, suggesting that adaptive coordination between Main and Tail flows is not sensitive to fine-grained parameter selection. Extremely aggressive routing settings may disturb the

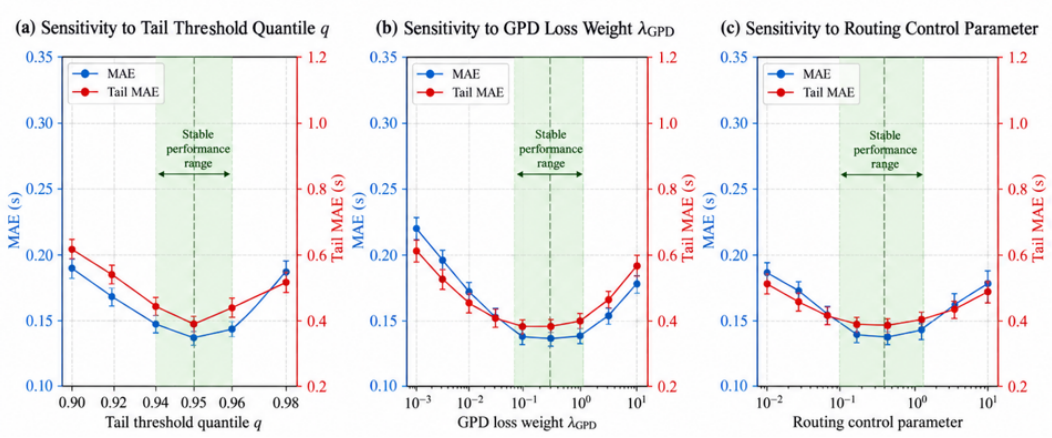


Figure 12: **Hyperparameter sensitivity of SeisTailFlow.** (a) Sensitivity to the tail-threshold quantile q. (b) Sensitivity to the GPD loss weight λ_{GPD}. (c) Sensitivity to the routing control parameter. For each experiment, only one hyperparameter is varied while all other settings remain fixed. Performance is evaluated using both MAE and Tail MAE. The results show that SeisTailFlow remains stable within a reasonable parameter range, while overly weak or aggressive high-error regularization may degrade tail-error performance.

balance between dominant-pattern modeling and high-error-aware correction, resulting in degraded tail-error performance.

Overall, the sensitivity analysis confirms that SeisTailFlow is robust to reasonable variations of its key hyperparameters. The observed trends further support the design principle of coordinating dominant-pattern modeling with explicit high-error-aware supervision and regularization.

F SUPPLEMENTARY DETAILS OF EMPIRICAL EXPERIMENTS

F.1 SUPPLEMENTARY DETAILS OF STEAD

We provide additional details on the data splits, preprocessing, and data governance protocol used in the STEAD experiments. All data partitions and preprocessing rules are determined before evaluation on the final holdout and are applied consistently across all compared methods.

Input and Target. For the i-th record, the model input is the raw three-component seismic waveform $\mathbf{X}_i \in \mathbb{R}^{3\times T}$, with $T = 6000$, where the three channels correspond to the Z, N, and E components, respectively. STEAD waveforms are sampled at $100\,\mathrm{Hz}$, so each record corresponds to a $60\,\mathrm{s}$ observation window. The supervision target y_i is the manually annotated P-wave arrival time in seconds. All compared methods use the same three-component inputs, data partitions, and evaluation samples.

Event-level Data Splits. To prevent correlated waveforms from the same earthquake event from appearing across model-development and evaluation sets, we perform all data splitting at the earthquake-event level according to `event_id`. STEAD chunk2 is partitioned into mutually disjoint training, calibration, and development sets, while chunk3 is reserved as the frozen final holdout. The resulting subsets and their roles are summarized in Table **??**.

The training, calibration, and development splits are mutually disjoint at the event level. The chunk3 holdout is also event-disjoint from the chunk2 data used during model development. Consequently, multiple traces associated with the same earthquake event are not distributed across model-development and final-evaluation sets.

