Accelerating Posterior Inference from Pulsar Light Curves via Learned Latent Representations and Local Simulator-Guided Optimization

Farhana Taiyebah [1] **Abu Bucker Siddik** [1] **Indronil Bhattacharjee** [1] **Diane Oyen** [1] **Soumi De** [1] **Greg Olmschenk** [2] **Constantinos Kalapotharakos** [2]

Abstract

Posterior inference from pulsar light curves is commonly performed using Markov chain Monte Carlo (MCMC), which is accurate but computationally expensive. We introduce a framework that accelerates inference while maintaining accuracy by combining learned latent representations with simulator-guided optimization. A masked U-Net is pretrained to reconstruct complete light curves from partial observations and to produce informative latent embeddings. For a query light curve, we retrieve similar simulations from a precomputed bank using distances in this embedding space, yielding an initial empirical approximation to the posterior over parameters. This estimate is then refined via local hill-climbing updates guided by a forward simulator, progressively concentrating on higher-likelihood regions. Experiments on the observed light curve of PSR J0030+0451, captured by NASA's Neutron Star Interior Composition Explorer (NICER), show that our method recovers key posterior support and multimodal structure while remaining broadly consistent with MCMC, achieving a $120\times$ speedup and reducing inference time from 24 hours to 12 minutes.

1. Introduction

Pulsars are rapidly rotating neutron stars that emit beams of electromagnetic radiation, observed as periodic pulses when aligned with the Earth (Sturrock, 1971; Smith, 1977). Characterizing pulsar properties–such as mass, radius, surface temperature, and magnetic field geometry (Lattimer & Prakash, 2016; Özel & Freire, 2016)–requires solving an inverse problem: inferring the posterior distribution over physical parameters given observed light curves. Traditional Markov chain Monte Carlo (MCMC) sampling (Metropolis et al., 1953; Hastings, 1970; Gilks et al., 1995) provides accurate posterior estimates (Kalapotharakos et al., 2021) but is computationally very expensive, often requiring days to weeks on thousands of CPUs due to the high-dimensional parameter space and parameter degeneracies that create complex, multimodal posterior landscapes. Olmschenk et al. (2025) recently developed a convolutional neural network simulator that replaces the expensive physical simulator within MCMC for pulsar posterior estimation, achieving a $> 400\times$ speedup and enabling convergence in approximately one day on 4000 CPUs–a task that would otherwise require over a year. However, even this accelerated approach still requires substantial computation time and remains constrained by the inherently iterative nature of MCMC sampling, which explores the parameter space through many local steps to adequately characterize strongly multimodal posteriors. Siddik et al. (2025) developed a Transformer–LSTM–based model to estimate degenerate pulsar parameters from light curves with high posterior likelihood. However, this approach does not estimate the full posterior distribution and does not guarantee coverage of all high–posterior-likelihood regions in the parameter space.

We introduce a four-stage inference framework that bypasses MCMC while preserving multimodal and degenerate posterior structure. Given an observed light curve, the method retrieves similar simulated light curves from a simulation bank using a learned embedding space. The parameters associated with the retrieved light curves define an empirical posterior approximation, which is subsequently refined via a simulator-guided hill-climbing procedure. This local optimization sharpens posterior mass within each region of support without collapsing distinct modes or erasing parameter degeneracies captured during the retrieval stage. While this method does not aim to generate exact posterior samples, it offers a computationally efficient approximation that preserves key posterior characteristics, such as multimodality and parameter degeneracies, and demonstrates meaningful agreement with MCMC in settings where full Bayesian sampling is infeasible.

[1]Los Alamos National Laboratory, Los Alamos, NM, USA [2]NASA Goddard Space Flight Center, Greenbelt, MD, USA. Correspondence to: Farhana Taiyebah <taiyebahfarhana@gmail.com>.

Proceedings of the AI4Physics Workshop at the 43rd International Conference on Machine Learning (AI4Physics@ICML 2026), Seoul, South Korea. 2026. Copyright 2026 by the author(s).

The contributions of our work are the following: (**1**) a masked U-Net architecture that learns multi-scale latent representations of time-series data through self-supervised masked reconstruction, enabling similarity-based posterior approximation via retrieval in latent embedding space; (**2**) a hill-climbing optimization procedure that refines retrieved parameter sets through adaptive local search guided by the forward simulator, preserving multimodality while increasing posterior density; (**3**) experimental validation on PSR J0030+0451 demonstrating $120\times$ speedup over traditional MCMC while recovering posterior support and multimodal structure, with results consistent with MCMC; (**4**) demonstration of the proposed approach on a physics inference problem, specifically a gamma-ray spectrum (GRS) inference task (Mo-Ghani, 2025); and (**5**) ablation studies establishing that multi-level embeddings outperform both single-level embeddings and direct cosine similarity on raw light curves for the retrieval task (Appendix C).

2. Methods

Let $\mathcal{D} = \{(\mathbf{x}_i, \boldsymbol{\theta}_i)\}_{i=1}^N$ denote a simulation bank of N light curves $\mathbf{x}_i \in \mathbb{R}^T$ and their parameters $\boldsymbol{\theta}_i \in \mathbb{R}^P$ (e.g. for pulsar parameter estimation, $T = 64$ phase bins and $P = 11$ parameters).

Problem Statement. Our goal is to infer the posterior distribution, $p(\boldsymbol{\theta}|\mathbf{x}^*)$, over physical parameters given an observed light curve $\mathbf{x}^*$. The proposed workflow for posterior prediction (see Figure 1) consists of:

(a) **Pretrain masked U-Net.** Train a 1D U-Net adapted from (Ronneberger et al., 2015) to reconstruct masked light curves, yielding an encoder that provides meaningful multi-scale features for retrieval.

(b) **Build an embedding database.** For each simulated light curve $\mathbf{x}_i$ in the simulation bank, compute a *multi-level* embedding by passing the light curve through the frozen encoder of the pretrained U-Net.

(c) **Retrieve neighbors for a query observation.** For an observed light curve $\mathbf{x}^*$, we compute its embedding using the frozen encoder and retrieve the top-K neighbors in embedding space; the corresponding parameters $\{\boldsymbol{\theta}_k\}$ provide an initial empirical approximation to the posterior.

(d) **Hill-climbing refinement.** Starting from the retrieved parameters $\{\boldsymbol{\theta}_k\}$, minimize the Poisson negative log-likelihood using an adaptive P-dimensional sign-search hill-climbing procedure. At each iteration, we evaluate the $2^P \pm$ perturbation patterns, accept the best improvement, and adapt the step size until a target negative log-likelihood is reached or progress plateaus.

3. Motivation and Rationale for Retrieval and Local Refinement in Posterior Estimation

Our approach decomposes posterior inference into two complementary stages: a global retrieval stage that constructs an empirical approximation of the posterior (Figure 1(a–c)), followed by a simulator-guided local refinement stage (Figure 1(d)) that sharpens the posterior density locally.

Let $\boldsymbol{\theta} \in \mathbb{R}^P$ denote the P physical parameters of an observation, and let $\mathbf{x}^*$ be an observation. Our goal is to characterize the posterior distribution

$$p(\boldsymbol{\theta} \mid \mathbf{x}^*) \propto p(\boldsymbol{\theta})\, p(\mathbf{x}^* \mid \boldsymbol{\theta}), \tag{1}$$

which is challenging due to parameter degeneracies in the observation: distinct parameter configurations can generate nearly indistinguishable observations. As a result, the posterior distribution is spread over extended, possibly disconnected regions of parameter space rather than concentrated near isolated modes. While Markov chain Monte Carlo (MCMC) methods can, in principle, explore such posteriors accurately, their computational cost can range from days to months for a single observation, particularly when the posterior geometry is highly non-trivial (Olmschenk et al., 2025).

Retrieval approximates the global posterior support Let $f_\phi : \mathcal{X} \to \mathbb{R}^m$ be a learned encoder of a masked U-Net model trained via masked observation reconstruction, where $\mathcal{X}$ denotes the space of observations and $\mathbb{R}^m$ an m-dimensional latent embedding space. For any observation $\mathbf{x} \in \mathcal{X}$, we define its embedding as $\mathbf{z} = f_\phi(\mathbf{x})$.

