# Energy-Based Operator Learning in Function Space

**Anonymous Authors**[1]

## Abstract

We propose Energy-Based Operators (EBOs), an architecture-agnostic framework for learning conditional distributions over functions on continuous domains. An EBO defines a scalar energy over target functions given an input function and induces a probability model through a Gaussian reference. The resulting score field is obtained as the gradient of the parametrized energy to perform function-space iterative energy minimization (EM). Our model shows strong performance in function generation, such as super-resolution and forecasting, over various 1D function classes (oscillations, damping, and Izhikevich) in comparison with prediction operators and denoising operators over various architecture backbones. Moreover, it achieves strong performance over PDEs, namely Navier-Stokes, Darcy flow and Burgers. Notably, our model successfully detects anomalous functions by automatically assigning high energy without any supervision. It enables seizure detection and volatility prediction after learning neural dynamics and market microstructure dynamics without pre-defined labels during training, highlighting its effectiveness for both learning dynamical systems and detecting functional anomalies arising in scientific simulations.

## 1. Introduction

Energy-based models (EBMs) (Hopfield, 1982; Ackley et al., 1985; LeCun et al., 2006) provide a flexible framework for modeling complex distributions by associating a scalar energy with each configuration, where lower energy corresponds to higher probability. Initially inspired by theoretical models of neural dynamics underlying associative memory in the brain, EBMs have demonstrated significant efficacy in capturing structures of various machine learning tasks, including image generation (Du & Mordatch, 2019; Wang & Du, 2025), language modeling (Gladstone et al., 2025) and logical reasoning (Oarga & Du, 2025).

While EBMs have achieved success in finite-dimensional settings, extending them to *function spaces* presents unique challenges. Many scientific and engineering problems—from solving partial differential equations to modeling physiological signals—are naturally posed over continuous domains, where the objects of interest are functions. Naively discretizing these functions and applying standard EBMs ignores the underlying continuum structure, which introduces sensitivity to resolution changes (Berner et al., 2025).

Recent extensions to score-based diffusion operators (Lim et al., 2025; Wang et al., 2025; Franzese et al., 2023; Zhang & Wonka, 2024) parameterize the score directly, thereby supporting sampling but exposing no scalar energy and tying inference to a fixed noise schedule with a time-dependent score. We propose *Energy-Based Operators* (EBOs), which parameterize the energy functional itself, combining the flexibility of energy-based modeling with the discretization-invariance of neural operators (Figure 1). The scalar energy enables unsupervised anomaly detection on real-world signals, and the time-independent energy supports test-time scaling, where more optimization steps can be allocated to harder instances without retraining (Du et al., 2022; Gladstone et al., 2025). Our key contributions are:

- We generalize energy-based models to function space neural operators, defining a scalar energy over functions and deriving the score as its gradient.
- We theoretically analyze EBO's discretization invariance and universal approximation properties.
- We demonstrate strong performance of EBO in function generation and functional anomaly detection over both diverse function classes and realistic datasets.

## 2. Energy-Based Operators

We fix two finite-dimensional domains $\Omega_X \subset \mathbb{R}^{n_x}$ and $\Omega_Y \subset \mathbb{R}^{n_y}$ equipped with Borel $\sigma$-algebras $\mathcal{B}(\Omega_X), \mathcal{B}(\Omega_Y)$ and reference measures $\lambda_X$ and $\lambda_Y$[1]. We consider a *con-*

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

[1]We can typically assume that they are Lebesgue measures.

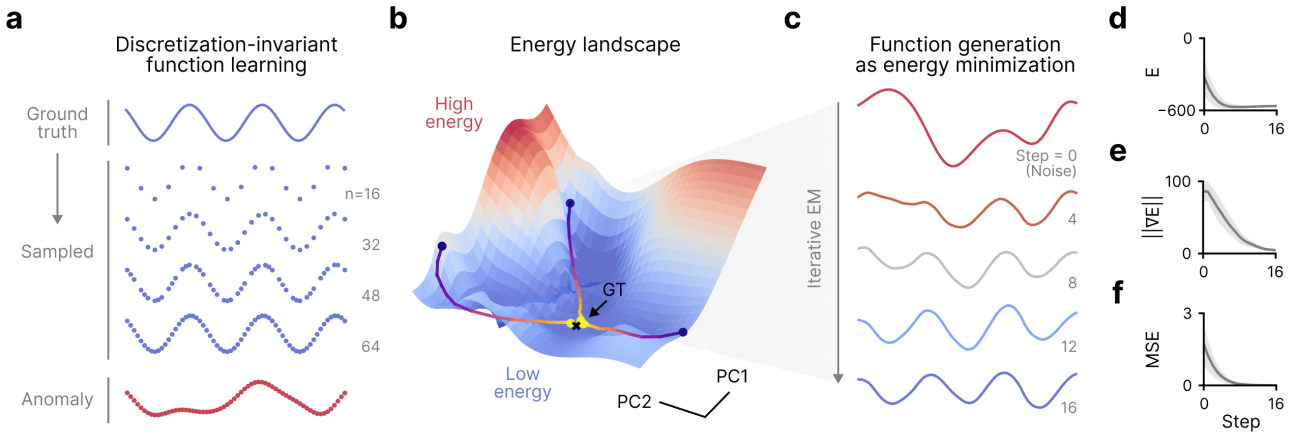


*Figure 1.* **Energy-Based Operator for discretization-invariant function learning.** (**a**) A sinusoidal ground truth function and various sampled examples with different sampling ratios. An anomaly function is defined as a function that violates the dynamics of the ground truth function. (**b**) Energy landscape of the EBO learned for the sinusoidal function, obtained by Principal Component Analysis. Red and blue colors indicate high energy and low energy, respectively. Three trajectories of energy minimization starting from different initial noises are displayed. (**c**) Function generation as energy minimization. Sampled functions displayed from a representative EM trajectory in the energy landscape. (**d-f**) Convergence of the iterative EM trajectory. (**d**) Energy. (**e**) Norm of the gradient of energy. (**f**) Mean squared error between the sampled function and the given context.

*text function space* $\mathcal{H}_X \triangleq \mathcal{F}(\Omega_X, \mathbb{R}^d)$, and a *target function space* $\mathcal{H}_Y \triangleq \mathcal{F}(\Omega_Y, \mathbb{R}^m)$, where $\mathcal{H}_X$ and $\mathcal{H}_Y$ are assumed to be real separable Hilbert spaces with appropriate inner products[2]. To make point-wise losses and point-wise expressions $u(y)$ meaningful at the continuum level, we assume point-wise evaluation is continuous on $\mathcal{H}_Y$ with $u(y) \triangleq \delta_y(u)$.

### 2.1. Definition and Formulation

We make the following definition of our method in functional form.

**Definition 2.1** (Energy-Based Operator)**.** An **Energy-Based Operator (EBO)** is a map

$$\Phi_\theta : \mathcal{H}_X \times \mathcal{H}_Y \longrightarrow \mathbb{R}, \qquad (f, u) \mapsto \Phi_\theta(f, u), \tag{1}$$

such that, for each fixed context $f \in \mathcal{H}_X$, the map $\Phi_\theta^f(\,\cdot\,) \triangleq \Phi_\theta(f, \cdot) : \mathcal{H}_Y \to \mathbb{R}$ is an energy functional on the target function space $\mathcal{H}_Y$.

Given a context function $f$, the energy scalar $\Phi_\theta(f, u)$ scores a candidate $u \in \mathcal{H}_Y$ representing the energy level of the target function given a fixed context function. Ideally, an energy scalar should be low for likely function pairs while high for rare function cases.

### 2.2. Score Functions and Reference Measures

We define a target measure on the function space $\mathcal{H}_Y$ via a log Radon–Nikodym derivative with respect to a Gaussian reference. Let $\mu_0 = \mathcal{N}(0, C)$ be a Gaussian random field on $\mathcal{H}_Y$ with covariance operator $C : \mathcal{H}_Y \to \mathcal{H}_Y$ (self-adjoint, positive, trace-class in typical settings). Let $\mathcal{H}_{\mu_0} = C^{1/2}(\mathcal{H}_Y)$ denote the Cameron–Martin space of $\mu_0$. Given $f \in \mathcal{H}_X$, we define a conditional measure on $\mathcal{H}_Y$ by

$$\frac{d\nu_\theta(\cdot \mid f)}{d\mu_0}(u) \;=\; \frac{1}{Z_\theta(f)} \exp\big(-\Phi_\theta(f, u)\big), \tag{2}$$

where $Z_\theta(f) = \int_{\mathcal{H}_Y} \exp\big(-\Phi_\theta(f, u)\big)\, d\mu_0(u)$ is a normalizing constant (when it exists).

**Assumption 2.2** (Cameron–Martin differentiability)**.** For each $f \in \mathcal{H}_X$, the map $u \mapsto \Phi_\theta(f, u)$ is Fréchet differentiable along $\mathcal{H}_{\mu_0}$, and $D^u_{\mathcal{H}_{\mu_0}} \Phi_\theta(f, u) \in \mathcal{H}^*_{\mu_0}$.

Let $R : \mathcal{H}^*_{\mu_0} \to \mathcal{H}_{\mu_0}$ be the Riesz map induced by the $\mathcal{H}_{\mu_0}$ inner product. We define the *score* (Riesz representative) by:

$$S_\theta(f; u) \triangleq -R\, D^u_{\mathcal{H}_{\mu_0}} \Phi_\theta(f, u) \in \mathcal{H}_{\mu_0} \subset \mathcal{H}_Y. \tag{3}$$

In practice, $\Phi_\theta(f, u)$ is computed from a discretized representation $u \in \mathbb{R}^n$, where $n$ is the number of spatial discretization points, and automatic differentiation yields the Euclidean gradient $\nabla_u \Phi_\theta \in \mathbb{R}^n$. The score field in iterative energy minimization is then $S_\theta(f; u) = -C \nabla_u \Phi_\theta(f, u)$, where $C \in \mathbb{R}^{n \times n}$ is the discretized covariance matrix of the reference GRF. In the continuum limit, this corresponds to applying the covariance operator $C$ to the $L^2$-gradient of the energy, yielding a score in the Cameron-Martin space of the reference measure.

### 2.3. Training with Iterative Energy Minimization

We assume supervised training data $(f, u^*) \sim \mathcal{D}$, where $f \in \mathcal{H}_X$ and $u^* \in \mathcal{H}_Y$. Let $U^K_\theta(f, \omega) \in \mathcal{H}_Y$ denote the

[2]Typical choices include $L^2$-type spaces or Sobolev spaces $H^s(\Omega_X; \mathbb{R}^d)$ and $H^t(\Omega_Y; \mathbb{R}^m)$.

function after $K$ minimization steps for context $f$, where $\omega$ collects all randomness. We introduce a general *task loss functional*

$$\mathcal{J} : \mathcal{H}_Y \times \mathcal{H}_Y \times \mathcal{H}_X \to \mathbb{R}_{\geq 0}, \quad (u, u^*; f) \mapsto \mathcal{J}(u, u^*; f). \tag{4}$$

This allows modeling discrepancies with application-specific functionals[3].

**Definition 2.3** (Sampling-based training objective)**.** We define the training objective as loss $\mathcal{J}$

$$\mathcal{L}(\theta) \triangleq \mathbb{E}_{(f,u^*)\sim\mathcal{D}}\, \mathbb{E}_{\omega}\Big[\mathcal{J}\big(U_\theta^K(f,\omega), u^*; f\big)\Big]. \tag{5}$$

This definition is similar to the loss defined in EBT (Gladstone et al., 2025), which introduces a sampling-based loss for energy-based models. In our setting, the randomness in $\omega$ comes from the GRF initialization $U_\theta^{(0)} \sim \mathcal{N}(0, C_N)$, while the subsequent inference steps are deterministic GRF-preconditioned energy-minimization steps. The update map in Algorithm 1 is $u \mapsto u + hS_\theta(f; u)$. If one backpropagates through all minimization steps, gradients $\nabla_\theta \mathcal{L}(\theta)$ may involve Hessian-vector products of $\Phi_\theta$. This can be computed by modern autodiff frameworks, but may increase memory and runtime. In practice, we detach intermediate states as in (Gladstone et al., 2025), so that training does not require full backpropagation through time. Appendix B.3 shows that, under standard smoothness assumptions and sufficiently small step size, the deterministic GRF-preconditioned update monotonically decreases the discretized energy.

### 2.4. Operator Architecture for the Energy Functional

We specify the architecture for $\Phi_\theta(f, u)$. Let $\{x_i\}_{i=1}^{M_X} \subset \Omega_X$ and $\{y_j\}_{j=1}^{M_Y} \subset \Omega_Y$ be sampling points with quadrature weights $\{w_i^X\}$ and $\{w_j^Y\}$. We represent $f_i \triangleq f(x_i) \in \mathbb{R}^d$ and $u_j \triangleq u(y_j) \in \mathbb{R}^m$ as a discretization of both input and output functions and include positional encoding $\gamma_X(x_i) \in \mathbb{R}^{d_\gamma}$ and $\gamma_Y(y_j) \in \mathbb{R}^{d_\gamma}$ such as SIREN encoding (Sitzmann et al., 2020). EBO encodes input function and output function with various neural operator architectures such as transformer (Vaswani et al., 2017) and Transolver (Wu et al., 2024). We then use projection layers to return a 1-channel energy evaluation per point, integrated using a mean operator (equivalent to the integral in continuum) as the final estimation of energy functionals. We want to emphasize that our paradigm is independent of the underlying architectures.