Train-only Normalization. All input-related statistics are estimated exclusively from the chunk2 training split. Specifically, we compute the global mean and standard deviation of each of the three waveform channels from the training data and apply these statistics unchanged to the calibration, development, and holdout sets. We do not perform trace-wise z-score normalization, because doing so would remove relative amplitude differences across records that may contain useful discriminative information. The normalization statistics for the P-wave arrival-time targets are likewise estimated using only the training data, and model outputs are transformed back to the original time scale in seconds for evaluation.

Background-referenced Spectral Conditioning. For spectral conditioning, the first 400 samples of each waveform, corresponding to a $4\,\mathrm{s}$ background window, are used to estimate the three-component background power spectral density (PSD). To ensure that the target phase does not contaminate the background estimate, all records included in the experiments satisfy

$$\mathrm{sample}(y_i) \geq 400.$$

Therefore, the PSD estimation window always lies before the manually annotated P-wave arrival. The STFT representation of the complete three-component waveform is then normalized with respect to the corresponding background PSD to construct the background-referenced spectral representation used by SeisTailFlow.

Event-wise OOF Tail Supervision. Tail supervision is constructed strictly within the training data. We perform five-fold event-wise out-of-fold (OOF) prediction on the chunk2 training split. For each fold, the auxiliary predictor is trained using events from the remaining four folds and generates P-wave arrival-time predictions only for events in the held-out fold. Thus, every training record receives an approximately out-of-sample absolute residual,

$$r_i^{\mathrm{OOF}} = \left| y_i - \hat{y}_i^{\mathrm{OOF}} \right|.$$

The tail threshold is determined from a high quantile of the training-set OOF residual distribution and is then used to construct the tail supervision required by Seis TIS and the Tail Flow. For the frozen training split used in the reported experiments, the 95th-percentile threshold is

$$u_{0.95} = 2.72264\ \mathrm{s},$$

such that approximately 5% of the training records are identified as high-error tail samples. Neither holdout labels nor holdout prediction errors are used to determine the tail threshold or construct training-time tail labels.

Train-defined SNR Stratification. The SNR stratification used for conditional robustness analysis is also defined using only the training data. The SNR-bin boundaries are fixed before final holdout evaluation and are subsequently applied unchanged to the development and holdout sets. Accordingly, all compared methods are evaluated on identical SNR subgroups, and holdout performance does not influence either the subgroup boundaries or the definition of low-SNR samples.

F.2 Tail Error Distribution and Extreme Risk

The main-text evaluation summarizes high-error behavior using P95, P99, Tail MAE, ES@99, and the fixed-threshold exceedance rate Exc.@0.3. To examine whether these improvements extend beyond a small number of discrete operating points, we further analyze the complete empirical error tail using the complementary cumulative distribution function (CCDF) and expected shortfall over multiple quantile levels.

For the absolute P-wave arrival-time error

$$e_i = |\hat{y}_i - y_i|\,, \tag{73}$$

the empirical CCDF at an error threshold δ is defined as

$$\widehat{S}(\delta) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}\,(e_i > \delta)\,. \tag{74}$$

Thus, $\widehat{S}(\delta)$ directly measures the fraction of holdout samples whose arrival-time error exceeds δ. In particular, $\widehat{S}(0.3\,\mathrm{s})$ corresponds to the Exc.@0.3 metric reported in the main text.