Given a simulation bank $\{(\boldsymbol{\theta}_i, \mathbf{x}_i)\}_{i=1}^N$ of size N, where $\boldsymbol{\theta}_i$ denotes a parameter configuration and $\mathbf{x}_i$ the corresponding light curve simulated with that parameter configuration, we first embed each simulated light curve as $\mathbf{z}_i = f_\phi(\mathbf{x}_i)$. For an observed light curve $\mathbf{x}^*$ with embedding $\mathbf{z}^* = f_\phi(\mathbf{x}^*)$, retrieval is performed in the latent space by selecting the K simulated light curves whose embeddings $\{\mathbf{z}_i\}$ are closest to $\mathbf{z}^*$ under a chosen similarity metric. The corresponding parameter configurations $\{\boldsymbol{\theta}_i\}$ are then retained, inducing the empirical distribution

$$\hat{p}_0(\boldsymbol{\theta} \mid \mathbf{x}^*) \;=\; \frac{1}{K} \sum_{i \in \mathcal{I}_K(\mathbf{x}^*)} \delta(\boldsymbol{\theta} - \boldsymbol{\theta}_i), \tag{2}$$

where $\mathcal{I}_K(\mathbf{x}^*)$ denotes the index set of the retrieved simulations, K is the number of nearest-neighbor simulations retained, and $\delta(\cdot)$ denotes the Dirac delta distribution (i.e., a point mass located at $\boldsymbol{\theta}_i$). This distribution forms a uniform mixture over parameter values whose simulated light curves are most similar to the query observation $\mathbf{x}^*$ under the learned representation. Thus, $\hat{p}_0(\boldsymbol{\theta} \mid \mathbf{x}^*)$ approximates regions of parameter space supported by the observation rather

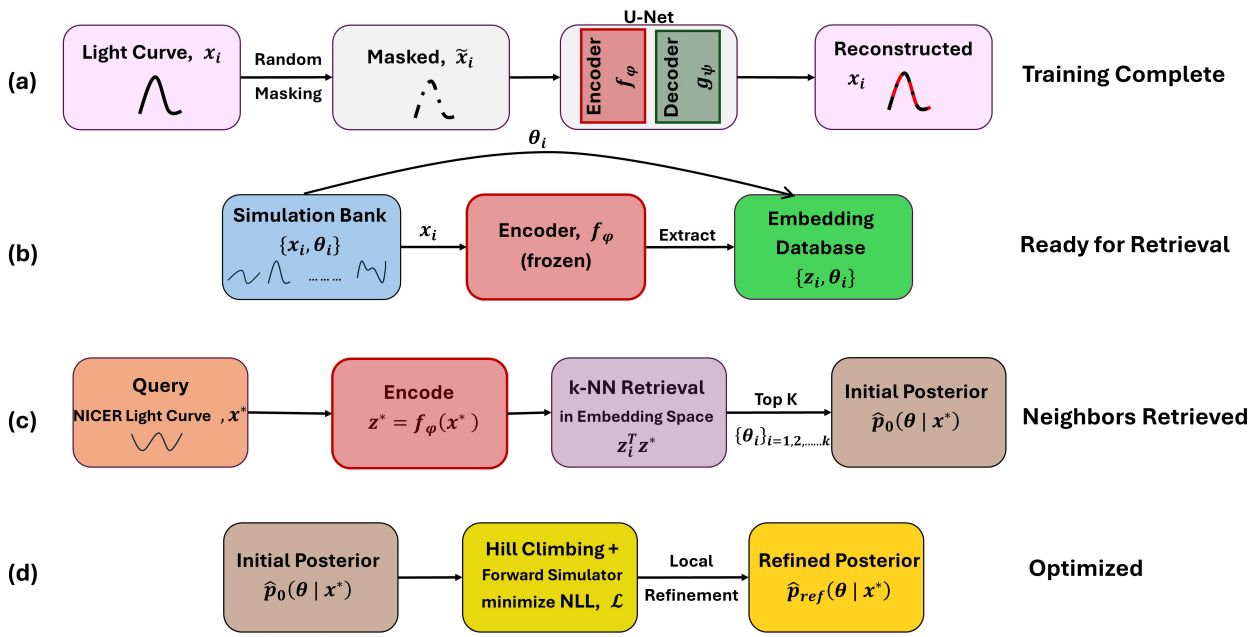

Figure 1. Method overview for accelerated posterior inference from pulsar light curves. (a) A masked 1D U-Net is pretrained to reconstruct partially masked pulsar light curves, learning multi-scale latent representations. (b) The pretrained encoder is used to embed all simulated light curves in the database. (c) For a query observation, nearest-neighbor retrieval in the learned embedding space identifies similar simulated light curves, whose associated parameters form an initial empirical approximation to the posterior. (d) Local hill-climbing refinement guided by the forward simulator improves likelihood locally while preserving multimodal structure and parameter degeneracies captured during retrieval.

than a calibrated posterior density. This step is conceptually aligned with approximate Bayesian computation, where inference is performed using informative low-dimensional summary statistics (Biau et al., 2015). Crucially, because retrieval operates globally over a simulation bank, it naturally preserves parameter degeneracies in the empirical posterior by retaining multiple, distinct configurations that explain the data equally well.

Local refinement sharpens the posterior mass While retrieval identifies plausible regions of parameter space, the retrieved samples are not guaranteed to be locally optimal with respect to the likelihood of the observed light curve. Unlike sampling-based approaches such as MCMC, which rely on rejecting and resampling candidates, our method improves the fidelity of each retrieved sample. Specifically, we apply local hill climbing initialized from each $\boldsymbol{\theta}_k \in \mathcal{I}_K(\mathbf{x}^*)$ by minimizing a Poisson negative log-likelihood

$$\mathcal{L}_{\text{Poisson}}(\boldsymbol{\theta}_k) = \frac{1}{T}\sum_{t=1}^{T}\Big(F_t(\boldsymbol{\theta}_k) - x_t^* \log\big(F_t(\boldsymbol{\theta}_k)+\varepsilon\big)\Big) \quad (3)$$

where $\varepsilon > 0$ is a small pseudocount added inside the logarithm to avoid numerical instability at zero counts. $\mathcal{L}_{\text{Poisson}}$ is the mean Poisson objective between the simulator output $F(\boldsymbol{\theta}_k)$ and the query observation $\mathbf{x}^*$ and T is the number of phase bins in the observation.

Updates are proposed within a restricted neighborhood—initially modifying single parameters and subsequently allowing larger parameter subsets—and are accepted only if they reduce $\mathcal{L}_{\text{Poisson}}(\boldsymbol{\theta}_k)$. This produces a sequence $\{\boldsymbol{\theta}_k^{(n)}\}$ satisfying

$$\mathcal{L}_{\text{Poisson}}(\boldsymbol{\theta}_k^{(n+1)}) \leq \mathcal{L}_{\text{Poisson}}(\boldsymbol{\theta}_k^{(n)}), \quad (4)$$

which implies a monotonic non-decrease in the corresponding likelihood $p(\mathbf{x}^* \mid \boldsymbol{\theta}_k^{(n)})$.

Importantly, this refinement remains strictly local: each trajectory improves the fit within the basin of attraction defined by its retrieved initialization. As a result, local optimization sharpens posterior mass within each region of support without collapsing distinct modes or erasing parameter degeneracies captured by the retrieval stage.

Because the embeddings of simulated observations in the simulation bank are computed offline for the retrieval task, and because local refinement guided by the forward simulator is computationally efficient, the proposed retrieval-based approach with local refinement is comparatively tractable, often completing within minutes, relative to traditional MCMC methods, which may require days of computation using large-scale CPU resources to generate posteriors for a single observation. While the method does not aim to produce exact posterior samples or offer asymptotic guar-

antees, it provides a practical approximation that retains multimodality and parameter degeneracies in settings where exhaustive Bayesian sampling is computationally challenging. Importantly, the refined distribution should not be interpreted as a fully calibrated posterior density, but rather as a fast approximation to posterior support that can be used for rapid analysis or to initialize downstream sampling methods.

4. Masked U-Net for Light Curve Representation Learning

We pretrain a 1D U-Net autoencoder adapted from (Ronneberger et al., 2015) to learn multi-scale representations of pulsar light curves through masked reconstruction. The encoder produces latent features at multiple resolutions, which we subsequently leverage for similarity-based retrieval. More details about the U-Net architecture are presented in Appendix A.

We set the base channel width `base_filters` $= 64$, which determines the number of filters at each encoder level following a doubling pattern: $64 \to 128 \to 256 \to 512$ channels across the three encoder blocks and bottleneck.

Multi-scale feature hierarchy. The encoder progressively expands channel capacity ($1 \to 64 \to 128 \to 256 \to 512$ for `base_filters` $= 64$) while reducing temporal resolution ($64 \to 32 \to 16 \to 8$), yielding representations with increasingly large receptive fields. Because each encoder block contains two convolutional layers and each downsampling step doubles the effective stride, the receptive field grows rapidly with depth. For the architecture used here (three encoder blocks followed by a two-layer bottleneck), the effective receptive field at the bottleneck spans approximately 68 phase bins, exceeding the input length of $T = 64$. As a result, bottleneck features integrate information from essentially the entire light curve, capturing global pulse-profile structure, while intermediate features retain higher temporal resolution and encode more localized morphology.