[3]For our paper, we assume $L_2$ functional as our default task loss functional.

## 3. Related Works

**Neural operators.** Neural operators (Kovachki et al., 2023) are architectures respecting discretization invariance and can efficiently learn mappings between infinite-dimensional function spaces. Transformer (Vaswani et al., 2017) architecture can be extended to proper neural operators with examples such as GNOT (Hao et al., 2023), UPT (Alkin et al., 2024), and Transolvers (Wu et al., 2024). On another hands, there have been diverse attention mechanisms such as Galerkin (Cao, 2021) and FactFormer (Li et al., 2023) for improving naive attention mechanisms in PDE modeling. Unlike other methods, EBO is agnostic to underlying operator architecture.

**Energy-based models.** Energy-based models have been explored in various forms (Song & Kingma, 2021). Energy transformer (Hoover et al., 2023) was proposed to generalize transformers into energy-based paradigms. Energy-based transformer (Gladstone et al., 2025) succeeded in scaling general energy-based models into language models and video models. However, since they are all trained with regular networks, they cannot adapt to irregular grids and do not enjoy discretization invariance in comparison with neural operators. There are several works focusing on function-space diffusion models (Lim et al., 2025; Wang et al., 2025; Franzese et al., 2023; Zhang & Wonka, 2024), which are intrinsically energy-based operators as they are training the derivative of the log of the energy function. However, since they are not directly learning the energy functional itself, it is not possible to make estimation of energy level directly with those methods. Zhang et al. (2025) and Oh et al. (2024) both introduce energy-base models, and EBO generalizes this construction to a backbone-agnostic framework with explicit discretization-invariance (Theorem 4.1) and universal-approximation (Theorem 4.2) guarantees.

## 4. Theoretical Analysis

In this section, we establish two fundamental theoretical properties of Energy-Based Operators: (i) *discretization invariance*, ensuring that our energy estimates converge to well-defined continuum limits despite discretization choices, and (ii) *universal approximation*, demonstrating that EBOs can approximate any continuous energy functional to arbitrary precision.

### 4.1. Discretization Invariance

A central motivation for EBOs is robustness to changes in discretization. Unlike naive grid-based approaches that relearn models for each resolution, an EBO trained on one resolution should generalize to others. We formalize this through the relationship between continuum and discretized energy functionals under the architectural assumptions de-

*Table 1.* **Across architectures, EBO is the best method on smooth signals (Cosine, Damping); Operator is best on the non-smooth Izhikevich regime.** Test MSE ± std on B=128 held-out samples at resolution $N = 128$ with 60% context, in both super-resolution (SR) and forecasting (FC) settings. Bold marks the best method within each (backbone, dataset, mask) cell.

| | | Cosine | | Damping | | Izhikevich | |
|---|---|---|---|---|---|---|---|
| **Backbone** | **Method** | SR | FC | SR | FC | SR | FC |
| Transformer | Op. | 8.83e-04±5.98e-04 | 1.59e-03±1.58e-03 | 1.47e-03±1.26e-03 | 1.00e-03±1.15e-03 | **1.99e-02±2.33e-02** | **3.85e-02±4.12e-02** |
| | EBO | **2.75e-04±1.87e-04** | **1.42e-03±2.34e-03** | **2.54e-04±1.80e-04** | **3.22e-04±3.89e-04** | 3.21e-02±3.01e-02 | 4.81e-02±4.40e-02 |
| | DDO | 9.76e-03±2.42e-02 | 2.02e-02±7.72e-02 | 1.39e-01±7.88e-02 | 5.09e-03±7.89e-03 | 1.78e-01±1.35e-01 | 9.55e-02±1.05e-01 |
| Factformer | Op. | 1.43e-03±1.39e-03 | 1.90e-03±1.73e-03 | 1.65e-03±1.93e-03 | 7.11e-04±1.02e-03 | **1.95e-02±2.33e-02** | **3.90e-02±4.09e-02** |
| | EBO | **2.71e-04±1.65e-04** | **1.03e-03±1.80e-03** | **4.16e-04±4.47e-04** | **3.73e-04±5.17e-04** | 3.24e-02±3.27e-02 | 4.57e-02±4.26e-02 |
| | DDO | 1.92e-02±4.95e-02 | 1.08e-02±2.37e-02 | 1.34e-01±7.83e-02 | 9.11e-03±2.00e-02 | 2.28e-01±1.11e-01 | 1.03e-01±8.26e-02 |
| Galerkin | Op. | 1.93e-03±2.00e-03 | 2.26e-03±2.36e-03 | 3.04e-03±3.22e-03 | 8.56e-04±1.14e-03 | **3.70e-02±4.31e-02** | **4.20e-02±4.30e-02** |
| | EBO | **4.66e-04±3.28e-04** | **1.65e-03±1.49e-03** | **4.63e-04±5.00e-04** | **3.23e-04±5.39e-04** | 4.19e-02±4.17e-02 | 4.51e-02±4.25e-02 |
| | DDO | 1.38e-02±3.75e-02 | 1.25e-02±2.86e-02 | 1.42e-01±7.81e-02 | 1.54e-02±3.12e-02 | 1.84e-01±1.17e-01 | 1.11e-01±8.37e-02 |

tailed in Appendix B.

**Theorem 4.1** (Discretization invariance of EBO)**.** *Let $\Phi_\theta$ be an EBO as in Definition B.5 with continuous kernels $\Psi_t \in C^p(\Omega_Y \times \Omega_X, \mathbb{R}^{d_h \times d_h})$, Lipschitz activations with constant $L_\sigma$, and a Lipschitz energy head with constant $L_\psi$. Assume that Assumptions B.1 and B.2 hold on a compact set $K \subset \mathcal{H}_X \times \mathcal{H}_Y$. Let $\mathcal{D}_{M_X}, \mathcal{D}_{M_Y}$ be discrete refinements of $\Omega_X, \Omega_Y$ with quadrature rules of order $p \geq 1$. Then the discretized EBO $\Phi_\theta^{M_X,M_Y}$ of Definition B.6 satisfies*

$$\sup_{(f,u)\in K} \left|\Phi_\theta^{M_X,M_Y}(f,u) - \Phi_\theta(f,u)\right| = O\Big(M_X^{-p/n_x} + M_Y^{-p/n_y}\Big), \qquad M_X, M_Y \to \infty. \quad (6)$$

The proof decomposes the error into layer-wise quadrature error (via Lemma B.8) and error propagation through Lipschitz-stable layers (Lemma B.9). Lemma B.10 combines these bounds layer-by-layer, yielding $O(M_X^{-p/n_x})$ error in the hidden representations. Output-side pooling contributes $O(M_Y^{-p/n_y})$, and Lipschitz continuity of the energy head returns the scalar energy. Full details appear in Appendix B.

### 4.2. Universal Approximation

We next establish that EBOs are universal approximators of continuous energy functionals.

**Theorem 4.2** (Universal approximation for EBO)**.** *Let $\Phi^* : \mathcal{H}_X \times \mathcal{H}_Y \to \mathbb{R}$ be a continuous energy functional. For any compact set $K \subset \mathcal{H}_X \times \mathcal{H}_Y$ and any $\varepsilon > 0$, there exists an EBO $\Phi_\theta$ as in Definition B.5, with sufficiently large hidden dimension $d_h$ and number of layers $T$, such that*

$$\sup_{(f,u)\in K} \left|\Phi_\theta(f,u) - \Phi^*(f,u)\right| < \varepsilon. \quad (7)$$

The proof reduces the problem to finite-dimensional approximation via point evaluations (Lemma B.21), encodes measurements through integral pooling (Lemma B.22), and applies neural operator universality (Lemma B.23), building on (Chen & Chen, 1995) and (Kovachki et al., 2023). The final energy head is approximated via standard neural network universality (Assumption B.20). See Appendix B for complete details.

**Corollary 4.3** (Weak convergence of induced measures)**.** *Suppose $\Phi^*$ induces a well-defined conditional probability measure $\pi_f^*(\mathrm{d}u) \propto \exp\big(-\Phi^*(f,u)\big)\,\mathrm{d}\pi_0(u)$, where $\pi_0$ is a Gaussian reference measure on $\mathcal{H}_Y$. Under mild integrability assumptions on $\pi_0$, if $\Phi_\theta$ approximates $\Phi^*$ with error going to zero on compact sets, then the EBO-induced measure $\pi_{\theta,f}$ converges weakly to $\pi_f^*$.*

This ensures that the EBO-induced conditional measure converges to the target measure in the universal-approximation limit. In the main experiments, we use deterministic GRF-preconditioned energy minimization as a scalable refinement procedure rather than exact sampling from this measure.

## 5. Experiments

In this section, we conduct extensive experiments to empirically validate the performance of EBO. Since EBO is architecture-agnostic, we use a transformer neural operator (noted as operator), which takes bidirectional attention for embedding and a point-wise projection layer as a predictor as our key baseline, and adapt it minimally by intentionally adding an integral layer for scalar outputs.

We compare EBO and transformer operators across diverse function classes. In addition, we evaluate scalability and real-world applicability using seizure EEG and high-frequency financial data. For detailed setup of each task, please refer to Appendix C, E, and F.

### 5.1. Function-level Reconstruction and Generation

During training of EBO, we used various functions with different discretizations and sampling ratios over irregular

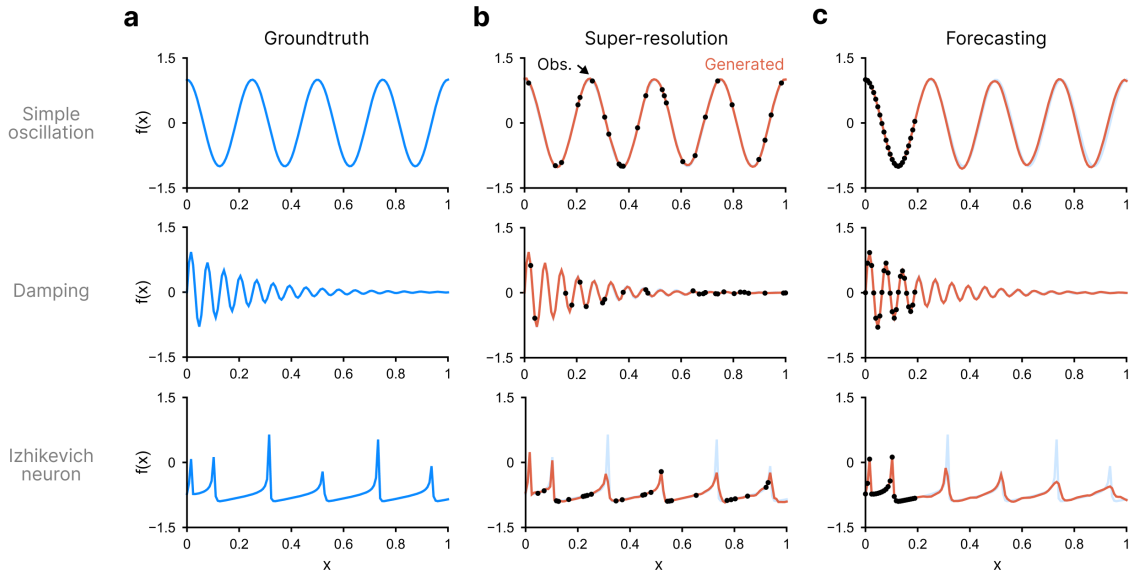


*Figure 2.* **Generation of various functions with limited context.** (**a**) Ground-truth functions. Three function classes are used: simple oscillation, damping, and Izhikevich neuron model. (**b**) Super-resolution experiment. Only sparse observations (20%) are given, and reconstruction to 128 samples is performed. Dots and orange lines indicate the given context and the generated functions by EBO, respectively. (**c**) Forecasting experiment. The first 20% sample points are given, and the model reconstructs the full future values.

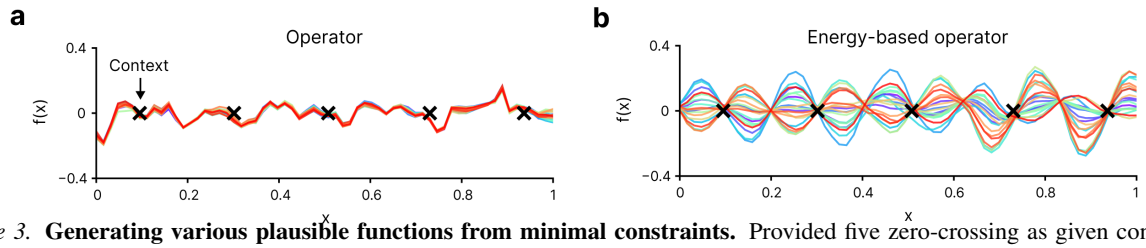


*Figure 3.* **Generating various plausible functions from minimal constraints.** Provided five zero-crossing as given contexts, (**a**) Generated functions through direct prediction (Operator). (**b**) Generated functions through EBO.

grids (Figure 1a). Due to discretization invariance, our model successfully maps functions to low energy regardless of sampling ratio, whereas anomalous functions are assigned high energy values in the energy landscape (Figure 1b).