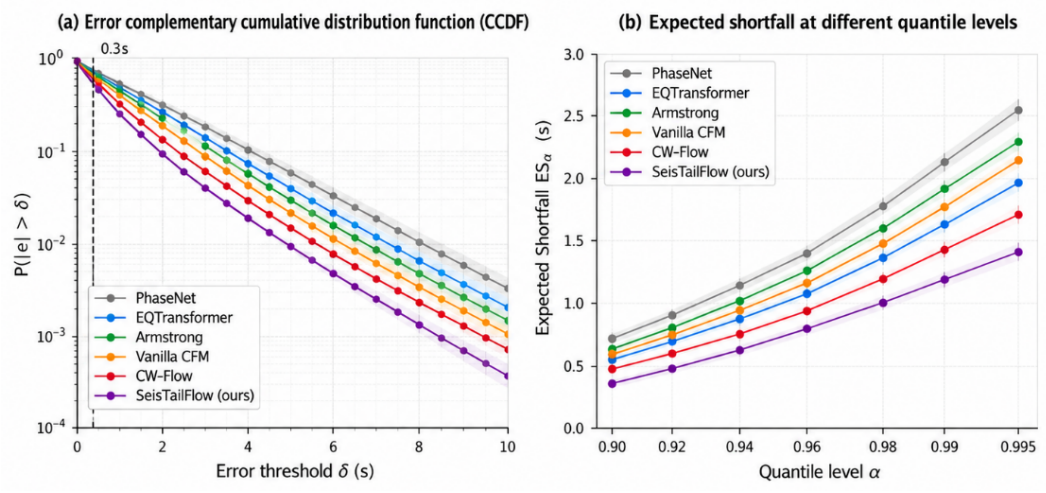


Figure 13: **Tail-error distribution and extreme-risk analysis on STEAD.** (a) Empirical complementary cumulative distribution functions (CCDFs) of absolute P-wave arrival-time errors. The dashed vertical line at $0.3\,\mathrm{s}$ denotes the fixed error threshold used for Exc.@0.3 in the main comparison. (b) Expected shortfall evaluated across increasingly extreme quantile levels; the value at $\alpha = 0.99$ corresponds to ES@99 reported in the main text. Lower CCDF and expected-shortfall values indicate lower frequency and severity of large arrival-time errors, respectively.

As shown in Figure 13(a), differences among the compared methods become increasingly apparent as the error threshold moves into the high-error region. SeisTailFlow maintains a lower exceedance probability over the tail of the absolute-error distribution, consistent with its lower P95, P99, and Exc.@0.3 in the main comparison. The improvement is therefore not confined to isolated quantile values, but is also reflected in the broader decay behavior of the empirical error tail.

We further evaluate the severity of errors after entering increasingly extreme regions of the distribution. For a quantile level $\alpha \in (0, 1)$, let

$$q_\alpha = Q_\alpha\left(\{e_i\}_{i=1}^N\right), \tag{75}$$

where $Q_\alpha(\cdot)$ denotes the empirical α-quantile. Expected shortfall is defined as

$$\mathrm{ES}_\alpha = \frac{\sum_{i=1}^N e_i \, \mathbb{I}(e_i > q_\alpha)}{\sum_{i=1}^N \mathbb{I}(e_i > q_\alpha)}. \tag{76}$$

Whereas q_α determines the boundary of a high-error region, ES_α measures the average error magnitude conditional on exceeding that boundary.

Figure 13(b) shows that the conditional error severity increases for all methods as α approaches the extreme upper tail. SeisTailFlow maintains lower expected shortfall across these increasingly stringent operating points, with the value at $\alpha = 0.99$ corresponding to the ES@99 reported in the main text. Together, the CCDF and expected-shortfall analyses show that the improvement of SeisTailFlow is reflected in both a reduced probability of large errors and a reduced severity conditional on such errors occurring.

Importantly, the tail quantities analyzed in this section are computed only for evaluating the final predictions on the frozen holdout. They do not redefine the training-time tail labels. As described in Appendix F.1, Seis TIS tail supervision is constructed exclusively from event-wise OOF residuals within the training split, and no holdout prediction error is used to define the training tail.

F.3 Calibration and Coverage-Matched Sharpness

A reliable probabilistic phase picker should produce uncertainty estimates that are both well calibrated and informative. High empirical coverage alone is insufficient, because overly conservative predictive distributions may achieve large coverage simply by producing excessively wide prediction intervals. We therefore jointly evaluate calibration and coverage-matched sharpness.

For a nominal coverage level $1 - \alpha$, let the central predictive interval for the i-th sample be

$$I_i^{(1-\alpha)} = \left[L_i^{(1-\alpha)}, U_i^{(1-\alpha)}\right]. \tag{77}$$

The corresponding empirical coverage is

$$\widehat{C}_{1-\alpha} = \frac{1}{N} \sum_{i=1}^N \mathbb{I}\left(y_i \in I_i^{(1-\alpha)}\right), \tag{78}$$

and the coverage error is

$$\mathrm{CE}_{1-\alpha} = \left|\widehat{C}_{1-\alpha} - (1 - \alpha)\right|. \tag{79}$$

An ideally calibrated model therefore follows $\widehat{C}_{1-\alpha} = 1 - \alpha$.