4.1. Masked Reconstruction Pretraining

Random masking. For each simulated light curve used in training, $\mathbf{x}$, we randomly mask a fraction $\rho = 0.75$ of phase bins. Given normalized input $\tilde{\mathbf{x}} \in \mathbb{R}^T$, we sample a random permutation π of indices $\{1, \ldots, T\}$ and retain only $T_{\text{keep}} = \lfloor (1-\rho)T \rfloor = 16$ positions. The binary mask $\mathbf{m} \in \{0,1\}^T$ is:

$$m_t = \begin{cases} 0 & \text{if } t \in \{\pi(1), \ldots, \pi(T_{\text{keep}})\} \\ 1 & \text{otherwise} \end{cases}. \tag{5}$$

Masked positions are zeroed in the input:

$$\tilde{\mathbf{x}}^{\text{mask}} = (1 - \mathbf{m}) \odot \tilde{\mathbf{x}}. \tag{6}$$

Masking ratio. During training, we apply per-sample random masking with a mask fraction of 0.75, so that 75% of phase bins are masked and reconstructed. During inference and embedding extraction, we disable masking by setting `mask_ratio` $= 0.0$, ensuring that the encoder processes the complete light curve when extracting bottleneck and multi-level latent representations.

Reconstruction objective. The U-Net decoder g_ψ reconstructs the *original* unmasked input from the bottleneck and skip connections. Let $\hat{\mathbf{x}} = g_\psi(f_\phi(\tilde{\mathbf{x}}^{\text{mask}}))$ denote the reconstruction. The loss is computed *only on masked positions*:

$$\mathcal{L}(\phi, \psi) = \mathbb{E}_{(\mathbf{x},\mathbf{m})\sim\mathcal{D}} \left[\frac{\sum_{t=1}^{T} m_t (\hat{x}_t - \tilde{x}_t)^2}{\sum_{t=1}^{T} m_t} \right], \tag{7}$$

where the expectation is over light curves $\mathbf{x}$ from training distribution $\mathcal{D}$ and random masks $\mathbf{m}$. By penalizing reconstruction error only where the model lacks direct observations, this objective encourages the encoder to learn contextual representations that capture phase-resolved structure necessary for inferring missing flux values.

Training procedure. We pretrain on $N = 10M$ simulated light curves using AdamW with learning rate $\eta_0 = 10^{-5}$, linearly decayed to zero over 200 epochs.

5. Embedding Construction and Retrieval

After pretraining, the encoder f_ϕ produces multi-scale features from a light curve that can be aggregated into fixed-dimensional embeddings for similarity-based retrieval. We compare two aggregation strategies: single-level (bottleneck only) and multi-level (intermediate + bottleneck).

5.1. Single-Level Embeddings

The bottleneck representation $\mathbf{b} \in \mathbb{R}^{512\times 8}$ from Eq. (15) (Appendix) captures global context through its large receptive field. We obtain a fixed-length embedding via temporal pooling (Lin et al., 2013):

$$\mathbf{z}^{\text{single}} = \text{GAP}(\mathbf{b}) = \frac{1}{8} \sum_{t=1}^{8} \mathbf{b}_{:,t} \in \mathbb{R}^{512}, \tag{8}$$

where GAP denotes global average pooling over the temporal dimension. This produces a permutation-invariant descriptor emphasizing coarse periodic structure.

5.2. Multi-Level Embeddings

Single-level embeddings primarily encode global shape through deep features with large receptive fields. To additionally capture local pulse morphology, we fuse intermediate features with the bottleneck.

Intermediate feature extraction. At encoder level 3, the feature map $\mathbf{e}^{(3)} \in \mathbb{R}^{256\times 16}$ maintains higher temporal resolution and smaller receptive fields than the bottleneck. We extract this via forward hooks during inference and apply temporal pooling:

$$\mathbf{f}^{(3)} = \mathrm{GAP}(\mathbf{e}^{(3)}) \in \mathbb{R}^{256}. \tag{9}$$

Hierarchical concatenation. We concatenate the pooled intermediate features with the single-level embedding:

$$\mathbf{z}_{\text{raw}}^{\text{multi}} = [\mathbf{f}^{(3)} \oplus \mathbf{z}^{\text{single}}] \in \mathbb{R}^{768}, \tag{10}$$

where $\oplus$ denotes concatenation. This parameter-free fusion preserves information from both hierarchical levels without requiring additional training.

Normalization for cosine retrieval. All embeddings are ℓ_2-normalized to unit length:

$$\mathbf{z} = \frac{\mathbf{z}_{\text{raw}}}{\|\mathbf{z}_{\text{raw}}\|_2 + \epsilon}, \quad \epsilon = 10^{-8}. \tag{11}$$

Under this normalization, cosine similarity equals inner product: $\cos(\mathbf{z}_i, \mathbf{z}_j) = \mathbf{z}_i^\top \mathbf{z}_j$, enabling efficient retrieval via matrix multiplication.

5.3. k-Nearest Neighbor (k-NN) Retrieval

Given a query light curve $\mathbf{x}^*$, we use two-stage retrieval. First, we compute cosine similarity in embedding space:

$$s_i = \mathbf{z}^{*\top} \mathbf{z}_i, \quad \text{where } \mathbf{z}^* = f_\phi(\mathbf{x}^*), \tag{12}$$

and retrieve the top 20,000 most similar light curves from the database. We intentionally over-retrieve at this stage to preserve broad posterior support and avoid prematurely discarding potentially relevant modes. Second, we re-rank these 20,000 candidates using the MdNSE metric (Olmschenk et al., 2025)—a scale-invariant measure in unnormalized signal space (Appendix B)—selecting the final $K = 4{,}000$ neighbors with lowest reconstruction error. The choice of $K = 4{,}000$ is empirically justified in Section 7.3 through posterior convergence analysis.

This hybrid approach balances efficiency (cosine similarity over 10M embeddings) with accuracy (MdNSE refinement in physical flux space). The parameters corresponding to the top-K retrieved light curves form the initial empirical posterior $\hat{p}_0(\boldsymbol{\theta} \mid \mathbf{x}^*)$ as described in Section 3.

6. Hill Climbing Optimization for Refined Posterior Prediction

We refine the retrieval-initialized empirical posterior using a forward model (Olmschenk et al., 2025)–guided local search (Tovey, 1985) that climbs toward higher likelihood regions while remaining simulator-consistent. The refinement operates in parameter space $\boldsymbol{\theta} \in \mathbb{R}^{11}$ and on light-curve outputs of length $T = 64$.

6.1. Objective functions

Given target (observed) light curve $\mathbf{x}^* \in \mathbb{R}^{64}$ and simulator $F(\boldsymbol{\theta})$, we optimize a Poisson negative log-likelihood

$$\mathcal{L}_{\text{Poisson}}(\boldsymbol{\theta}) = \frac{1}{T}\sum_{t=1}^{T}\Big(F_t(\boldsymbol{\theta}) - x_t^* \log\big(F_t(\boldsymbol{\theta}) + \varepsilon\big)\Big) \tag{13}$$

where $\varepsilon > 0$, $\mathcal{L}_{\text{Poisson}}$ is the mean Poisson loss between the simulator output $F(\theta)$ and the observed light curve $\mathbf{x}^*$ and T is the number of phase bins in the light curve.

6.2. From 1-D to P-D hill climbing

A simple baseline updates one parameter at a time: at each step it proposes $\boldsymbol{\theta}_j \pm \eta$ for a randomly chosen j and accepts the better move. This is robust but can be slow when parameters are correlated (Wright, 1996; Kolda et al., 2003).

We generalize to block hill climbing over P parameters chosen adaptively during each full cycle of the block hill-climb update. For a chosen index set $I = \{i_1, i_2, \ldots, i_P\}$, we evaluate all 2^P signed moves of magnitude η along those axes.

Let the step magnitudes be $\Delta = \big(\Delta_1, \ldots, \Delta_P\big) \in \mathbb{R}_+^P$. We enumerate all 2^P sign patterns

$$\sigma \in \{-1, +1\}^P,$$

and form the candidate set, we pick the best joint move by discrete search:

$$\sigma^\star \;=\; \arg\min_{\sigma\in\{-1,+1\}^P} \mathcal{L}\Big(\boldsymbol{\theta}^{(\sigma)}\Big), \quad \boldsymbol{\theta}^+ \;=\; \boldsymbol{\theta}^{(\sigma^\star)}.$$

6.3. Adaptive step control

After each block, update the step magnitudes by

$$\Delta \leftarrow \begin{cases} \gamma_{\text{grow}}\,\Delta, & \text{if } \mathcal{L}(\boldsymbol{\theta}^+) < \mathcal{L}(\boldsymbol{\theta}) - \varepsilon, \\ \gamma_{\text{shrink}}\,\Delta, & \text{otherwise,} \end{cases}$$

with $\gamma_{\text{grow}} > 1$, $\gamma_{\text{shrink}} \in (0, 1)$, and tolerance $\varepsilon > 0$. Optionally stop when no improvement occurs for p consecutive blocks (plateau) or when $\min_j \Delta_j < \Delta_{\min}$.

6.4. Block Refinement

At the core is a repeated block refinement loop. Given a seed $\boldsymbol{\theta}^{(0)}$, we perform a sequence of blocks; each block

executes N_{trial} inner trial steps. In a trial, we randomly select an index set

$$\tilde{\boldsymbol{\theta}}_i \leftarrow \boldsymbol{\theta}_i + s\,\sigma_i, \quad i \in I, \quad \sigma_i \in \{-1, +1\},$$

with clamping to parameter bounds. We accept the proposal if it reduces the negative log-likelihood $\mathcal{L}$. Within each block we anneal the step size s if the acceptance rate falls below a target band.