Starting from initial GRF noise, we perform iterative EM to generate functions conditioned on a given context (Figure 1b,c). The refined candidate function's energy gradually decreases (Figure 1d). The norm of the energy gradient also converges toward zero, indicating that the iterative EM dynamics stabilize (Figure 1e). During this process, the gap between the generated function and the ground-truth function narrows (Figure 1f). Check Appendix C for details.

**Super-resolution and forecasting** We tested super-resolution and forecasting with 60% context ratio across three function classes (oscillation, damping, and Izhikevich; Figure 2a), using three backbones (Transformer (Vaswani et al., 2017), FactFormer (Li et al., 2023), and Galerkin Transformer (Cao, 2021)) with detailed setups in Appendix C. According to Table 1, EBO is the best method on smooth signals (oscillation and damping) for every backbone tested, improving over one-shot prediction (Op.) by up to $\sim 5\times$ in MSE, and is never beaten by the diffusion baseline (DDO). On the non-smooth Izhikevich regime, one-shot prediction (Op.) wins across all backbones — consistent with the GP prior in EBO being best matched to smooth functions. Even with limited context, EBO successfully captures dynamics such as periodicity (oscillation), decay (damping), and neural spike timing (Izhikevich) in both super-resolution (Figure 2b) and forecasting (Figure 2c).

**Function generation from sparse observations** To investigate whether EBO can generate various plausible functions given minimal constraints, we provided 5 uniformly spaced zero-crossing context points and performed super-resolution to predict the full function. While direct predictions fail to capture the global periodicity of cosine functions (Figure 3a), EBO successfully generates diverse cosine-like functions (Figure 3b). Importantly, EBO demonstrates strong diversity, producing varied functions that satisfy the constraints, whereas predictors exhibit mode collapse. These results highlight EBO's capacity to learn probabilistic energy distributions.

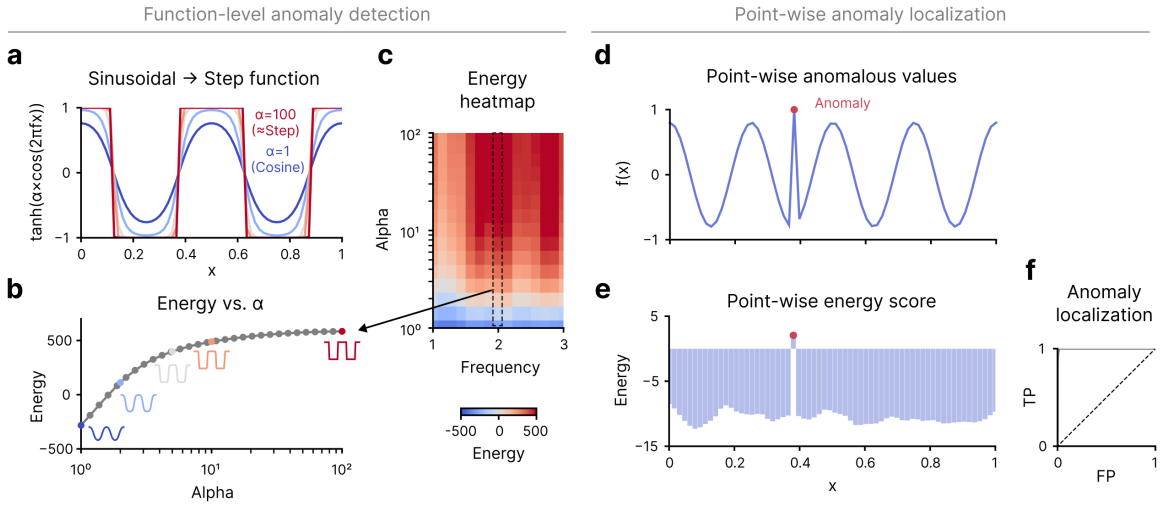


*Figure 4.* **Function-level anomaly detection and anomaly localization using energy.** (**a**-**c**) Function-level anomaly detection. (**a**) Synthesized function space transition from sinusoidal function (learned) to step function (anomaly). (**b**) Energy vs. $\alpha$. $\alpha = 1$ indicates a perfect cosine function, but increasing $\alpha$ modulates the waveform into a radical transition, finally converging to a step function. (**c**) Generalized energy heatmap with varying $\alpha$ and frequency of sinusoidal function. (**d**-**f**) Point-wise energy score for anomaly localization. (**d**) Sinusoidal function with point-wise anomaly. (**e**) Point-wise energy score. (**f**) Anomaly localization task performance.

**Function-level anomaly detection** We use total energy as an OOD score. Test functions $f_\alpha(x) = \tanh(\alpha \cdot \cos(2\pi\omega x))$ transition from smooth cosine ($\alpha \approx 1$) to square wave ($\alpha \gg 1$) (Figure 4a). Energy increases monotonically with $\alpha$, enabling energy-score-based anomaly detection (Figure 4b). We built an energy heatmap, where the x-axis indicates diverse cosine functions with different frequencies and the y-axis indicates $\alpha$. We observed that energy generally increases as $\alpha$ increases but remains invariant to cosine frequency, as EBO is trained only on the cosine function class (Figure 4c). These results suggest that the energy score, trained without supervision, can capture the general energy landscape of defined function classes.

**Point-wise anomaly detection** To track not only whether a function is anomalous but also where the anomaly occurs, we leverage per-point energy $E_i(f)$ for anomaly localization. After injecting a discontinuity at a single point of a cosine function instance, the anomalous location consistently exhibits the highest energy (Figure 4a). We tested various random anomalous locations and confirmed that simple energy thresholding is sufficient for anomaly localization (Figure 4b). These results empirically validate that EBO can perform both detection and spatial localization.

**Test-time scaling** We make comparisons of different steps (either optimization or denoising step) during test-time for Operator, EBO and DDO. In Figure 5a, naive operator has fixed performance regardless of the steps since it is a deterministic operator. In contrast, DDO and EBO improve with the number of steps; however, EBO achieves stronger performance than direct prediction after 8 steps. This enables test-time scaling by adapting EM steps based on computation budget.

**Sparse reconstruction of PDE solution fields** At resolution 128, EBO reduces missing-region relative $L^2$ on 10 of the 12 reported PDE settings in Table 2. The strongest improvements occur on Darcy flow, where EBO lowers error from $0.108$ to $0.026$ at 20% context and from $0.082$ to $0.020$ at 40% context, corresponding to roughly $4\times$ lower error in both cases. Navier–Stokes remains harder, but EBO still improves the operator baseline by $43.2\%$ at 20% context and $41.1\%$ at 40% context. Burgers shows the main limitation: EBO improves 20% and 40% context, but at 60% and 80% context the corrected operator is lower-error, suggesting that high-context 1D reconstruction can be solved effectively by the supervised operator once context consistency is enforced. Moreover, Figure 5b shows that EBO improves high frequency accuracy with increasing EM steps, demonstrating its ability to recover high frequency information. Check Appendix D for detailed setups.

### 5.2. Function-level Anomaly in Real-World Tasks

**Epilepsy seizure detection.** We evaluate EBO on the Temple University Seizure Corpus (TUSZ) (Shah et al., 2018; Obeid & Picone, 2016), a large-scale annotated EEG dataset for seizure detection. We use the 22-channel TCP bipolar montage and segment recordings into 12-second windows (Figure 7a; see Appendix E for dataset details and preprocessing). Notably, the EEG datasets are recorded at various sampling frequencies due to measurement conditions (Figure 7b). Although conventional approaches interpolate or

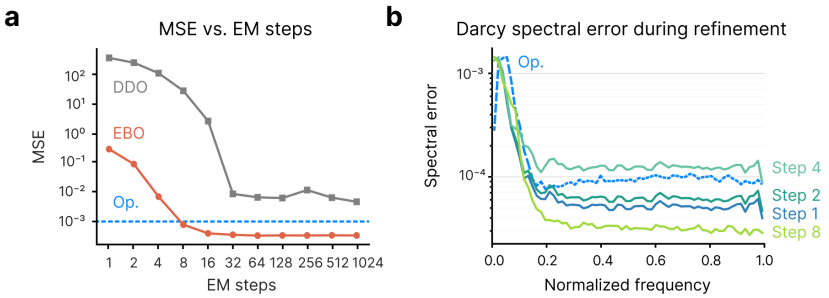


*Figure 5.* **EBO inference dynamics over iterative EM steps.** (a) For $N{=}128$ over 60% context, EBO crosses Operator at $\sim$8 steps and converges to $\approx 3.3 \times 10^{-4}$ by 32 steps. In contrast, DDO and EBO improve with the number of steps. However, DDO saturates above the operator baseline, whereas EBO surpasses direct prediction after approximately 8 steps. (b) For Darcy flow at 20% context, EBO reduces missing-region relative $L^2$ from 0.108 to 0.026 in the selected setting with improvements via iterative EM steps over high frequency.

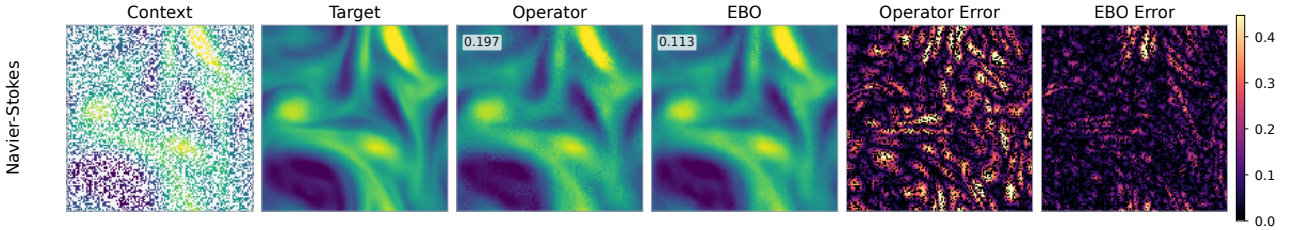


*Figure 6.* **Navier–Stokes sparse reconstruction.** At 40% context, EBO reduces missing-region relative $L^2$ from 0.197 to 0.113, a 42.6% reduction. The qualitative comparison shows that energy refinement reduces structured missing-region artifacts while preserving the observed context.

*Table 2.* **Sparse PDE reconstruction at resolution 128**. We report missing-region relative $L^2$ error, evaluating only unobserved locations. EBO improves all Darcy and Navier–Stokes settings and the lower-context Burgers settings.

| PDE | Context | Op. ↓ | EBO ↓ |
|---|---|---|---|
| Darcy | 20% | 0.108 | **0.026** |
| | 40% | 0.082 | **0.020** |
| | 60% | 0.055 | **0.019** |
| | 80% | 0.042 | **0.020** |
| Burgers | 20% | 0.175 | **0.062** |
| | 40% | 0.074 | **0.042** |
| | 60% | **0.037** | 0.048 |
| | 80% | **0.032** | 0.044 |
| Navier–Stokes | 20% | 0.220 | **0.125** |
| | 40% | 0.192 | **0.113** |
| | 60% | 0.179 | **0.116** |
| | 80% | 0.169 | **0.147** |

*Table 3.* **Anomaly detection performance metrics across resolution.** EBO improves AUROC and AUPRC across all three anomaly tasks, with the largest gains on seizure detection.

| Anomaly | Metric | Op. ↑ | EBO ↑ |
|---|---|---|---|
| Cosine → Step | AUROC | 0.9685 | **0.9809** |
| | AUPRC | 0.9701 | **0.9827** |
| | Det. ($\alpha$=0.05) | 0.8750 | **0.9150** |
| | Det. ($\alpha$=0.10) | 0.8750 | **0.8975** |
| High volatility | AUROC | 0.9783 | **0.9860** |
| | AUPRC | 0.9670 | **0.9710** |
| | Det. ($\alpha$=0.01) | 0.6503 | **0.8703** |
| | Det. ($\alpha$=0.05) | 0.9410 | **0.9610** |
| Seizure | AUROC | 0.7088 | **0.8480** |
| | AUPRC | 0.8081 | **0.8859** |
| | Det. ($\alpha$=0.05) | 0.4887 | **0.7375** |
| | Det. ($\alpha$=0.10) | 0.7212 | **0.7388** |

downsample the raw signal to match regular grids, we used raw signals to train EBO by leveraging discretization invariance. The model is trained to learn neural dynamics on background (normal) EEG segments and evaluated on its ability to distinguish seizure events via energy-based scoring (Figure 7c). Remarkably, the energy measured on seizures is significantly higher than on normal EEG (Figure 7c). In addition, energy scores for generalized seizure EEG are significantly higher than for focal seizure, indicating that energy does not merely reflect the presence of an anomaly but provides meaningful information regarding the degree of anomaly (Figure 7c). Overall, our model, trained to learn neural dynamics, is capable of distinguishing function-level anomalies such as seizures even without supervision (Figure 7d, AUROC $= 0.848$).