The mean interval width at the same nominal level is

$$\mathrm{Width}_{1-\alpha} = \frac{1}{N} \sum_{i=1}^N \left(U_i^{(1-\alpha)} - L_i^{(1-\alpha)}\right). \tag{80}$$

Because interval width is meaningful only relative to the coverage it achieves, we compare sharpness at matched empirical coverage. For model m and a target empirical coverage c, we define

$$\mathrm{CMS}_m(c) = \mathrm{Width}_m \mid \widehat{C}_m \approx c, \tag{81}$$

where a smaller value indicates a sharper predictive distribution at the same realized coverage. In the main comparison, the coverage-matched width at approximately 95% empirical coverage is reported as W@95.

Figure 14(a) compares reliability across multiple coverage levels. SeisTailFlow remains close to the ideal calibration diagonal and achieves the lowest CE@95 among the probabilistic methods, with $\mathrm{CE@95} = 0.0012$. The corresponding values for Armstrong, Vanilla CFM, and CW-Flow are

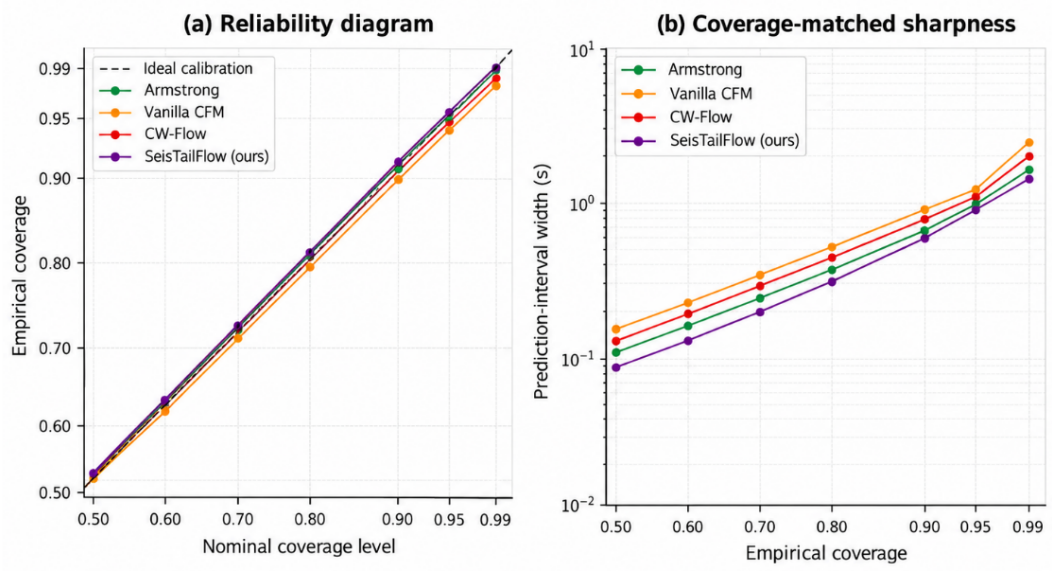


Figure 14: **Calibration and coverage-matched sharpness on STEAD.** (a) Reliability diagrams comparing nominal and empirical coverage for the probabilistic methods; the diagonal denotes ideal calibration. (b) Mean prediction-interval width compared at matched empirical coverage. Better probabilistic predictions remain close to the calibration diagonal while maintaining narrower intervals at the same realized coverage.

0.0018, 0.0035, and 0.0029, respectively. Together with its lower CRPS of 0.255, this indicates that the conditional predictive distribution of SeisTailFlow more closely matches the realized uncertainty in P-wave arrival times.

Calibration alone, however, does not determine whether the predictive distribution is informative. Figure 14(b) therefore compares prediction-interval widths at matched empirical coverage. At approximately 95% empirical coverage, SeisTailFlow achieves W@95 = 1.071 s, compared with 1.103 s for Armstrong, 1.188 s for Vanilla CFM, and 1.124 s for CW-Flow. Thus, its improved coverage reliability is not obtained by simply widening the predictive intervals.