Stopping. We maintain the best value per seed, $\mathcal{L}^*$, the block gain Δb (improvement after block b), and a patience counter. Refinement stops when (i) $\Delta b < \epsilon$ for N_{plateau} consecutive blocks (plateau), or (ii) a target objective τ is reached (e.g., the best refined baseline), or (iii) a maximum number of blocks is exceeded.

7. Experiments and Results

We evaluate our framework on PSR J0030+0451, a millisecond pulsar observed by NICER (Gendreau et al., 2016), using a 10M light curve simulation database (Olmschenk et al., 2025). Each simulation is parameterized by an 11-dimensional vector $\boldsymbol{\theta} = (x_D, y_D, z_D, \alpha_D, \phi_D, \quad x_Q, y_Q, z_Q, \alpha_Q, \phi_Q, \quad B_Q/B_D)$, where (x_D, y_D, z_D) and (x_Q, y_Q, z_Q) are the Cartesian offsets of the dipole and quadrupole components relative to the neutron-star center, (α_D, ϕ_D) and (α_Q, ϕ_Q) are their corresponding orientation angles (inclination and azimuth), and B_Q/B_D is the quadrupole-to-dipole surface field-strength ratio.

7.1. Embedding Space and Retrieval Performance

We embedded the 10M light curve database using the pretrained masked U-Net encoder, extracting 768-dimensional multi-level embeddings by concatenating pooled encoder level-3 (256-dim; Eq. (9)) and bottleneck (512-dim; Eq. (8)) features. This representation captures both local phase-resolved structure and global pulse morphology, enabling cosine-similarity retrieval in latent space (Eq. (12)). The retrieval completes in $\sim$17.6 seconds on a CPU, as reported in Table 2.

For the observed PSR J0030+0451 light curve, we compared three retrieval strategies: (i) cosine similarity on raw z-scored light curves, (ii) single-level bottleneck embeddings, and (iii) multi-level embeddings. Retrieval quality is assessed using the median normalized squared error (MdNSE) metric introduced by Olmschenk et al. (2025) (Eq. (18)). The qualitative retrieval overlays are reported in Appendix C. Multi-level embeddings produced retrieved curves that most faithfully match phase-dependent structure, including peak width, asymmetry, and baseline morphology. Bottleneck-only embeddings showed modest degradation, while raw-space cosine similarity yielded frequent peak misalignment and baseline mismatch.

Table 1. Hill-climbing refinement improves mean negative log-likelihood toward the MCMC reference. Each row shows results after refining all 4,000 seeds with the specified number of jointly perturbed parameters P.

P	Mean NLL	Δ from Initial	Δ from MCMC
Initial	$-41{,}513.00$	—	-612.14
1	$-42{,}089.93$	$+576.93$	-35.21
2	$-42{,}122.51$	$+609.51$	-2.63
3	$-42{,}123.58$	$+610.58$	-1.56
4	$-42{,}124.07$	$+611.07$	-1.07
5	$-42{,}124.30$	$+611.30$	-0.84
MCMC	$-42{,}125.14$	—	—

7.2. Hill-Climbing Refinement Results

Starting from K=4,000 retrieved candidates (see Section 7.3 for the choice of K), we applied adaptive hill-climbing refinement with P jointly-perturbed parameters at each iteration. Table 1 shows systematic improvement in mean negative log-likelihood toward the MCMC reference as P increases from 1 to 5.

Refinement with P=5 achieves within 0.84 negative log-likelihood units of nested sampling MCMC, with most improvement occurring by P=2. Beyond P=3, gains plateau with diminishing returns: the improvement from P=4 to P=5 is only $+0.23$ negative log-likelihood units. This suggests that jointly perturbing 5 parameters is sufficient to navigate the complex, degenerate posterior landscape of the 11-dimensional pulsar parameter space.

We further analyze the effect of refinement on posterior uncertainty in Appendix E. While hill-climbing improves likelihood and modestly reduces coverage error, its impact on uncertainty calibration is mixed across parameters, and it does not fully recover calibrated posterior uncertainty.

Figure 3 shows refined posterior distributions for parameters of PSR J0030+0451. We map them to corresponding predicted light curves using our light curve simulator. We find that our predicted parameter posteriors show qualitative agreement with MCMC in terms of posterior support and multimodal structure, though differences remain in marginal widths and mode weights for several parameters (Appendix D). The parameter posteriors map to the same region of the light-curve space as the observed light curve and align with predicted light curves from the MCMC inference approach used as a baseline in this work. The quadrupole parameters $\{x_Q, y_Q, z_Q, \alpha_Q, \phi_Q, B_Q/B_D\}$ exhibit particularly strong agreement with MCMC, while the quadrupole orientation angles $\{\alpha_Q, \phi_Q\}$ preserve the expected multimodal structure arising from parameter degeneracies. No-

tably, for α_D and ϕ_D, the refined posterior captures multiple distinct modes that reflect the inherent symmetries in the magnetic field geometry. This demonstrates that local refinement successfully sharpens posterior density within each mode without collapsing the degenerate structure captured during retrieval.

To further quantify agreement with the MCMC posterior, we report distance metrics including Wasserstein distance, KS statistic, Jensen–Shannon divergence, and credible interval overlap (Appendix D).

7.3. Why is the Retrieval Size K=4,000?

We empirically determine the optimal number of retrieved neighbors K by measuring convergence of the approximate posterior as we increase K s.t. $K_0 < K_1 < \cdots < K_{\text{max}}$. For each parameter θ_i, we compute the KL divergence between the empirical distribution formed by the top-K_k retrieved samples and the top-K_{max} retrieved samples.

$$D_{KL}^{(i)}(K) = \sum_{\text{bins}} p_{\text{retrieved}}^{(i)}(\theta_i; K_{\text{max}}) \log \frac{p_{\text{retrieved}}^{(i)}(\theta_i; K_{\text{max}})}{p_{\text{retrieved}}^{(i)}(\theta_i; K)}, \tag{14}$$

where both distributions are discretized using histogram binning with consistent bin edges across all K values.

In Figure 2, the average KL divergence monotonically decreases with the increase of K, where we measure against $K_{\text{max}} = 4,000$. Both refinement strategies ($P = 1$ and $P = 5$) converge so that the estimated posteriors for $K = 3,000$ and $K = 4,000$ are nearly identical.

Based on this convergence, we select $K = 4{,}000$ for all experiments to ensure the retrieval stage captures the full support of the posterior while maintaining computational efficiency.

7.4. Computational Performance

Table 2 summarizes end-to-end computational costs. Our method achieves a **120× speedup** over MCMC (24 hours → 12 minutes) while recovering key posterior support and multimodal structure.

Key insights: (i) Offline preprocessing (6.6 min GPU for 10M light curves at 25,300 light curves per second) is amortized across all queries—a 120× speedup over CPU (14 hours). **(ii)** Online inference (12 min) consists of retrieval (17.6s) plus refinement (11.8 min). **(iii)** On GPUs, processing all 4,000 refinement seeds in parallel takes 66.5 seconds per seed, compared to 363 seconds per seed on CPU (a 5.5× speedup). **(iv)** Retrieval remains efficient at 10M-scale.

Table 2. End-to-end computational cost breakdown for posterior inference.

Component	Hardware	Time
Offline (one-time)		
Embed 10M light curves	RTX A5000 GPU	∼6.6 min
Online (per query)		
Query embedding	Apple M-series CPU	<1 sec
k-NN retrieval	Apple M-series CPU	17.6 sec
Refinement (P=1 − 5)	4,000 CPUs	11.8 min
Total online		**∼12 min**
Baseline		
MCMC	4,000 CPUs	**24 hours**

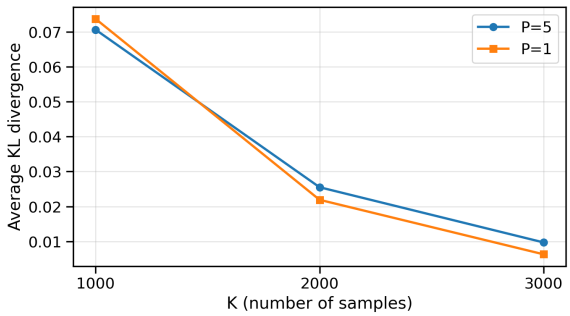

Figure 2. Average KL divergence versus retrieval size K for P=1 (orange) and P=5 (blue) refinement. KL divergence decreases monotonically and levels off at $K \approx 3{,}000$ (KL ≈ 0.01), justifying the use of K=4,000.