**Financial volatility regime shift detection.** We apply EBO to limit order book (LOB) data from high-frequency financial markets to learn market microstructure dynamics (Figure 8a; we used custom-collected LOB data from the Korea futures market between July 2, 2024 – Dec 12, 2024 for training and Jan 17, 2025 – Mar 13, 2025 for testing. See Appendix F for dataset details and preprocessing). LOB

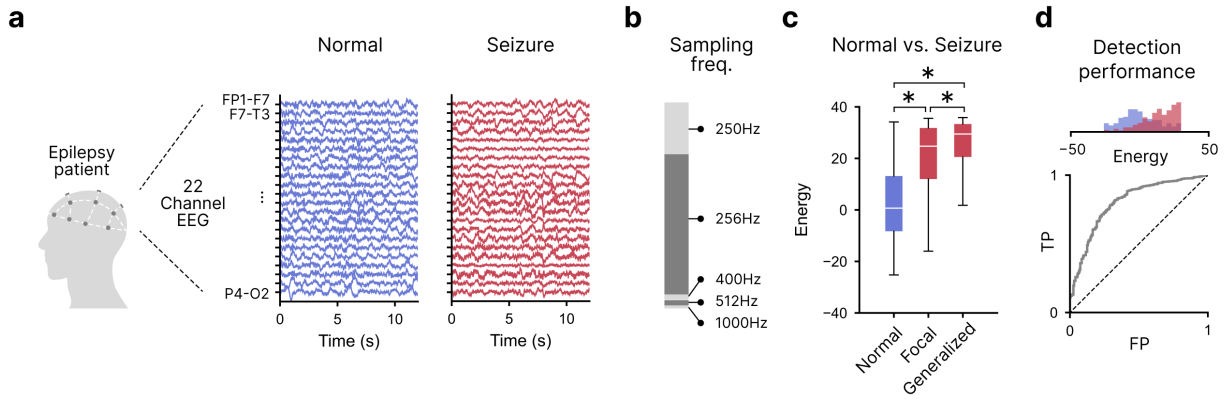


*Figure 7.* **Seizure as function level anomaly on electroencephalogram trained model.** (**a**) Electroencephalogram (EEG) measured on epilepsy patients using the 10-20 system and TCP bipolar montage. Example shows a 22-channel EEG signal during normal and seizure states. (**b**) Diversity of sampling frequency. The EEG corpus intrinsically contains 250 Hz, 256 Hz, 400 Hz, 512 Hz, and 1000 Hz due to medical measurement setup. (**c**) Energy of normal and seizure states. Two types of seizure are used: focal seizure and generalized seizure. (**d**) Seizure detection performance. Energy distribution (top) and receiver operating characteristic curve (bottom).

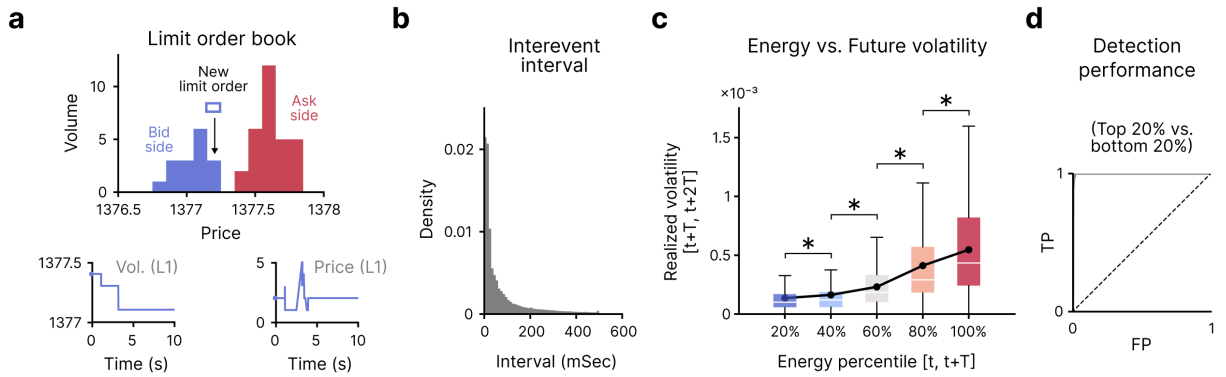


*Figure 8.* **Energy predicts future volatility.** (**a**) Example snapshot of a limit order book (left). Red indicates ask orders and blue indicates bid orders, with cumulative volume at each price level. The volume and price at each level vary over time due to market events such as new limit orders, market orders, and cancellations (right). The data were collected from the Korea Exchange (KRX 106V9) on July 3, 2024, at 10:09:14.08. (**b**) Interevent intervals. (**c**) Energy vs. future volatility. Current energy (computed from data in [t, t+T]) is binned by percentile, and future realized volatility (computed from data in [t+T, t+2T]) is measured for each bin.

contains raw-level market events related to orders and executions, providing rich information to model dynamics across market participants (Gould et al., 2013). Previous literature leverages LOB data to predict toxic market flow, specifically anomalous market behavior such as volatility regime shifts or market crashes (Easley et al., 2011; 2012). We evaluated the capacity of EBO trained on LOB data to detect regime shifts in volatility. Notably, since LOB is a collection of market events whose arrivals follow a Poisson process, the limit order book time series inherently lies on an irregular temporal grid (Figure 8b). EBO's discretization invariance naturally encodes 10-second windows containing a variable number of market events. After learning the energy landscape of market dynamics, we compute energy with respect to various LOB snapshots. Notably, we confirmed that energy computed from current LOB snapshots is positively correlated with future realized volatility (Spearman's $\rho = 0.563$). As current energy increases, the corresponding future volatility also increases (Figure 8c). These results suggest that EBO provides meaningful energy scores that reflect market abnormalities leading to volatility regime transitions.

## 6. Conclusion

We introduce Energy-Based Operators (EBO), a framework for learning probability distributions over function spaces using energy functionals. By combining neural operators with energy-based models and iterative energy minimization, EBO provides a flexible generative framework in function space. We prove discretization invariance and universal approximation of continuous energy functionals. Empirically, EBO achieves strong results in function reconstruction and generation, and its energy score serves as a principled, domain-agnostic anomaly metric, enabling effective detection in tasks such as EEG-based seizure identification and financial volatility regime prediction without anomaly-specific supervision.

## References


Ackley, D. H., Hinton, G. E., and Sejnowski, T. J. A learning algorithm for boltzmann machines. *Cognitive science*, 9 (1):147–169, 1985.

Alkin, B., Fürst, A., Schmid, S., Gruber, L., Holzleitner, M., and Brandstetter, J. Universal physics transformers: A framework for efficiently scaling neural operators. *Advances in Neural Information Processing Systems*, 37: 25152–25194, 2024.

Berner, J., Liu-Schiaffini, M., Kossaifi, J., Duruisseaux, V., Bonev, B., Azizzadenesheli, K., and Anandkumar, A. Principled approaches for extending neural architectures to function spaces for operator learning. *arXiv preprint arXiv:2506.10973*, 2025.

Cao, S. Choose a transformer: Fourier or galerkin. *Advances in neural information processing systems*, 34: 24924–24940, 2021.

Chen, T. and Chen, H. Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems. *IEEE Transactions on Neural Networks*, 6(4):911–917, 1995.

Cybenko, G. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2(4):303–314, 1989.

Du, Y. and Mordatch, I. Implicit generation and modeling with energy based models. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Du, Y., Li, S., Tenenbaum, J., and Mordatch, I. Learning iterative reasoning through energy minimization. In *International Conference on Machine Learning*, pp. 5570–5582. PMLR, 2022.

Easley, D., De Prado, M. M. L., and O'Hara, M. The microstructure of the" flash crash": Flow toxicity, liquidity crashes, and the probability of informed trading. *Journal of Portfolio Management*, 37(2):118, 2011.

Easley, D., López de Prado, M. M., and O'Hara, M. Flow toxicity and liquidity in a high-frequency world. *The Review of Financial Studies*, 25(5):1457–1493, 2012.

Franzese, G., Corallo, G., Rossi, S., Heinonen, M., Filippone, M., and Michiardi, P. Continuous-time functional diffusion processes. *Advances in Neural Information Processing Systems*, 36:37370–37400, 2023.

Gladstone, A., Nanduru, G., Islam, M. M., Han, P., Ha, H., Chadha, A., Du, Y., Ji, H., Li, J., and Iqbal, T. Energy-based transformers are scalable learners and thinkers. *arXiv preprint arXiv:2507.02092*, 2025.

Gould, M. D., Porter, M. A., Williams, S., McDonald, M., Fenn, D. J., and Howison, S. D. Limit order books. *Quantitative Finance*, 13(11):1709–1742, 2013.

Hao, Z., Wang, Z., Su, H., Ying, C., Dong, Y., Liu, S., Cheng, Z., Song, J., and Zhu, J. Gnot: A general neural operator transformer for operator learning. In *International Conference on Machine Learning*, pp. 12556–12569. PMLR, 2023.

Hoover, B., Liang, Y., Pham, B., Panda, R., Strobelt, H., Chau, D. H., Zaki, M., and Krotov, D. Energy transformer. *Advances in neural information processing systems*, 36: 27532–27559, 2023.

Hopfield, J. J. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558, 1982.

Hornik, K., Stinchcombe, M., and White, H. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989.

Izhikevich, E. M. Simple model of spiking neurons. *IEEE Transactions on neural networks*, 14(6):1569–1572, 2003.

Kovachki, N., Li, Z., Liu, B., Azizzadenesheli, K., Bhattacharya, K., Stuart, A., and Anandkumar, A. Neural operator: Learning maps between function spaces with applications to pdes. *Journal of Machine Learning Research*, 24(89):1–97, 2023.

LeCun, Y., Chopra, S., Hadsell, R., Ranzato, M., Huang, F., et al. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006.

Li, Z., Shu, D., and Barati Farimani, A. Scalable transformer for pde surrogate modeling. *Advances in Neural Information Processing Systems*, 36:28010–28039, 2023.

Lim, J. H., Kovachki, N. B., Baptista, R., Beckham, C., Azizzadenesheli, K., Kossaifi, J., Voleti, V., Song, J., Kreis, K., Kautz, J., et al. Score-based diffusion models in function space. *Journal of Machine Learning Research*, 26(158):1–62, 2025.

Oarga, A. and Du, Y. Generalizable reasoning through compositional energy minimization. *arXiv preprint arXiv:2510.20607*, 2025.

Obeid, I. and Picone, J. The temple university hospital eeg data corpus. *Frontiers in neuroscience*, 10:196, 2016.

Oh, T. I., Nam, H. C., Park, C., and Cho, H. Funcanode: A function level anomaly detection in device simulation. In *2024 International Conference on Simulation of Semiconductor Processes and Devices (SISPAD)*, pp. 01–04. IEEE, 2024.

Rahimi, A. and Recht, B. Random features for large-scale kernel machines. In Platt, J., Koller, D., Singer, Y., and Roweis, S. (eds.), *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007.

Shah, V., Von Weltin, E., Lopez, S., McHugh, J. R., Veloso, L., Golmohammadi, M., Obeid, I., and Picone, J. The temple university hospital seizure detection corpus. *Frontiers in neuroinformatics*, 12:83, 2018.

Sitzmann, V., Martel, J., Bergman, A., Lindell, D., and Wetzstein, G. Implicit neural representations with periodic activation functions. *Advances in neural information processing systems*, 33:7462–7473, 2020.

Song, Y. and Kingma, D. P. How to train your energy-based models. *arXiv preprint arXiv:2101.03288*, 2021.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Wang, R. and Du, Y. Equilibrium matching: Generative modeling with implicit energy-based models. *arXiv preprint arXiv:2510.02300*, 2025.

Wang, S., Dou, Z., Shan, S., Liu, T.-R., and Lu, L. Fundiff: Diffusion models over function spaces for physics-informed generative modeling. *arXiv preprint arXiv:2506.07902*, 2025.

Wu, H., Luo, H., Wang, H., Wang, J., and Long, M. Transolver: A fast transformer solver for pdes on general geometries. In *International Conference on Machine Learning*, 2024.

Zhang, B. and Wonka, P. Functional diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4723–4732, June 2024.

Zhang, Q., Krotov, D., and Karniadakis, G. E. Operator learning for reconstructing flow fields from sparse measurements: an energy transformer approach. *Journal of Computational Physics*, pp. 114148, 2025.

# A. Random Fourier Features for Gaussian Random Field Sampling

While FFT can be applied to efficiently sample Gaussian random fields with regular grids, it is not applicable to dense and irregular fields. Sampling from a Gaussian Random Field (GRF) with covariance kernel $K$ at $n$ arbitrary locations traditionally requires $\mathcal{O}(n^3)$ computation for Cholesky decomposition. We adopt Random Fourier Features (RFF) (Rahimi & Recht, 2007) to reduce this to $\mathcal{O}(nD)$, where $D$ is the number of features.