Taken together, the reliability and coverage-matched sharpness analyses show that SeisTailFlow provides a more favorable calibration–sharpness trade-off: its predictive intervals better match their intended coverage while remaining more concentrated at comparable realized coverage.

F.4 Conditional Performance across SNR

To examine how phase-picking performance varies with signal quality, we further evaluate all methods across the SNR strata defined in Appendix F.1. The subgroup boundaries are determined before final holdout evaluation and are applied identically to all compared methods, so the analysis is independent of model predictions.

For the k-th SNR subgroup $\mathcal{B}_k^{\mathrm{SNR}}$, we compute

$$\mathrm{MAE}_k^{\mathrm{SNR}} = \frac{1}{|\mathcal{B}_k^{\mathrm{SNR}}|} \sum_{i \in \mathcal{B}_k^{\mathrm{SNR}}} |\hat{y}_i - y_i| \,. \tag{82}$$

To quantify how the advantage of SeisTailFlow changes with signal quality, we further report its relative MAE reduction with respect to EQTransformer, which is the strongest baseline under the Low-SNR MAE metric in the main comparison:

$$\mathrm{RER}_k = \frac{\mathrm{MAE}_k^{\mathrm{EQTransformer}} - \mathrm{MAE}_k^{\mathrm{SeisTailFlow}}}{\mathrm{MAE}_k^{\mathrm{EQTransformer}}} \times 100\%. \tag{83}$$

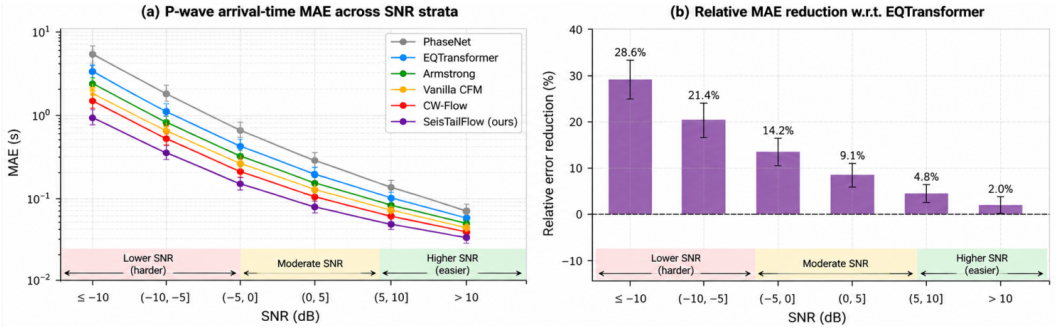


Figure 15: **Conditional performance across SNR levels on STEAD.** (a) P-wave arrival-time MAE across train-defined SNR strata for all compared methods. (b) Relative MAE reduction of SeisTailFlow with respect to the fixed EQTransformer reference across the same strata. Larger relative reduction indicates a greater advantage over the reference method under the corresponding signal-quality condition.

Figure 15(a) shows that arrival-time errors systematically increase as SNR decreases across all methods, indicating that weak signal quality is associated with more difficult P-wave picking. The separation among methods also becomes more apparent in the lower-SNR strata, whereas their performance is more similar when the phase arrival is observed under higher-SNR conditions.

Figure 15(b) makes this conditional behavior more explicit. Relative to the fixed EQTransformer reference, the error reduction achieved by SeisTailFlow is larger in the lower-SNR groups and progressively decreases as signal quality improves. This pattern is consistent with the main-text Low-SNR MAE results: SeisTailFlow achieves a Low MAE of $0.463\,\mathrm{s}$, compared with $0.501\,\mathrm{s}$ for EQTransformer, corresponding to an aggregate reduction of approximately 7.6% under the reported low-SNR evaluation.

These results show that the gain of SeisTailFlow is not distributed uniformly across the holdout set. Instead, its advantage is more pronounced when the P-wave arrival is less distinguishable from the surrounding waveform fluctuations, supporting the role of tail-aware and background-referenced spectral modeling under noisy observation conditions.