7.5. Generalization to Gamma-Ray Spectroscopy

To assess domain transferability, we applied the framework to Ba-133 photopeak fitting—a 4-parameter inference problem in nuclear spectroscopy. The task is to estimate amplitude a, peak center x_c, width w, and background y_0 from a Gaussian-plus-background model $f(E; \boldsymbol{\theta}) = a \exp(-(E - x_c)^2/2w^2) + y_0$, where $E \in [156, 164]$ keV is photon energy. We generated 10M synthetic photopeaks by sampling parameters from priors centered on an observed Ba-133 spectrum and applying Poisson observation noise. Each spectrum was resampled to 40 uniformly-spaced energy bins. Training the same U-Net architecture (base_filters=64, 768-dim embeddings, 20 epochs) on 10M synthetic spectra top-$K = 4{,}000$ retrieval achieved cosine similarity > 0.999 with observed photopeaks (Figure 5 in Appendix). Hill-climbing refinement (Poisson loss, $P = 3$) reduced mean negative log-likelihood from -9227.6425 to -9228.7829, matching the MCMC reference. Results are included in Appendix F.

8. Conclusions

We introduced an accelerated framework for posterior inference from pulsar light curves that avoids computationally

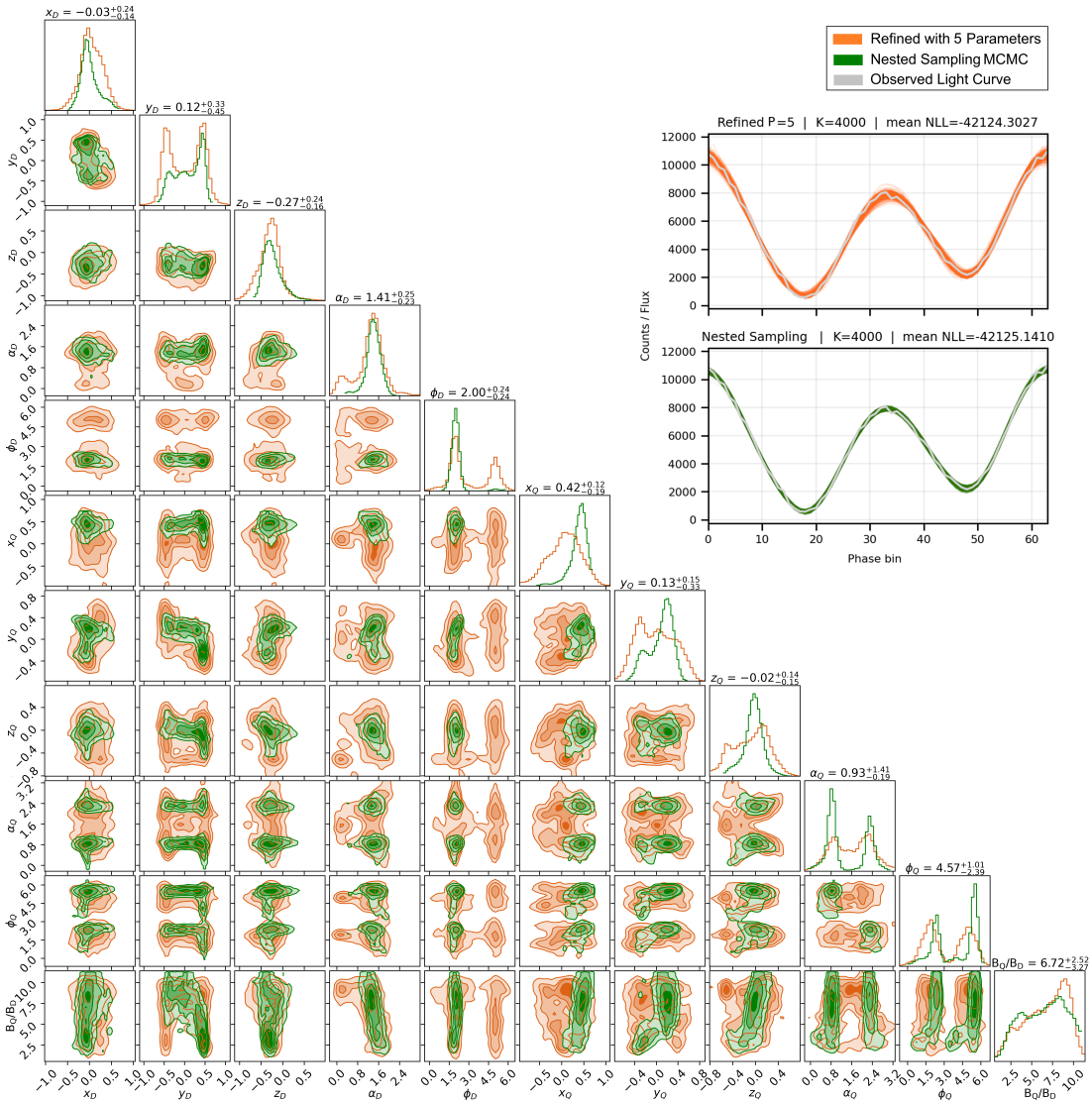

Figure 3. **Left**: Posterior distributions of pulsar parameters comparing results from (i) the embedding retrieval and hill climbing refinement over 5 parameters approach in the work (in orange) and (ii) a traditional MCMC informed by a nested sampling approach (in green). **Right**: Predicted light curves corresponding to parameter posterior configurations from this work (orange) compared to those from the baseline MCMC pipeline (green). The predicted posteriors map to light curves that align well with the observed light curve and show qualitative agreement with the MCMC baseline.

expensive MCMC sampling. We pretrain a U-Net model to reconstruct masked light curves, yielding an encoder that learns a meaningful multi-scale embedding space for pulsar light-curve representations. Given a query light curve, our approach retrieves simulated light curves with similar characteristics from a simulation bank by measuring similarity in the learned embedding space. The parameter sets associated with the retrieved light curves define an empirical approximation to the posterior distribution, naturally capturing multimodality and parameter degeneracies present in the data. We then locally refine each retrieved region of parameter space using hill-climbing optimization guided by a forward simulator, sharpening posterior mass within each region of support while preserving distinct modes and degeneracies identified during the retrieval stage. When applied to NICER observations of PSR J0030+0451, our approach achieves a $120\times$ reduction in inference time while recovering key posterior support and multimodal structure, with results broadly consistent with MCMC. The method offers an empirically effective approximation for likelihood-free inference when exhaustive Bayesian sampling is computationally impractical. In addition to pulsar modeling, results on a gamma-ray spectrum inference task indicate that the framework generalizes to other physics-based inference problems involving complex posterior structure.

Acknowledgements

This work was supported by the NASA theoretical and computational astrophysics network project with grant number 22-TCAN22-0027. A.B.S., D.O., and S.D. were supported by the Laboratory Directed Research and Development (LDRD) program of LANL through its Information Science & Technology Institute under project number 20258368CT-IST. This research used resources provided by the Los Alamos National Laboratory Institutional Computing Program, which is supported by the U.S. Department of Energy National Nuclear Security Administration under Contract No. 89233218CNA000001.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Biau, G., Cérou, F., and Guyader, A. New insights into approximate bayesian computation. In *Annales de l'IHP Probabilités et statistiques*, volume 51, pp. 376–403, 2015.

Gendreau, K. C., Arzoumanian, Z., et al. The neutron star interior composition explorer (nicer): Design and development. In den Herder, J.-W. A., Takahashi, T., and Bautz, M. (eds.), *Space Telescopes and Instrumentation 2016: Ultraviolet to Gamma Ray*, volume 9905 of *Proceedings of SPIE*, pp. 99051H, 2016. doi: 10.1117/12.2231304.

Gilks, W. R., Richardson, S., and Spiegelhalter, D. J. (eds.). *Markov Chain Monte Carlo in Practice*. Chapman and Hall/CRC, New York, 1 edition, 1995. ISBN 9780429170232. doi: 10.1201/b14835.

Hastings, W. K. Monte Carlo Sampling Methods using Markov Chains and their Applications. *Biometrika*, 57 (1):97–109, April 1970. doi: 10.1093/biomet/57.1.97.

Kalapotharakos, C., Wadiasingh, Z., Harding, A. K., and Kazanas, D. The Multipolar Magnetic Field of the Millisecond Pulsar PSR J0030+0451. *The Astrophysics Journal*, 907(2):63, February 2021. doi: 10.3847/1538-4357/abcec0.

Kolda, T. G., Lewis, R. M., and Torczon, V. Optimization by direct search: New perspectives on some classical and modern methods. *SIAM Review*, 45(3):385–482, 2003.

Lattimer, J. M. and Prakash, M. The equation of state of hot, dense matter and neutron stars. *Physics Report*, 621:127–164, March 2016. doi: 10.1016/j.physrep.2015.12.005.

Lin, M., Chen, Q., and Yan, S. Network In Network. *arXiv e-prints*, art. arXiv:1312.4400, December 2013. doi: 10.48550/arXiv.1312.4400.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. Equation of State Calculations by Fast Computing Machines. *Journal of Chemical Physics*, 21(6):1087–1092, June 1953. doi: 10.1063/1.1699114.

Mo-Ghani. Sampling: A collection of mcmc sampling examples. https://mo-ghani.github.io/Sampling/, 2025. Accessed: January 2026.