**Background.** For a shift-invariant kernel $K(x, y) = k(x - y)$, Bochner's theorem guarantees a spectral density $p(\omega)$ such that:

$$k(x-y) = \int p(\omega) e^{i\omega^\top (x-y)} d\omega = \mathbb{E}_{\omega\sim p}\left[\cos(\omega^\top x + b)\cos(\omega^\top y + b)\right] \tag{8}$$

where $b \sim \text{Uniform}(0, 2\pi)$. For the RBF kernel $K(x, y) = \sigma^2 \exp\left(-\frac{\|x-y\|^2}{2\ell^2}\right)$, the spectral density is $p(\omega) = \mathcal{N}(0, \ell^{-2} I)$.

**Method.** We approximate the kernel via Monte Carlo sampling:

$$K(x, y) \approx \phi(x)^\top \phi(y), \quad \text{where} \quad \phi(x) = \sigma\sqrt{\frac{2}{D}}\cos(Wx + b) \tag{9}$$

with $W \in \mathbb{R}^{D\times d}$ having rows $\omega_j \sim \mathcal{N}(0, \ell^{-2} I)$ and $b \in \mathbb{R}^D$ with entries $b_j \sim \text{Uniform}(0, 2\pi)$. As a result, we apply the kernel trick to sample from approximate Gaussian random fields.

To sample $f \sim \mathcal{N}(0, C)$ at coordinates $X = \{x_1, \ldots, x_n\}$, we compute:

$$f(X) = \Phi w, \quad w \sim \mathcal{N}(0, I_D) \tag{10}$$

where $\Phi \in \mathbb{R}^{n\times D}$ is the feature matrix with rows $\phi(x_i)^\top$.

**Convergence.** The approximation error satisfies $\sup_{x,y} |K(x, y) - \hat{K}(x, y)| = \mathcal{O}(D^{-1/2})$ with high probability (Rahimi & Recht, 2007). In practice, $D = 512$ suffices for kernel error below $5\%$.

**Advantages.** This approach offers three key benefits: (1) $\mathcal{O}(nD)$ complexity enables sampling at thousands of points, (2) no matrix factorization avoids numerical instability from ill-conditioned covariance matrices, and (3) arbitrary coordinates are handled naturally without grid assumptions.

# B. Theoretical Analysis

In this section, we establish two fundamental theoretical properties of Energy-Based Operators: (i) discretization invariance, ensuring that our energy estimates converge to well-defined continuum limits, and (ii) universal approximation, demonstrating that EBOs can approximate any continuous energy functional to arbitrary precision.

## B.1. Preliminaries and Notation

We first establish the necessary notation and assumptions, building on the notation introduced in Section 2. Recall that $\Omega_X \subset \mathbb{R}^{n_x}$ and $\Omega_Y \subset \mathbb{R}^{n_y}$ are bounded domains with Lebesgue measures $\lambda_X$ and $\lambda_Y$ (hence $\lambda_X(\Omega_X) < \infty$ and $\lambda_Y(\Omega_Y) < \infty$), and $\mathcal{H}_X$, $\mathcal{H}_Y$ are separable Hilbert spaces of functions on these domains.

**Additional notation for this section.** We introduce the following notation for the architecture specification:

- $d_h \in \mathbb{N}$: hidden representation dimension in the transformer layers.
- $T \in \mathbb{N}$: number of cross-attention layers (distinct from the minimization step count $K$).
- $P_X : \mathbb{R}^d \times \mathbb{R}^{n_x} \to \mathbb{R}^{d_h}$, $P_Y : \mathbb{R}^m \times \mathbb{R}^{n_y} \to \mathbb{R}^{d_h}$: lifting maps.
- $\kappa^{(t)} : \Omega_Y \times \Omega_X \to \mathbb{R}^{d_h \times d_h}$: cross-attention kernel at layer $t$.
- $\phi : \mathbb{R}^{d_h} \to \mathbb{R}$: energy head (output projection to scalar).

- $\mathcal{K}$: compact subsets of function spaces (to avoid confusion with minimization steps $K$).

**Assumption B.1** (Regularity of function spaces)**.** We assume:

1. $\mathcal{H}_X \hookrightarrow C(\overline{\Omega_X}; \mathbb{R}^d)$ and $\mathcal{H}_Y \hookrightarrow C(\overline{\Omega_Y}; \mathbb{R}^m)$ with continuous embeddings.
2. There exist constants $C_X, C_Y > 0$ such that $\|f\|_{C(\overline{\Omega_X})} \le C_X \|f\|_{\mathcal{H}_X}$ and $\|u\|_{C(\overline{\Omega_Y})} \le C_Y \|u\|_{\mathcal{H}_Y}$.

This assumption is satisfied, for instance, when $\mathcal{H}_X = H^s(\Omega_X; \mathbb{R}^d)$ with $s > n_x/2$ by the Sobolev embedding theorem.

**Assumption B.2** ($C^p$ regularity on $\mathcal{K}$)**.** Fix an integer $p \ge 1$ and a compact set $\mathcal{K} \subset \mathcal{H}_X \times \mathcal{H}_Y$. Assume that for all $(f, u) \in \mathcal{K}$:

1. The lifted input satisfies
$$h_f \in C^p(\overline{\Omega_X}; \mathbb{R}^{d_h}) \quad \text{with} \quad \|h_f\|_{C^p(\overline{\Omega_X})} \le C_{\mathcal{K}}.$$
2. The pooled integrand satisfies
$$\phi \circ v^{(T)} \in C^p(\overline{\Omega_Y}) \quad \text{with} \quad \|\phi \circ v^{(T)}\|_{C^p(\overline{\Omega_Y})} \le C_{\mathcal{K}}.$$

**Definition B.3** (Discrete refinement)**.** A **discrete refinement** of a bounded domain $\Omega \subset \mathbb{R}^n$ is a sequence of finite sets $(D_L)_{L=1}^{\infty}$ with $|D_L| = L$ such that for any $\epsilon > 0$, there exists $L(\epsilon) \in \mathbb{N}$ with

$$\Omega \subseteq \bigcup_{x \in D_L} B(x, \epsilon) \quad \text{for all } L \ge L(\epsilon), \tag{11}$$

where $B(x, \epsilon) = \{y \in \mathbb{R}^n : \|y - x\| < \epsilon\}$.

**Definition B.4** (Quadrature rule)**.** A **quadrature rule** on $\Omega$ associated with discretization $D_L = \{x_1, \dots, x_L\}$ is a set of weights $\{w_\ell\}_{\ell=1}^L \subset \mathbb{R}_+$ satisfying $\sum_{\ell=1}^L w_\ell = \lambda(\Omega)$. We say the quadrature rule has **order** $p$ if for all $g \in C^p(\overline{\Omega})$:

$$\left| \int_\Omega g(x)\, d\lambda(x) - \sum_{\ell=1}^L w_\ell g(x_\ell) \right| \le C_p \|g\|_{C^p} L^{-p/n}. \tag{12}$$

### B.2. Discretization Invariance

We now formalize and prove discretization invariance for EBOs.

**Definition B.5** (EBO Architecture - Continuum Form)**.** An Energy-Based Operator $\Phi_\theta : \mathcal{H}_X \times \mathcal{H}_Y \to \mathbb{R}$ has the following structure:

1. **Lifting:** Pointwise maps $P_X : \mathbb{R}^d \times \mathbb{R}^{n_x} \to \mathbb{R}^{d_h}$ and $P_Y : \mathbb{R}^m \times \mathbb{R}^{n_y} \to \mathbb{R}^{d_h}$ yield lifted representations:
$$h_f(x) = P_X(f(x), x), \quad h_u(y) = P_Y(u(y), y). \tag{13}$$
2. **Cross-Attention Layers:** For $t = 1, \dots, T$, integral kernel operators:
$$v^{(t)}(y) = \sigma\left( W^{(t)} v^{(t-1)}(y) + \int_{\Omega_X} \kappa^{(t)}(y, x) h_f(x)\, d\lambda_X(x) + b^{(t)}(y) \right), \tag{14}$$
where $v^{(0)}(y) = h_u(y)$, $\kappa^{(t)} \in C^p(\overline{\Omega_Y} \times \overline{\Omega_X}; \mathbb{R}^{d_h \times d_h})$ are continuous kernels of order $p \ge 1$, and $\sigma$ is a Lipschitz activation.
3. **Energy Head:** A continuous map $\phi : \mathbb{R}^{d_h} \to \mathbb{R}$ applied pointwise, followed by integration:
$$\Phi_\theta(f, u) = \int_{\Omega_Y} \phi(v^{(T)}(y))\, d\lambda_Y(y). \tag{15}$$

**Definition B.6** (Discretized EBO)**.** Given discretizations $D_{M_X} = \{x_i\}_{i=1}^{M_X} \subset \Omega_X$ and $D_{M_Y} = \{y_j\}_{j=1}^{M_Y} \subset \Omega_Y$ with quadrature weights $\{w_i^X\}$ and $\{w_j^Y\}$, the **discretized EBO** is:

1. **Discretized Cross-Attention:**

$$v_j^{(t)} = \sigma\left(W^{(t)} v_j^{(t-1)} + \sum_{i=1}^{M_X} w_i^X \kappa^{(t)}(y_j, x_i) h_f(x_i) + b^{(t)}(y_j)\right). \tag{16}$$

2. **Discretized Energy:**

$$\Phi_\theta^{(M_X, M_Y)}(f, u) = \sum_{j=1}^{M_Y} w_j^Y \phi(v_j^{(T)}). \tag{17}$$

**Definition B.7** (Discretization Invariance for Energy Functionals)**.** Let $\Theta \subseteq \mathbb{R}^{p_\theta}$ be a parameter space. A parametric family of energy functionals $\{\Phi_\theta\}_{\theta\in\Theta}$ is **discretization-invariant** if, for any discrete refinements $(D_{M_X})$ of $\Omega_X$ and $(D_{M_Y})$ of $\Omega_Y$ with associated quadrature rules of order $p \geq 1$, the following holds:

For any $\theta \in \Theta$ and any compact set $\mathcal{K} \subset \mathcal{H}_X \times \mathcal{H}_Y$:

$$\lim_{M_X, M_Y \to \infty} \sup_{(f,u)\in\mathcal{K}} \left|\Phi_\theta^{(M_X, M_Y)}(f, u) - \Phi_\theta(f, u)\right| = 0. \tag{18}$$

We now prove discretization invariance through a sequence of lemmas.

**Lemma B.8** (Quadrature Convergence for Integral Kernels)**.** *Let $\kappa : \Omega_Y \times \Omega_X \to \mathbb{R}^{d_h \times d_h}$ be continuous, and assume that for each $y$, the map $x \mapsto \kappa(y, x)$ is $C^p(\Omega_X)$. Let $h : \Omega_X \to \mathbb{R}^{d_h}$ be $C^p$-smooth in $x$. For a quadrature rule of order $p$ on $\Omega_X$ with $M_X$ points $\{x_i\}$ and weights $\{w_i^X\}$, one has:*

$$\sup_{y\in\Omega_Y} \left\| \int_{\Omega_X} \kappa(y, x) h(x)\, d\lambda_X(x) - \sum_{i=1}^{M_X} w_i^X\, \kappa(y, x_i) h(x_i) \right\| \;\leq\; C\, \|\kappa\|_{C^p(\overline{\Omega_Y}\times\overline{\Omega_X})}\, \|h\|_{C^p(\overline{\Omega_X})}\, M_X^{-p/n_x},$$

*where $C$ depends on $p$ and the domain $\Omega_X$.*

*Proof.* Fix $y \in \Omega_Y$ and define $g_y : \Omega_X \to \mathbb{R}^{d_h}$ by $g_y(x) = \kappa(y, x)h(x)$.

**Step 1:** We show $g_y$ inherits regularity from $\kappa$ and $h$. Since $\kappa$ is continuous on the compact set $\overline{\Omega_Y} \times \overline{\Omega_X}$, it is uniformly continuous. For any $\alpha$ with $|\alpha| \leq p$:

$$\|D_x^\alpha g_y\|_\infty \leq \sum_{|\beta|\leq|\alpha|} C_{\alpha,\beta} \|D_x^\beta \kappa(y, \cdot)\|_\infty \|D_x^{\alpha-\beta} h\|_\infty. \tag{19}$$

Thus $\|g_y\|_{C^p(\Omega_X)} \leq C_1 \|\kappa\|_{C^p} \|h\|_{C^p}$ uniformly in $y$.

**Step 2:** Apply the quadrature error bound componentwise. For each component $k \in \{1, \ldots, d_h\}$:

$$\left| \int_{\Omega_X} [g_y(x)]_k\, d\lambda_X(x) - \sum_{i=1}^{M_X} w_i^X [g_y(x_i)]_k \right| \leq C_p \|[g_y]_k\|_{C^p} M_X^{-p/n_x}. \tag{20}$$

**Step 3:** Combine components.

$$\left\| \int_{\Omega_X} \kappa(y, x) h(x)\, d\lambda_X(x) - \sum_{i=1}^{M_X} w_i^X \kappa(y, x_i) h(x_i) \right\| \tag{21}$$

$$\leq \sqrt{d_h} \max_k \left| \int_{\Omega_X} [g_y(x)]_k\, d\lambda_X - \sum_{i=1}^{M_X} w_i^X [g_y(x_i)]_k \right| \tag{22}$$

$$\leq \sqrt{d_h} C_p \|g_y\|_{C^p} M_X^{-p/n_x} \tag{23}$$

$$\leq C \|\kappa\|_{C^p} \|h\|_{C^p} M_X^{-p/n_x}, \tag{24}$$

where $C = \sqrt{d_h} C_p C_1$.