F.5 Robustness across Random Seeds

The STEAD results reported in the main text are averaged over three independent training runs with different random seeds while keeping the data splits, preprocessing statistics, OOF tail definition, model configuration, and evaluation protocol fixed.

As reported in Table 2 of the main text, the standard deviations remain small for both overall and tail-sensitive metrics. For SeisTailFlow, the standard deviations are 0.008 for MAE, 0.019 for P95, 0.034 for P99, 0.027 for Tail MAE, 0.041 for ES@99, and 0.005 for CRPS. These results indicate that the observed improvements are stable across independent training runs and are not driven by a single favorable initialization.

F.6 Additional Waveform Case Studies

The main text presents a representative low-SNR event to illustrate the behavior of different methods when the P-wave onset is weakly distinguishable from surrounding waveform fluctuations. To complement this example, we further present representative records spanning different waveform conditions and examine how prediction behavior changes as the phase information becomes less distinct.

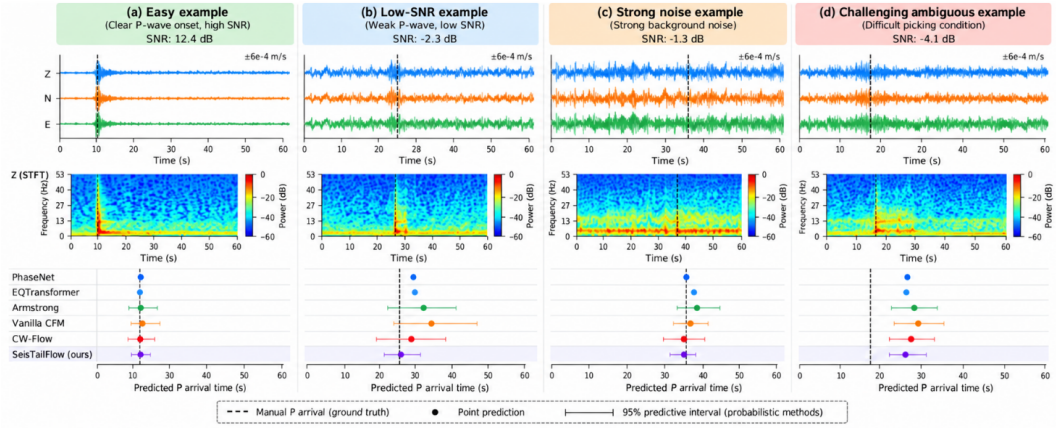


Figure 16: **Additional representative waveform cases on STEAD.** Four representative examples are shown: (a) an easy case with a clear P-wave onset and high SNR; (b) a low-SNR case with a weak P-wave signal; (c) a noisy case with strong background interference; and (d) a challenging ambiguous case in which the P-wave onset is difficult to distinguish from surrounding waveform structure. For each example, we show the three-component waveform, the time–frequency representation of the Z component, and the P-wave arrival-time predictions produced by the compared methods. The black dashed line denotes the manually annotated P-wave arrival, markers denote point predictions, and horizontal intervals indicate 95% predictive intervals for methods that provide probabilistic predictions.

For the relatively easy example in Figure 16(a), the compared methods produce similar arrival-time estimates around the manually annotated P-wave onset. SeisTailFlow remains accurate under this regular high-SNR condition, consistent with the overall picking performance reported in the main text.

The differences among methods become more apparent in Figures 16(b)–(d). When the P-wave signal is weak, background interference is strong, or the local waveform structure is ambiguous, several baseline predictions exhibit larger deviations from the manual annotation, whereas SeisTailFlow remains closer to the annotated arrival in these representative cases. The corresponding time–frequency representations further illustrate that waveforms with weak or ambiguous temporal onset cues may still exhibit phase-related energy patterns in the spectral domain, providing complementary information for arrival-time prediction.

These qualitative examples are consistent with the aggregate analyses in Figures 13 and 15. The SNR-stratified results show that the performance advantage becomes more pronounced as signal quality deteriorates, while the tail-distribution analysis shows a broader reduction in large arrival-time errors. Figure 16 therefore provides an intuitive waveform-level illustration of the conditional and tail-related trends observed in the quantitative evaluation.