Olmschenk, G., Broadbent, E., Kalapotharakos, C., Wallace, W. F., Lechien, T., Wadiasingh, Z., Kazanas, D., and Harding, A. Pioneering high-speed pulsar parameter estimation using convolutional neural networks. *The Astrophysical Journal*, 991(2):169, September 2025. ISSN 1538-4357. doi: 10.3847/1538-4357/ae03c0. URL http://dx.doi.org/10.3847/1538-4357/ae03c0.

Özel, F. and Freire, P. Masses, Radii, and the Equation of State of Neutron Stars. *Annual Review of Astronomy and Astrophysics*, 54:401–440, September 2016. doi: 10.1146/annurev-astro-081915-023322.

Ronneberger, O., Fischer, P., and Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. *arXiv e-prints*, art. arXiv:1505.04597, May 2015. doi: 10.48550/arXiv.1505.04597.

Siddik, A. B., Oyen, D., De, S., Olmschenk, G., and Kalapotharakos, C. Degeneracy-aware pulsar parameter estimation from light curves via deep learning and test-time optimization. *arXiv preprint arXiv:2510.18807*, 2025.

Smith, F. G. *Pulsars*. Cambridge Monographs on Physics, Cambridge: University Press, 1977.

Sturrock, P. A. A model of pulsars. *The Astrophysics Journal*, 164:529, March 1971. doi: 10.1086/150865.

Tovey, C. A. Hill climbing with multiple local optima. *SIAM Journal on Algebraic Discrete Methods*, 6(3):384–393, 1985. doi: 10.1137/0606040. URL https://doi.org/10.1137/0606040.

Wright, M. H. Direct search methods: Once scorned, now respectable. In Griffiths, D. F. and Watson, G. A. (eds.), *Numerical Analysis 1995*, volume 344 of *Pitman Research Notes in Mathematics Series*, pp. 191–208. CRC Press, Boca Raton, FL, 1996.

A. U-Net Architecture

A.1. Encoder Architecture

Let $\mathbf{x} \in \mathbb{R}^T$ denote a light curve with T phase bins. After z-score normalization and reshaping to channel-first format $\tilde{\mathbf{x}} \in \mathbb{R}^{1\times T}$, the encoder $f_\phi : \mathbb{R}^{1\times T} \to \mathbb{R}^{C_b \times T_b}$ applies $L = 3$ hierarchical blocks with progressive downsampling:

$$\begin{aligned}
\mathbf{e}^{(1)} &= \mathrm{Enc}_1(\tilde{\mathbf{x}}^{\text{mask}}) && \in \mathbb{R}^{64\times 64}, \\
\mathbf{e}^{(2)} &= \mathrm{Enc}_2(\mathrm{Pool}(\mathbf{e}^{(1)})) && \in \mathbb{R}^{128\times 32}, \\
\mathbf{e}^{(3)} &= \mathrm{Enc}_3(\mathrm{Pool}(\mathbf{e}^{(2)})) && \in \mathbb{R}^{256\times 16}, \\
\mathbf{b} &= \mathrm{Enc}_b(\mathrm{Pool}(\mathbf{e}^{(3)})) && \in \mathbb{R}^{512\times 8},
\end{aligned} \tag{15}$$

where $\mathbf{e}^{(\ell)}$ denotes the ℓ-th encoder feature map, $\mathbf{b}$ is the bottleneck representation, and $\mathrm{Pool}(\cdot)$ is max-pooling with stride 2.

Convolutional blocks. Each encoder block Enc_ℓ consists of two 1D convolutional layers with ReLU activations:

$$\mathrm{Enc}_\ell(\mathbf{u}) = \sigma(\mathbf{W}_\ell^{(2)} * \sigma(\mathbf{W}_\ell^{(1)} * \mathbf{u})), \tag{16}$$

where $*$ denotes 1D convolution with kernel size 3, padding 1, and $\sigma(\cdot) = \max(0, \cdot)$ is ReLU activation function. This configuration preserves spatial dimensions within each block while Pool halves temporal resolution between blocks.

A.2. Decoder Architecture

The U-Net decoder g_ψ mirrors the encoder hierarchy via three transpose-convolution upsampling stages with concatenated skip-connections. Let $\mathbf{b} \in \mathbb{R}^{512\times 8}$ be the bottleneck feature and $\mathbf{e}^{(\ell)}$ the encoder feature maps. The decoder computes:

$$\begin{aligned}
\mathbf{u}^{(3)} &= \mathrm{Up}_3(\mathbf{b}) \in \mathbb{R}^{256\times 16}, \\
\mathbf{d}^{(3)} &= \mathrm{Dec}_3\big([\mathbf{u}^{(3)} \| \mathbf{e}^{(3)}]\big) \in \mathbb{R}^{256\times 16}, \\
\mathbf{u}^{(2)} &= \mathrm{Up}_2(\mathbf{d}^{(3)}) \in \mathbb{R}^{128\times 32}, \\
\mathbf{d}^{(2)} &= \mathrm{Dec}_2\big([\mathbf{u}^{(2)} \| \mathbf{e}^{(2)}]\big) \in \mathbb{R}^{128\times 32}, \\
\mathbf{u}^{(1)} &= \mathrm{Up}_1(\mathbf{d}^{(2)}) \in \mathbb{R}^{64\times 64}, \\
\mathbf{d}^{(1)} &= \mathrm{Dec}_1\big([\mathbf{u}^{(1)} \| \mathbf{e}^{(1)}]\big) \in \mathbb{R}^{64\times 64}, \\
\hat{\mathbf{x}} &= \mathrm{Conv}_{1\times 1}\big(\mathbf{d}^{(1)}\big) \in \mathbb{R}^{1\times 64}.
\end{aligned}$$

Each Up_ℓ upsamples via transposed convolution (kernel size 2, stride 2), halving channel depth while doubling spatial resolution. Concatenation $[\cdot\|\cdot]$ along the channel dimension doubles the input channels to each Dec_ℓ block. Each decoder block applies:

$$\mathrm{Dec}_\ell(\mathbf{z}) = \sigma\big(W_\ell^{(2)} * \sigma(W_\ell^{(1)} * \mathbf{z})\big),$$

where the first convolution processes concatenated features ($2C_{\text{up}} \to C_{\text{out}}$) and the second refines within C_{out}.

B. Evaluation Metric for Retrieved Light Curve Quality

To quantitatively evaluate the quality of retrieved light curves, we employ a median normalized squared error (MdNSE) metric (Olmschenk et al., 2025) computed in the unnormalized signal space. Given a query light curve $\tilde{\mathbf{x}} \in \mathbb{R}^T$ and a retrieved candidate $\tilde{\mathbf{x}}_{\textit{retrieved}} \in \mathbb{R}^T$ (both z-score normalized), we first invert the normalization:

$$\mathbf{x} = \sigma\tilde{\mathbf{x}} + \mu, \quad \mathbf{x}_{\textit{retrieved}} = \sigma\tilde{\mathbf{x}}_{\textit{retrieved}} + \mu, \tag{17}$$

where μ and σ are the precomputed mean and standard deviation from the training data set.

$$\text{MdNSE} = \frac{\sum_{t=1}^{T}(x_{retrieved,t} - x_t)^2}{(\text{median}(\mathbf{x}))^2}, \tag{18}$$

where the numerator measures the sum of squared pointwise errors in unnormalized space, and the denominator normalizes by the squared median flux of the query light curve, making the metric scale-invariant. Lower MdNSE values indicate better agreement between the retrieved and query light curves. The MdNSE was chosen over χ^2 because the simulated data lack associated error values, and the metric treats all light curves more equally by focusing on shape rather than amplitude (Olmschenk et al., 2025).

C. Ablation Results

We ablate the retrieval representation by comparing three similarity spaces: (i) cosine similarity on raw z-scored light curves, (ii) bottleneck-only embeddings (global average pooled U-Net bottleneck), and (iii) multi-level embeddings formed by concatenating pooled encoder level-3 features with pooled bottleneck features. Across multiple query light curves, multi-level embeddings produce the most faithful nearest-neighbor sets, yielding retrieved curves that tightly track the query's phase-dependent morphology (peak width, asymmetry, and baseline structure), while bottleneck-only embeddings show modestly larger dispersion. In contrast, raw-space cosine similarity is substantially less reliable and often retrieves curves with mismatched peak shapes and off-peak baselines. These results indicate that combining intermediate and bottleneck features is important for preserving both local and global structure in the embedding space, improving retrieval fidelity for posterior initialization.

Figure 4 compares nearest-neighbor retrieval across three similarity spaces: bottleneck-only embeddings obtained via global average pooling of the U-Net bottleneck, multi-level embeddings that fuse encoder level-3 features with bottleneck features, and direct cosine similarity computed on raw, z-scored light curves. For each query (rows), we visualize the observed light curve (colored) together with the top-K, where $K = 5$, retrieved matches (gray, with opacity decreasing by rank). Across all queries, retrieval in the learned embedding space consistently produces matches that more closely follow the query morphology than direct cosine similarity on raw curves.