**Step 4:** The bound is uniform in $y$ since the constants depend only on $\|\kappa\|_{C^p(\Omega_Y \times \Omega_X)}$. □

**Lemma B.9** (Stability of Iterated Layers)**.** *Let $\sigma : \mathbb{R}^{d_h} \to \mathbb{R}^{d_h}$ be $L_\sigma$-Lipschitz. Define the layer map:*

$$\mathcal{L}[v, h](y) = \sigma\left(Wv(y) + \int_{\Omega_X} \kappa(y, x) h(x)\, d\lambda_X(x) + b(y)\right). \tag{25}$$

*If $v, \tilde{v} : \Omega_Y \to \mathbb{R}^{d_h}$ and $h, \tilde{h} : \Omega_X \to \mathbb{R}^{d_h}$ satisfy:*

$$\|v - \tilde{v}\|_\infty \le \delta_v, \quad \|h - \tilde{h}\|_\infty \le \delta_h, \tag{26}$$

*then:*

$$\|\mathcal{L}[v, h] - \mathcal{L}[\tilde{v}, \tilde{h}]\|_\infty \le L_\sigma \left(\|W\|_{\mathrm{op}} \delta_v + \|\kappa\|_\infty \lambda_X(\Omega_X) \delta_h\right). \tag{27}$$

*Proof.* For any $y \in \Omega_Y$:

$$\begin{aligned}
&\|\mathcal{L}[v, h](y) - \mathcal{L}[\tilde{v}, \tilde{h}](y)\| && (28)\\
&= \left\| \sigma\left(Wv(y) + \int_{\Omega_X} \kappa(y, x) h(x)\, d\lambda_X + b(y)\right) \right. && (29)\\
&\qquad \left. - \sigma\left(W\tilde{v}(y) + \int_{\Omega_X} \kappa(y, x) \tilde{h}(x)\, d\lambda_X + b(y)\right) \right\|. && (30)
\end{aligned}$$

By the Lipschitz property of $\sigma$:

$$\begin{aligned}
&\le L_\sigma \left\| W(v(y) - \tilde{v}(y)) + \int_{\Omega_X} \kappa(y, x)(h(x) - \tilde{h}(x))\, d\lambda_X(x) \right\| && (31)\\
&\le L_\sigma \left( \|W\|_{\mathrm{op}} \|v(y) - \tilde{v}(y)\| + \int_{\Omega_X} \|\kappa(y, x)\|_{\mathrm{op}} \|h(x) - \tilde{h}(x)\|\, d\lambda_X(x) \right) && (32)\\
&\le L_\sigma \left(\|W\|_{\mathrm{op}} \delta_v + \|\kappa\|_\infty \lambda_X(\Omega_X) \delta_h\right). && (33)
\end{aligned}$$

Taking supremum over $y \in \Omega_Y$ yields the result. □

**Lemma B.10** (Layer-wise Discretization Error)**.** *Let $v^{(t)}$ be the continuum representation at layer $t$ and $\tilde{v}^{(t)}$ be defined by:*

$$\tilde{v}^{(t)}(y) = \sigma\left(W^{(t)} \tilde{v}^{(t-1)}(y) + \sum_{i=1}^{M_X} w_i^X \kappa^{(t)}(y, x_i) h_f(x_i) + b^{(t)}(y)\right), \tag{34}$$

*with $\tilde{v}^{(0)} = v^{(0)} = h_u$. Then for quadrature of order $p$:*

$$\|v^{(t)} - \tilde{v}^{(t)}\|_\infty \le C_t M_X^{-p/n_x}, \tag{35}$$

*where $C_t$ depends on $t$, the layer parameters, and the uniform $C^p$ bound for $h_f$ from Assumption B.2.*

*Proof.* We proceed by induction on $t$.

**Base case ($t = 0$):** $v^{(0)} = \tilde{v}^{(0)} = h_u$, so the error is zero.

**Inductive step:** Assume $\|v^{(t-1)} - \tilde{v}^{(t-1)}\|_\infty \le C_{t-1} M_X^{-p/n_x}$.

Define the auxiliary function:

$$\hat{v}^{(t)}(y) = \sigma\left(W^{(t)} v^{(t-1)}(y) + \sum_{i=1}^{M_X} w_i^X \kappa^{(t)}(y, x_i) h_f(x_i) + b^{(t)}(y)\right). \tag{36}$$

**Step 1:** Bound $\|v^{(t)} - \hat{v}^{(t)}\|_\infty$ (quadrature error with exact previous layer).

By Lemma B.8 (with $h = h_f$) and Assumption B.2, we have

$$\|v^{(t)} - \hat{v}^{(t)}\|_\infty \le L_\sigma\, C_\kappa\, \|h_f\|_{C^p(\overline{\Omega_X})}\, M_X^{-p/n_x} \le L_\sigma\, C_\kappa\, C_{\mathcal{K}}\, M_X^{-p/n_x} =: A_t M_X^{-p/n_x}. \tag{37}$$

**Step 2:** Bound $\|\hat{v}^{(t)} - \tilde{v}^{(t)}\|_\infty$ (propagation of previous layer error).

Both $\hat{v}^{(t)}$ and $\tilde{v}^{(t)}$ use the same quadrature approximation for the integral. The only difference is in the previous layer: $v^{(t-1)}$ vs $\tilde{v}^{(t-1)}$.

For any $y \in \Omega_Y$:

$$\|\hat{v}^{(t)}(y) - \tilde{v}^{(t)}(y)\| \le L_\sigma \|W^{(t)}\|_{\mathrm{op}} \|v^{(t-1)}(y) - \tilde{v}^{(t-1)}(y)\| \tag{38}$$

$$\le L_\sigma \|W^{(t)}\|_{\mathrm{op}} C_{t-1} M_X^{-p/n_x} =: B_t C_{t-1} M_X^{-p/n_x}. \tag{39}$$

**Step 3:** Combine by triangle inequality.

$$\|v^{(t)} - \tilde{v}^{(t)}\|_\infty \le \|v^{(t)} - \hat{v}^{(t)}\|_\infty + \|\hat{v}^{(t)} - \tilde{v}^{(t)}\|_\infty \tag{40}$$

$$\le (A_t + B_t C_{t-1}) M_X^{-p/n_x} =: C_t M_X^{-p/n_x}. \tag{41}$$

This establishes the induction with $C_t = A_t + B_t C_{t-1}$ and $C_0 = 0$.

Unwinding the recursion: $C_t = \sum_{s=1}^{t} A_s \prod_{r=s+1}^{t} B_r$, which is finite for finite $t$. □

**Theorem B.11** (Discretization Invariance of EBO)**.** *Let $\Phi_\theta$ be an EBO as in Definition B.5 with:*

1. *Continuous kernels $\kappa^{(t)} \in C^p(\overline{\Omega_Y} \times \overline{\Omega_X}; \mathbb{R}^{d_h \times d_h})$.*
2. *Lipschitz activations $\sigma$ with constant $L_\sigma$.*
3. *Lipschitz energy head $\phi$ with constant $L_\phi$.*
4. *Lifting maps $P_X, P_Y$ that are Lipschitz in their first arguments.*

*Assume further that Assumption B.2 holds on the compact set $\mathcal{K} \subset \mathcal{H}_X \times \mathcal{H}_Y$. Let $(D_{M_X})$, $(D_{M_Y})$ be discrete refinements with quadrature rules of order $p \ge 1$.*

*Then for any compact $\mathcal{K} \subset \mathcal{H}_X \times \mathcal{H}_Y$:*

$$\sup_{(f,u)\in\mathcal{K}} \left|\Phi_\theta^{(M_X,M_Y)}(f,u) - \Phi_\theta(f,u)\right| = \mathcal{O}\left(M_X^{-p/n_x} + M_Y^{-p/n_y}\right) \quad \text{as } M_X, M_Y \to \infty. \tag{42}$$

*Proof.* Fix $(f,u) \in \mathcal{K}$. The proof proceeds in four steps.

**Step 1: Uniform bounds on compact sets.**

Since $\mathcal{K}$ is compact in $\mathcal{H}_X \times \mathcal{H}_Y$ and the embeddings into continuous functions are continuous (Assumption B.1), we have:

$$\sup_{(f,u)\in\mathcal{K}} \|f\|_{C(\overline{\Omega_X})} \le C_{\mathcal{K}}^X, \qquad \sup_{(f,u)\in\mathcal{K}} \|u\|_{C(\overline{\Omega_Y})} \le C_{\mathcal{K}}^Y. \tag{43}$$

The lifted representations satisfy:

$$\|h_f\|_\infty \le L_{P_X}(C_{\mathcal{K}}^X + \mathrm{diam}(\Omega_X)), \quad \|h_u\|_\infty \le L_{P_Y}(C_{\mathcal{K}}^Y + \mathrm{diam}(\Omega_Y)). \tag{44}$$

**Step 2: Discretization error in the attention layers.**

Let $v^{(T)}$ be the continuum output after $T$ layers and $\tilde{v}^{(T)}$ be defined by:

$$\tilde{v}^{(T)}(y_j) = v_j^{(T)} \quad \text{for } y_j \in D_{M_Y}, \tag{45}$$

extended continuously to all of $\Omega_Y$.

By Lemma B.10, at each quadrature point $y_j \in D_{M_Y}$:

$$\|v^{(T)}(y_j) - v_j^{(T)}\|_{\mathbb{R}^{d_h}} \le C_T M_X^{-p/n_x} \tag{46}$$

where $C_T$ depends on the architecture parameters, kernel regularity $\|\kappa^{(t)}\|_{C^p}$, and $\|h_f\|_\infty$ (hence on $C_{\mathcal{K}}^X$).

**Step 3: Discretization error in energy pooling.**

The continuum energy is:

$$\Phi_\theta(f,u) = \int_{\Omega_Y} \phi(v^{(T)}(y))\, d\lambda_Y(y). \tag{47}$$

The discretized energy (using exact $v^{(T)}$) would be:

$$\hat{\Phi}_\theta(f,u) = \sum_{j=1}^{M_Y} w_j^Y \phi(v^{(T)}(y_j)). \tag{48}$$

By Assumption B.2, the function

$$y \mapsto \phi(v^{(T)}(y))$$

belongs to $C^p(\overline{\Omega_Y})$ with a uniform $C^p$ bound over $(f,u) \in \mathcal{K}$. Since the quadrature rule on $\Omega_Y$ has order $p$, the quadrature error satisfies

$$\left|\Phi_\theta(f,u) - \hat{\Phi}_\theta(f,u)\right| \le C_\phi M_Y^{-p/n_y}, \tag{49}$$

where $C_\phi$ depends on $\|\phi \circ v^{(T)}\|_{C^p}$ but is bounded uniformly over $(f,u) \in \mathcal{K}$.

**Step 4: Combine errors.**

The actual discretized energy is:

$$\Phi_\theta^{(M_X,M_Y)}(f,u) = \sum_{j=1}^{M_Y} w_j^Y \phi(\tilde{v}^{(T)}(y_j)). \tag{50}$$

We decompose:

$$\left|\Phi_\theta^{(M_X,M_Y)}(f,u) - \Phi_\theta(f,u)\right| \tag{51}$$

$$\le \underbrace{\left|\sum_{j=1}^{M_Y} w_j^Y \phi(\tilde{v}^{(T)}(y_j)) - \sum_{j=1}^{M_Y} w_j^Y \phi(v^{(T)}(y_j))\right|}_{(I)} \tag{52}$$

$$+ \underbrace{\left|\sum_{j=1}^{M_Y} w_j^Y \phi(v^{(T)}(y_j)) - \int_{\Omega_Y} \phi(v^{(T)}(y))\, d\lambda_Y(y)\right|}_{(II)}. \tag{53}$$

**Bound on (I):** By Lipschitz property of $\phi$ and Step 2:

$$(I) \le \sum_{j=1}^{M_Y} w_j^Y \left| \phi(\tilde{v}^{(T)}(y_j)) - \phi(v^{(T)}(y_j)) \right| \tag{54}$$

$$\le L_\phi \sum_{j=1}^{M_Y} w_j^Y \|\tilde{v}^{(T)}(y_j) - v^{(T)}(y_j)\| \tag{55}$$

$$\le L_\phi \lambda_Y(\Omega_Y) \cdot C_T M_X^{-p/n_x}. \tag{56}$$

**Bound on (II):** By Step 3:

$$(II) \le C_\phi M_Y^{-p/n_y}. \tag{57}$$

**Final bound:**

$$\left| \Phi_\theta^{(M_X, M_Y)}(f,u) - \Phi_\theta(f,u) \right| \le L_\phi \lambda_Y(\Omega_Y) C_T M_X^{-p/n_x} + C_\phi M_Y^{-p/n_y}. \tag{58}$$

Since all constants are uniform over the compact set $\mathcal{K}$, taking the supremum:

$$\sup_{(f,u)\in\mathcal{K}} \left| \Phi_\theta^{(M_X, M_Y)}(f,u) - \Phi_\theta(f,u) \right| = \mathcal{O}\left( M_X^{-p/n_x} + M_Y^{-p/n_y} \right) \quad \text{as } M_X, M_Y \to \infty. \tag{59}$$

This completes the proof. □

*Remark* B.12 (Convergence rates). On quasi-uniform grids with $M_X = M_Y = M$ and first-order quadrature ($p = 1$), the bound becomes

$$O\Big( M^{-1/\max(n_x, n_y)} \Big).$$

Higher-order quadrature rules increase $p$ and yield faster convergence. This shows that incorporating quadrature weights into attention and pooling makes EBOs inherently compatible with multi-resolution and irregular discretizations, a practical advantage for scientific datasets.

### B.3. Stability of GRF-Preconditioned Energy Minimization

We analyze the deterministic inference procedure used by EBO after the GRF initialization. For a fixed context function $f \in \mathcal{H}_X$, write

$$\Phi(u) \equiv \Phi_\theta(f, u).$$

Let $\mathcal{H}_Y$ be a Hilbert space equipped with inner product $\langle \cdot, \cdot \rangle$. Let $C : \mathcal{H}_Y \to \mathcal{H}_Y$ denote the covariance operator of the Gaussian random field reference. We assume that $C$ is self-adjoint, positive semidefinite, and bounded; for a trace-class covariance operator this boundedness holds automatically. In the discretized implementation, $C$ is represented by the GRF covariance matrix $C_N$.

The deterministic GRF-preconditioned energy-minimization flow is

$$\frac{dU_t}{dt} = -C \nabla \Phi(U_t), \tag{60}$$

where $\nabla\Phi$ denotes the Riesz representative of the Fréchet derivative of $\Phi$ with respect to the ambient Hilbert-space inner product.

**Proposition B.13** (Energy descent under GRF-preconditioned flow)**.** *Assume $\Phi : \mathcal{H}_Y \to \mathbb{R}$ is Fréchet differentiable and $C : \mathcal{H}_Y \to \mathcal{H}_Y$ is self-adjoint positive semidefinite. Then along the flow (60), the energy is non-increasing:*

$$\frac{d}{dt}\Phi(U_t) = -\left\langle \nabla\Phi(U_t), C\nabla\Phi(U_t) \right\rangle = -\left\| C^{1/2} \nabla\Phi(U_t) \right\|^2 \le 0. \tag{61}$$

*If $C$ is injective on the subspace containing $\nabla\Phi(U_t)$, then every stationary point of the preconditioned flow is also a stationary point of the energy functional.*

*Proof.* By the chain rule for Fréchet differentiable functionals,

$$\frac{d}{dt}\Phi(U_t) = \left\langle \nabla\Phi(U_t), \frac{dU_t}{dt} \right\rangle.$$

Substituting (60) gives

$$\frac{d}{dt}\Phi(U_t) = -\left\langle \nabla\Phi(U_t), C\nabla\Phi(U_t) \right\rangle.$$

Since $C$ is self-adjoint and positive semidefinite, it admits a positive semidefinite square root $C^{1/2}$, and therefore

$$\langle \nabla\Phi(U_t), C\nabla\Phi(U_t)\rangle = \left\| C^{1/2}\nabla\Phi(U_t) \right\|^2 \geq 0.$$

Hence $\frac{d}{dt}\Phi(U_t) \leq 0$. If $U_t$ is stationary for the preconditioned flow, then $C\nabla\Phi(U_t) = 0$. If $C$ is injective on the subspace containing $\nabla\Phi(U_t)$, this implies $\nabla\Phi(U_t) = 0$. □

We next state the corresponding finite-dimensional descent guarantee for the discretized update used in Algorithm 1. After flattening all spatial and channel dimensions, write the update as

$$U_{k+1} = U_k - hC_N\nabla\Phi(U_k), \tag{62}$$

where $\Phi$ now denotes the discretized scalar energy.

**Proposition B.14** (Discrete energy descent)**.** *Assume $\Phi : \mathbb{R}^n \to \mathbb{R}$ has $L$-Lipschitz gradient with respect to the Euclidean norm, and let $C_N \in \mathbb{R}^{n\times n}$ be symmetric positive semidefinite. If $C_N \neq 0$ and*

$$0 < h < \frac{2}{L\|C_N\|_{\mathrm{op}}}, \tag{63}$$

*then the update (62) satisfies*

$$\Phi(U_{k+1}) \leq \Phi(U_k) - h\left(1 - \frac{hL\|C_N\|_{\mathrm{op}}}{2}\right)\left\| C_N^{1/2}\nabla\Phi(U_k) \right\|^2. \tag{64}$$

*In particular, $\Phi(U_{k+1}) \leq \Phi(U_k)$ for any step size satisfying (63). If $C_N = 0$, the update is the identity map and the energy is unchanged.*

*Proof.* By $L$-smoothness of $\Phi$,

$$\Phi(U_{k+1}) \leq \Phi(U_k) + \langle \nabla\Phi(U_k), U_{k+1} - U_k\rangle + \frac{L}{2}\|U_{k+1} - U_k\|^2.$$

Using $U_{k+1} - U_k = -hC_N\nabla\Phi(U_k)$ gives

$$\Phi(U_{k+1}) \leq \Phi(U_k) - h\left\langle \nabla\Phi(U_k), C_N\nabla\Phi(U_k)\right\rangle + \frac{Lh^2}{2}\left\|C_N\nabla\Phi(U_k)\right\|^2.$$

Since $C_N$ is symmetric positive semidefinite,

$$\langle \nabla\Phi(U_k), C_N\nabla\Phi(U_k)\rangle = \left\| C_N^{1/2}\nabla\Phi(U_k) \right\|^2.$$

Moreover, for a symmetric positive semidefinite matrix, $C_N^2 \preceq \|C_N\|_{\mathrm{op}} C_N$, so

$$\left\|C_N\nabla\Phi(U_k)\right\|^2 = \left\langle \nabla\Phi(U_k), C_N^2\nabla\Phi(U_k)\right\rangle \leq \|C_N\|_{\mathrm{op}}\left\langle \nabla\Phi(U_k), C_N\nabla\Phi(U_k)\right\rangle.$$

Combining the above inequalities yields

$$\Phi(U_{k+1}) \leq \Phi(U_k) - h\left(1 - \frac{hL\|C_N\|_{\mathrm{op}}}{2}\right)\left\| C_N^{1/2}\nabla\Phi(U_k) \right\|^2.$$

The coefficient is positive whenever $0 < h < 2/(L\|C_N\|_{\mathrm{op}})$. □

**Corollary B.15** (Vanishing preconditioned gradient along stable EM)**.** *Under the assumptions of Proposition B.14, suppose in addition that $\Phi$ is bounded below and that $h$ satisfies (63). Then*

$$\sum_{k=0}^{\infty} \left\| C_N^{1/2} \nabla \Phi(U_k) \right\|^2 < \infty, \tag{65}$$

*and hence*

$$\left\| C_N^{1/2} \nabla \Phi(U_k) \right\| \longrightarrow 0. \tag{66}$$

*If $C_N$ is positive definite with smallest eigenvalue $\lambda_{\min}(C_N) > 0$, then $\|\nabla \Phi(U_k)\| \to 0$ as well.*

*Proof.* Let

$$\alpha = h \left( 1 - \frac{hL \|C_N\|_{\mathrm{op}}}{2} \right) > 0.$$

By Proposition B.14,

$$\Phi(U_{k+1}) \le \Phi(U_k) - \alpha \left\| C_N^{1/2} \nabla \Phi(U_k) \right\|^2.$$

Summing from $k = 0$ to $K - 1$ gives

$$\alpha \sum_{k=0}^{K-1} \left\| C_N^{1/2} \nabla \Phi(U_k) \right\|^2 \le \Phi(U_0) - \Phi(U_K).$$

Since $\Phi$ is bounded below, the right-hand side is bounded uniformly in $K$. Taking $K \to \infty$ proves summability. A nonnegative summable sequence must converge to zero, yielding

$$\left\| C_N^{1/2} \nabla \Phi(U_k) \right\| \to 0.$$

If $C_N$ is positive definite, then

$$\left\| C_N^{1/2} \nabla \Phi(U_k) \right\|^2 \ge \lambda_{\min}(C_N) \, \|\nabla \Phi(U_k)\|^2,$$

so $\|\nabla \Phi(U_k)\| \to 0$. □

*Remark* B.16 (Interpretation of the GRF preconditioner). The covariance preconditioner does not merely rescale the gradient numerically. It defines the geometry of the refinement dynamics. The direction $-C\nabla\Phi$ is the covariance-preconditioned descent direction induced by the Gaussian reference. Formally, on a finite-dimensional discretization, or on a subspace where $C^{-1}$ is well-defined, this direction is the steepest descent direction under the covariance-weighted inner product

$$\langle v, w \rangle_{C^{-1}} = \langle C^{-1} v, w \rangle.$$

Equivalently, high-frequency directions corresponding to small eigenvalues of $C$ are damped by the update. This prevents the refinement dynamics from reducing to grid-wise Euclidean descent and keeps the inference procedure compatible with the function-space geometry induced by the GRF reference.

*Remark* B.17 (Relation to stochastic Langevin sampling). The deterministic update analyzed above should be interpreted as GRF-preconditioned energy minimization, not exact sampling from the Gaussian-reference Gibbs measure

$$\nu_\theta(du \mid f) \propto \exp\{-\Phi_\theta(f, u)\} \, \mu_0(du).$$

To obtain a Langevin-style sampler for this measure, one must include both the Gaussian-reference drift and matched GRF noise, yielding the continuous-time preconditioned Langevin dynamics

$$dU_t = (-C \nabla_u \Phi_\theta(f, U_t) - U_t) \, dt + \sqrt{2} \, dW_t^C,$$

where $W_t^C$ is a $C$-Wiener process. The corresponding Euler–Maruyama update is

$$U_{k+1} = U_k + h \, (S_\theta(f; U_k) - U_k) + \sqrt{2h} \, \xi_k, \qquad \xi_k \sim \mathcal{N}(0, C_N).$$

Thus, adding noise alone to the deterministic EM update is not sufficient to recover Langevin sampling for the Gaussian-reference Gibbs measure; the reference drift $-U_k$ is also required. In the main experiments, we use the deterministic update in Algorithm 1 for stable and scalable energy refinement from a GRF initialization.

*Remark* B.18 (Scope of the descent guarantee). Propositions B.13 and B.14 prove monotonic descent of the learned scalar energy under the deterministic GRF-preconditioned update. They do not assert global convergence to a unique minimizer, which would require additional assumptions such as convexity or a suitable gradient-dominance condition. For nonconvex neural energies, the guarantee is stability of the refinement dynamics and convergence of the preconditioned gradient norm under the conditions of Corollary B.15.

### B.4. Universal Approximation

We now establish that the class of EBOs is expressive enough to approximate any continuous energy functional.

**Definition B.19** (Continuous Energy Functional)**.** A map $\Phi^\dagger : \mathcal{H}_X \times \mathcal{H}_Y \to \mathbb{R}$ is a **continuous energy functional** if it is continuous with respect to the product topology on $\mathcal{H}_X \times \mathcal{H}_Y$.

**Assumption B.20** (Neural Network Approximation)**.** We assume that:

1. The lifting maps $P_X, P_Y$ can be chosen from a class of neural networks that is dense in continuous functions on compact sets.
2. The kernels $\kappa^{(t)}$ can be parameterized by neural networks that are dense in $C(\overline{\Omega_Y} \times \overline{\Omega_X}; \mathbb{R}^{d_h \times d_h})$ on compact sets.
3. The energy head $\phi$ can be chosen from a class dense in $C(\mathbb{R}^{d_h}; \mathbb{R})$ on compact sets.

This assumption is satisfied by standard feed-forward networks with non-polynomial activations (e.g., ReLU, sigmoid, tanh) by the classical universal approximation theorem (Hornik et al., 1989; Cybenko, 1989).

**Lemma B.21** (Finite-dimensional reduction via point evaluations)**.** *Let $F : C(\overline{\Omega_Y}; \mathbb{R}^{d_h}) \to \mathbb{R}$ be continuous and let $K_v \subset C(\overline{\Omega_Y}; \mathbb{R}^{d_h})$ be compact (in the sup norm). Then for every $\epsilon > 0$ there exist points $y_1, \dots, y_N \in \overline{\Omega_Y}$ and a continuous map $\Psi : \mathbb{R}^{Nd_h} \to \mathbb{R}$ such that*

$$\sup_{v \in K_v} \left| F(v) - \Psi\big(v(y_1), \dots, v(y_N)\big) \right| < \epsilon.$$

*Proof.* Since $F$ is continuous on the compact set $K_v$, it is uniformly continuous on $K_v$. Hence there exists $\delta > 0$ such that for all $v, w \in K_v$,

$$\|v - w\|_\infty < \delta \quad \Longrightarrow \quad |F(v) - F(w)| < \epsilon.$$

Because $K_v$ is compact in $C(\overline{\Omega_Y}; \mathbb{R}^{d_h})$, by Arzelà–Ascoli it is equicontinuous. Therefore, there exists $r > 0$ such that for every $v \in K_v$,

$$\|y - y'\| < r \quad \Longrightarrow \quad \|v(y) - v(y')\| < \delta \qquad \text{for all } y, y' \in \overline{\Omega_Y}.$$

Cover $\overline{\Omega_Y}$ by finitely many balls $B_r(y_1), \dots, B_r(y_N)$. Choose a continuous partition of unity $\{\eta_1, \dots, \eta_N\}$ subordinate to this cover, i.e. $\eta_j \geq 0$, $\operatorname{supp}(\eta_j) \subset B_r(y_j)$, and $\sum_{j=1}^N \eta_j(y) = 1$ for all $y$.