Multi-level embeddings yield the tightest retrieval sets, with the top-K curves forming compact envelopes around the query and exhibiting minimal phase-dependent deviations. This improvement arises because the multi-level representation combines complementary information from different depths of the encoder: bottleneck features encode global pulse-profile structure through their large receptive fields, while intermediate encoder features (level 3) retain higher phase resolution and capture localized morphological details such as peak width, asymmetry, and shoulder structure. Encoder level-1 features would be too local and susceptible to high-frequency fluctuations, whereas level-3 provides a "sweet spot"—local enough to preserve fine, phase-resolved structure, yet deep enough to suppress noise and irrelevant variations. In contrast, single-level embeddings rely solely on bottleneck features and therefore emphasize global shape while partially discarding fine-scale phase information, leading to slightly increased variability among retrieved matches.

Direct cosine similarity on raw light curves performs substantially worse, frequently retrieving curves with mismatched peak shapes and off-peak baselines. This behavior reflects the absence of learned invariances in raw phase space, where cosine similarity is overly sensitive to local fluctuations and insufficiently discriminative under strong parameter degeneracies. Overall, these results support the use of multi-level embeddings for retrieval-based posterior initialization, as they more faithfully preserve both global and local structure in pulsar light curves.

D. Quantitative Posterior Comparison with MCMC

We quantitatively compare the refined posterior produced by our method with the MCMC reference posterior for PSR J0030+0451 using multiple distributional metrics. Specifically, we report Wasserstein distance (WD), Kolmogorov–Smirnov (KS) statistic, Jensen–Shannon divergence (JSD), and 68% credible interval (CI) agreement for each parameter.

Table 3 summarizes the results.

These results indicate partial but non-uniform agreement with the MCMC posterior. While several parameters (e.g., B_Q/B_D) show strong alignment, others exhibit noticeable discrepancies, particularly in marginal widths and mode weights. This supports interpreting the refined distribution as an approximation to posterior support rather than a fully calibrated posterior density.

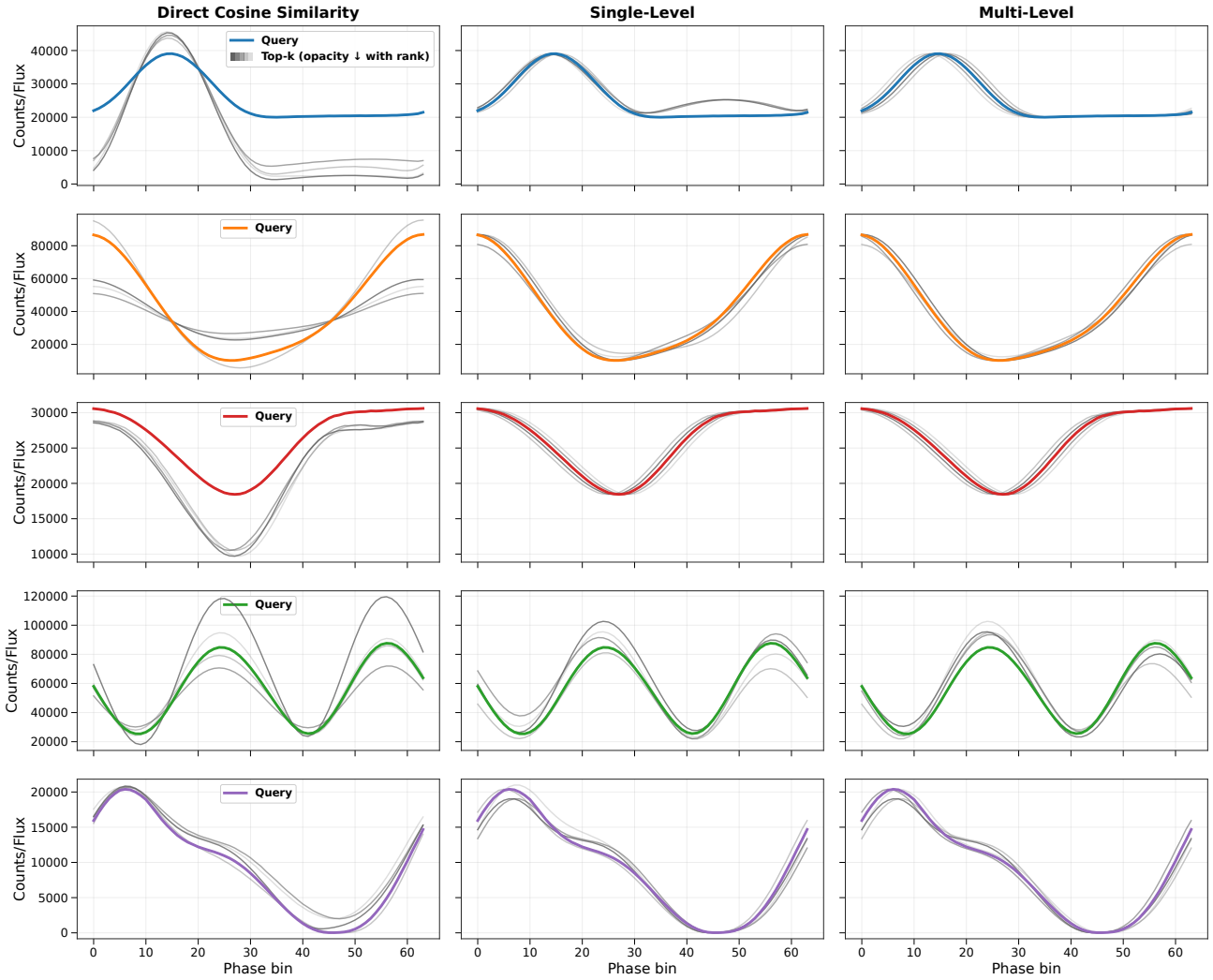

Figure 4. Visual diagnostic of retrieval fidelity. For each query (row), the spread of retrieved curves across ranks reflects how well the similarity space concentrates around the observed morphology; multi-level embeddings show the smallest spread, whereas raw-space cosine similarity exhibits larger deviations in peak and baseline structure.

E. Effect of Hill-Climbing on Posterior Uncertainty

To isolate the effect of hill-climbing refinement on posterior uncertainty, we compare the retrieved posterior (R) and refined posterior (F) against the MCMC reference using empirical coverage.

For each parameter θ_j and credibility level $c \in \{0.50, 0.68, 0.90\}$, we compute:

$$\text{CoverageError}_j(c) = |\text{Cov}_j(c) - c|, \tag{19}$$

where $\text{Cov}_j(c)$ is the fraction of MCMC samples falling within the corresponding interval of the approximate posterior. Low coverage indicates overly narrow (overconfident) intervals, while high coverage indicates overly broad (underconfident) intervals.

To examine parameter-wise behavior, we report empirical coverage for all parameters in Table 4.

As shown in Table 4, the effect of refinement is not uniform across parameters: some marginals show clear improvements (e.g., x_Q, particularly at higher coverage levels), while others exhibit minimal change or remain miscalibrated. Notably, refinement substantially improves coverage for some poorly calibrated parameters (e.g., x_Q), while having limited impact on already well-calibrated parameters (e.g., B_Q/B_D), highlighting heterogeneous effects across the parameter space.

Table 3. Quantitative comparison between refined posterior and MCMC reference. Lower WD, KS, and JSD indicate better agreement. CI_{68} measures overlap in credible intervals.

Parameter	WD	KS	JSD	CI_{68} (%)
x_D	0.285	0.154	0.192	57.85
y_D	0.250	0.158	0.210	47.33
z_D	0.264	0.130	0.225	64.13
α_D	0.460	0.176	0.249	48.50
ϕ_D	0.861	0.412	0.433	38.03
x_Q	0.986	0.496	0.438	26.18
y_Q	0.507	0.203	0.319	41.93
z_Q	0.526	0.216	0.323	31.25
α_Q	0.308	0.285	0.378	69.38
ϕ_Q	0.366	0.277	0.394	51.68
B_Q/B_D	0.118	0.059	0.141	68.83

Table 4. Empirical coverage (%) for all parameters comparing the retrieved (R) and refined (F) posteriors.

	50%		68%		90%	
Parameter	R	F	R	F	R	F
x_D	64.05	56.00	80.10	74.08	93.65	94.23
y_D	46.00	50.65	60.65	68.83	90.72	96.25
z_D	49.98	48.95	66.47	68.50	84.92	91.20
α_D	80.67	78.75	93.35	92.20	99.90	99.80
ϕ_D	74.85	64.95	87.80	81.53	99.20	99.12
x_Q	14.50	15.38	27.38	31.65	59.03	81.47
y_Q	72.65	75.83	90.60	92.65	99.40	99.55
z_Q	79.62	80.17	90.45	90.20	96.17	96.83
α_Q	34.52	35.02	86.10	82.97	96.75	96.50
ϕ_Q	42.60	42.43	50.62	50.30	96.43	95.43
B_Q/B_D	51.38	49.80	65.97	65.80	88.90	89.10

We summarize this behavior using mean absolute coverage error across parameters in Table 5.

Table 5. Mean absolute coverage error (**lower is better**).

Method	50%	68%	90%
Retrieved (R)	0.169	0.172	0.081
Refined (F)	**0.150**	**0.148**	**0.062**

Table 5 shows that hill-climbing reduces coverage error across all levels, indicating a modest improvement in uncertainty calibration on average.

Overall, hill-climbing redistributes posterior mass toward higher-likelihood regions, leading to improved average calibration while preserving variability in parameter-wise behavior. This reinforces that the refinement step enhances likelihood alignment but does not fully recover calibrated posterior uncertainty.

F. Gamma-Ray Spectroscopy: Posterior Visualization and Comparison

We evaluate the proposed retrieval-based posterior inference framework on a gamma-ray spectroscopy (GRS) problem involving photopeak fitting for a Ba-133 spectrum, following the standard formulation used in PyMC3 and dynesty demonstrations (Mo-Ghani, 2025).

F.1. Forward Model

Each photopeak is modeled as a Gaussian signal superimposed on a constant background:

$$F(E;\boldsymbol{\theta}) = a\exp\left(-\frac{(E-x_c)^2}{2w^2}\right) + y_0, \tag{20}$$

where $\boldsymbol{\theta} = (a, x_c, w, y_0)$ denote the amplitude, peak center (keV), standard deviation (keV), and background level, respectively. The energy axis E is discretized into 40 uniformly spaced bins over the photopeak region.

F.2. Synthetic Dataset and Retrieval

Synthetic spectra were generated by sampling parameters from Gaussian priors centered on an observed Ba-133 photopeak, consistent with the priors used in the PyMC3 and dynesty reference implementation (Mo-Ghani, 2025). Poisson noise was applied to simulate photon-counting statistics. A total of 10M synthetic spectra were generated and we used the same masked 1D U-Net architecture and training procedure described in Section 4, without modification.

For a query spectrum, the pretrained encoder produces a fixed-dimensional embedding, and cosine similarity in embedding space is used to retrieve the top-$K = 4{,}000$ nearest simulated spectra. The retrieved parameter sets form an empirical posterior approximation.

F.3. Results after Refinement

Figure 6 shows posterior marginals and predicted photopeaks for the Ba-133 inference task. The initial retrieval ensemble (top row) produces broad but correctly centered parameter distributions for (a, x_c, w, y_0). Local refinement sharpens the posterior and improves data agreement (see Figure 5).

Most posterior contraction occurs at the first refinement stage ($P = 1$), with minimal change for $P = 2$ and $P = 3$. Correspondingly, the mean Poisson loss improves from -9227.73 (initial) to -9228.78 after refinement, with differences across P below 10^{-2}. The predicted spectra after refinement closely match the observed photopeak.

Figure 6 compares the refined posterior to the MCMC reference. The refined posterior recovers the same high-probability region in parameter space and exhibits close agreement in the one-dimensional marginals for all parameters. While differences remain in the joint structure, the refined posterior captures the correct parameter scales and central values identified by MCMC.

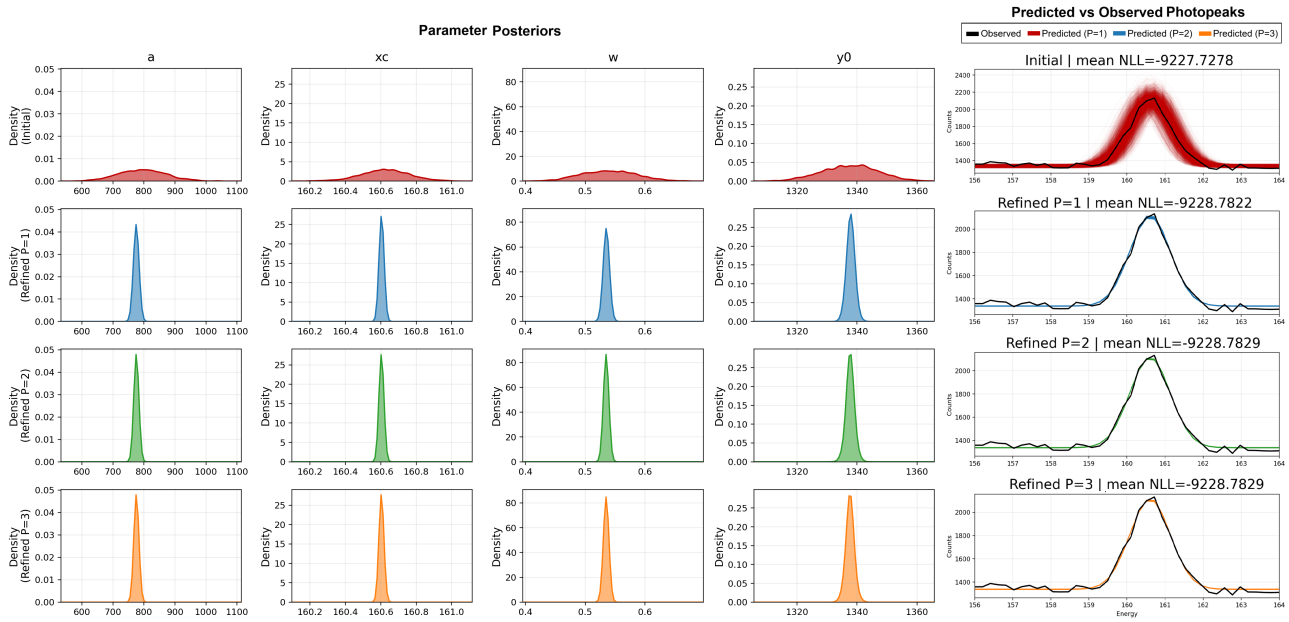

Figure 5. Predictions on gamma-ray spectroscopy data using the refinement framework in this work. The first 4 columns in each row show predicted posteriors of parameters describing a photopeak for a sample Ba-133. The last column shows the predicted photopeak profiles corresponding to the predicted parameters compared to the observed photopeak. The negative log-likelihood improves as the number of jointly perturbed parameters increases during refinement and converges at the 3-parameter stage (P =3).

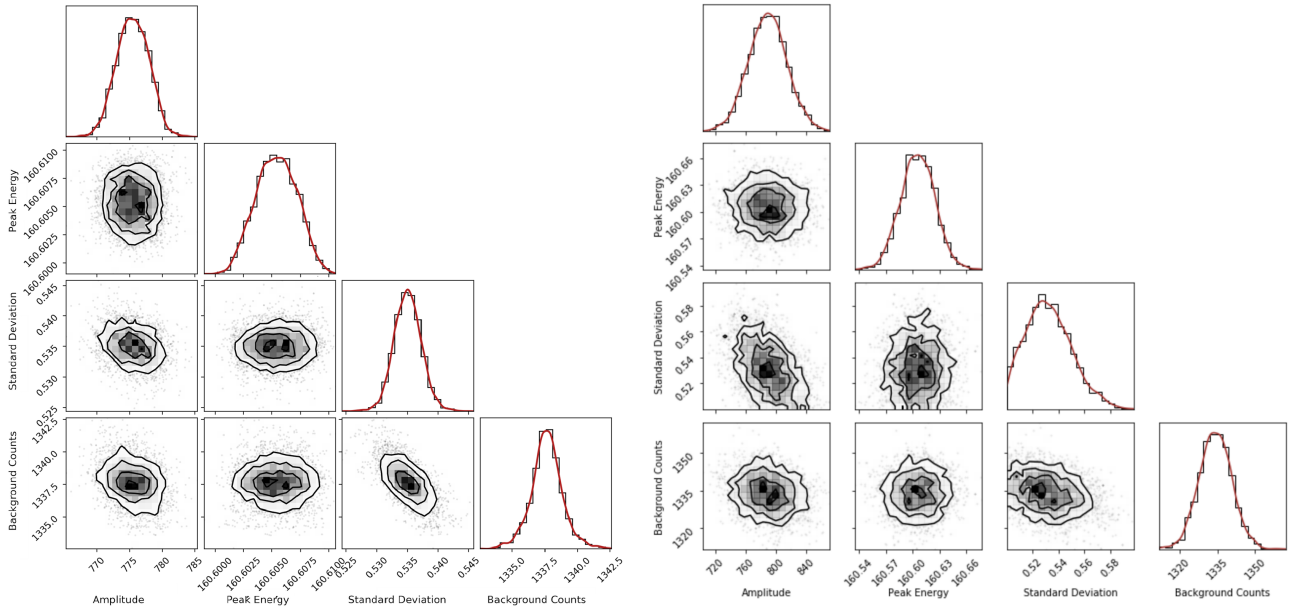

Figure 6. Joint and marginal posterior distributions for the Ba-133 photopeak parameters. **Left:** Posterior obtained from embedding-based retrieval followed by local refinement. **Right:** Reference posterior from MCMC sampling (Mo-Ghani, 2025). Both posteriors exhibit the same dominant correlations, including the amplitude–width degeneracy characteristic of Gaussian photopeak models.