Define the reconstruction operator $R : \mathbb{R}^{Nd_h} \to C(\overline{\Omega_Y}; \mathbb{R}^{d_h})$ by

$$(Rz)(y) := \sum_{j=1}^N \eta_j(y)\, z_j, \qquad z = (z_1, \dots, z_N) \in (\mathbb{R}^{d_h})^N.$$

Now define $\Psi : \mathbb{R}^{Nd_h} \to \mathbb{R}$ by

$$\Psi(z) := F(Rz).$$

This $\Psi$ is continuous because $R$ is continuous (linear and bounded) and $F$ is continuous.

For each $v \in K_v$, set $z(v) := (v(y_1), \dots, v(y_N))$ and consider $Rv := R(z(v))$. For any $y \in \overline{\Omega_Y}$, whenever $\eta_j(y) > 0$ we have $y \in B_r(y_j)$ and hence $\|v(y) - v(y_j)\| < \delta$. Using $\sum_j \eta_j(y) = 1$ and $\eta_j(y) \geq 0$,

$$\|v(y) - Rv(y)\| = \left\| \sum_{j=1}^N \eta_j(y) \big(v(y) - v(y_j)\big) \right\| \leq \sum_{j=1}^N \eta_j(y)\, \|v(y) - v(y_j)\| < \delta.$$

Taking the supremum over $y$ gives $\|v - Rv\|_\infty < \delta$, and by the choice of $\delta$,

$$|F(v) - F(Rv)| < \epsilon.$$

But $F(Rv) = \Psi(v(y_1), \dots, v(y_N))$ by definition of $\Psi$. Therefore

$$\sup_{v \in K_v} \left|F(v) - \Psi(v(y_1), \dots, v(y_N))\right| < \epsilon.$$

□

**Lemma B.22** (Integral pooling with outer readout)**.** *Under the assumptions of Lemma B.21, for every $\epsilon > 0$ there exist $N \in \mathbb{N}$, points $y_1, \dots, y_N$, continuous $\phi : \overline{\Omega_Y} \times \mathbb{R}^{d_h} \to \mathbb{R}^{Nd_h}$ and continuous $\rho : \mathbb{R}^{Nd_h} \to \mathbb{R}$ such that*

$$\sup_{v \in K_v} \left| F(v) - \rho\left( \int_{\Omega_Y} \phi(y, v(y))\, d\lambda_Y(y) \right) \right| < \epsilon.$$

*Proof.* Fix $\epsilon > 0$ and let $F : C(\overline{\Omega_Y}; \mathbb{R}^{d_h}) \to \mathbb{R}$ be continuous. Let $K_v \subset C(\overline{\Omega_Y}; \mathbb{R}^{d_h})$ be compact.

**Step 1: Finite-dimensional reduction.** By Lemma B.21, there exist points $y_1, \dots, y_N \in \overline{\Omega_Y}$ and a continuous map $\Psi : \mathbb{R}^{Nd_h} \to \mathbb{R}$ such that

$$\sup_{v \in K_v} \left| F(v) - \Psi\big(v(y_1), \dots, v(y_N)\big) \right| < \epsilon/2. \tag{67}$$

Define the compact set

$$E := \{(v(y_1), \dots, v(y_N)) : \ v \in K_v\} \subset \mathbb{R}^{Nd_h}.$$

**Step 2: Uniform continuity of $\Psi$ on a compact set containing both features.** We will also consider localized-average features (defined below) and the corresponding compact set

$$E_a := \{(a_1(v), \dots, a_N(v)) : \ v \in K_v\} \subset \mathbb{R}^{Nd_h}.$$

Since each map $v \mapsto a_j(v)$ is continuous (as a bounded linear functional on $C(\overline{\Omega_Y}; \mathbb{R}^{d_h})$), and $K_v$ is compact, we have that $E_a$ is compact. Hence $E \cup E_a$ is compact, and therefore $\Psi$ is uniformly continuous on $E \cup E_a$. Thus there exists $\alpha > 0$ such that for all $z, z' \in E \cup E_a$,

$$\|z - z'\| < \alpha \quad \Longrightarrow \quad |\Psi(z) - \Psi(z')| < \epsilon/2. \tag{68}$$

**Step 3: Replace point evaluations by localized averages.** Since $K_v$ is compact in $C(\overline{\Omega_Y}; \mathbb{R}^{d_h})$, it is equicontinuous. Hence there exists $r > 0$ such that for all $v \in K_v$ and all $y \in B_r(y_j)$,

$$\|v(y) - v(y_j)\| < \frac{\alpha}{\sqrt{N}}. \tag{69}$$

For each $j = 1, \dots, N$, choose a nonnegative continuous bump function $\eta_j : \overline{\Omega_Y} \to [0, \infty)$ with

$$\operatorname{supp}(\eta_j) \subset B_r(y_j), \qquad \int_{\Omega_Y} \eta_j(y)\, d\lambda_Y(y) = 1.$$

Define the averaged features

$$a_j(v) := \int_{\Omega_Y} v(y)\, \eta_j(y)\, d\lambda_Y(y) \in \mathbb{R}^{d_h}.$$

Then for any $v \in K_v$,

$$\|a_j(v) - v(y_j)\| = \left\| \int_{\Omega_Y} (v(y) - v(y_j)) \eta_j(y)\, dy \right\| \le \int_{\Omega_Y} \|v(y) - v(y_j)\| \eta_j(y)\, dy < \frac{\alpha}{\sqrt{N}},$$

using (69) and $\int \eta_j = 1$. Therefore, letting

$$z(v) := (v(y_1), \dots, v(y_N)), \qquad z_a(v) := (a_1(v), \dots, a_N(v)),$$

we obtain

$$\|z(v) - z_a(v)\|^2 = \sum_{j=1}^{N} \|v(y_j) - a_j(v)\|^2 < \sum_{j=1}^{N} \left(\frac{\alpha}{\sqrt{N}}\right)^2 = \alpha^2,$$

hence $\|z(v) - z_a(v)\| < \alpha$. Since $z(v) \in E$ and $z_a(v) \in E_a$, we may apply (68) to conclude

$$\sup_{v \in K_v} \left|\Psi\big(z(v)\big) - \Psi\big(z_a(v)\big)\right| < \epsilon/2. \tag{70}$$

**Step 4: Encode the averaged features as a single pooled integral.** Define $\phi : \overline{\Omega_Y} \times \mathbb{R}^{d_h} \to \mathbb{R}^{Nd_h}$ by

$$\phi(y, z) := (\eta_1(y)z,\ \eta_2(y)z,\ \ldots,\ \eta_N(y)z) \in (\mathbb{R}^{d_h})^N \cong \mathbb{R}^{Nd_h}.$$

Then for any $v \in K_v$,

$$\int_{\Omega_Y} \phi(y, v(y))\, d\lambda_Y(y) = \left(\int_{\Omega_Y} \eta_1(y)v(y)\, dy,\ \ldots,\ \int_{\Omega_Y} \eta_N(y)v(y)\, dy\right) = z_a(v).$$

Finally, set $\rho := \Psi$.

**Step 5: Combine the errors.** For any $v \in K_v$,

$$\begin{aligned}\left|F(v) - \rho\left(\int_{\Omega_Y} \phi(y, v(y))\, dy\right)\right| &= \left|F(v) - \Psi\big(z_a(v)\big)\right| \\ &\le \left|F(v) - \Psi\big(z(v)\big)\right| + \left|\Psi\big(z(v)\big) - \Psi\big(z_a(v)\big)\right|.\end{aligned}$$

Taking suprema over $v \in K_v$ and using (67) and (70), we obtain

$$\sup_{v \in K_v} \left|F(v) - \rho\left(\int_{\Omega_Y} \phi(y, v(y))\, d\lambda_Y(y)\right)\right| < \epsilon/2 + \epsilon/2 = \epsilon,$$

which proves the claim. □

**Lemma B.23** (Density of Neural Integral Operators)**.** *Let $K \subset C(\overline{\Omega_X}; \mathbb{R}^d) \times C(\overline{\Omega_Y}; \mathbb{R}^m)$ be compact, and let*

$$\mathcal{G}^\dagger : C(\overline{\Omega_X}; \mathbb{R}^d) \times C(\overline{\Omega_Y}; \mathbb{R}^m) \to C(\overline{\Omega_Y}; \mathbb{R}^{d_h})$$

*be a continuous operator such that $\mathcal{G}^\dagger(K)$ is equicontinuous. Then, for any $\delta > 0$, there exists a depth-$T$ neural integral operator $\mathcal{G}_\theta$ of the form*

$$v^{(0)}(y) = h_u(y), \qquad v^{(t)}(y) = \sigma\left(W^{(t)}v^{(t-1)}(y) + \int_{\Omega_X} \kappa^{(t)}(y, x)\, h_f(x)\, d\lambda_X(x) + b^{(t)}(y)\right), \tag{71}$$

*with continuous kernels $\kappa^{(t)}$, weight matrices $W^{(t)}$, biases $b^{(t)}$, and Lipschitz activation $\sigma$, such that*

$$\sup_{(f,u) \in K} \|\mathcal{G}_\theta(f, u) - \mathcal{G}^\dagger(f, u)\|_\infty < \delta. \tag{72}$$

*Proof sketch.* The equicontinuity assumption ensures that $\mathcal{G}^\dagger(K)$ is relatively compact in $C(\overline{\Omega_Y}; \mathbb{R}^{d_h})$ by the Arzelà–Ascoli theorem.

The claim follows from universal approximation theorems for nonlinear operators on function spaces. In particular, Theorems 4–5 of (Chen & Chen, 1995) show that, for a compact subset of a Banach space of continuous functions and a continuous (possibly nonlinear) operator with equicontinuous range, there exist neural-network operators (built from linear functionals and a fixed non-polynomial activation) that approximate the operator uniformly on that compact set. More recently, Theorem 11 of (Kovachki et al., 2023) establishes an analogous result in the specific setting of neural operators: given Banach spaces of functions $A$ and $U$, a continuous operator $G : A \to U$, and a compact set $K \subset A$, there exists,

for any $\delta > 0$, a depth-$T$ neural operator (a composition of affine integral kernel layers and pointwise nonlinearities) that satisfies

$$\sup_{a \in K} \|G(a) - \mathcal{G}(a)\|_U < \delta.$$

We instantiate these results with

$$A = C(\overline{\Omega_X}; \mathbb{R}^d) \times C(\overline{\Omega_Y}; \mathbb{R}^m), \qquad U = C(\overline{\Omega_Y}; \mathbb{R}^{d_h}), \qquad G = \mathcal{G}^\dagger,$$

and the compact domain $K \subset A$. The neural integral operator architecture used in this paper (iteration of affine maps of the form

$$v^{(t)}(y) \mapsto W^{(t)} v^{(t-1)}(y) + \textstyle\int_{\Omega_X} \kappa^{(t)}(y, x)\, h_f(x)\, d\lambda_X(x) + b^{(t)}(y),$$

followed by pointwise $\sigma$) matches the neural operator class considered by (Kovachki et al., 2023) up to notation. Therefore, by the cited theorems of (Chen & Chen, 1995) and (Kovachki et al., 2023), for any $\delta > 0$ there exists a choice of depth $T$ and parameters $\theta$ such that the resulting $\mathcal{G}_\theta$ satisfies

$$\sup_{(f,u) \in K} \|\mathcal{G}_\theta(f, u) - \mathcal{G}^\dagger(f, u)\|_\infty < \delta,$$

which is the desired conclusion. □

**Theorem B.24** (Universal Approximation for EBO). *Let $\Phi^\dagger : \mathcal{H}_X \times \mathcal{H}_Y \to \mathbb{R}$ be a continuous energy functional. For any compact set $\mathcal{K} \subset \mathcal{H}_X \times \mathcal{H}_Y$ and any $\epsilon > 0$, there exists an EBO $\Phi_\theta$ (as in Definition B.5) with sufficiently many layers $T$ and hidden dimension $d_h$ such that:*

$$\sup_{(f,u) \in \mathcal{K}} \left|\Phi_\theta(f, u) - \Phi^\dagger(f, u)\right| < \epsilon. \tag{73}$$

*Proof.* The proof constructs an approximating EBO in three stages.

**Stage 1: Reduction to continuous operators on function spaces.**

By Assumption B.1, the embeddings $\mathcal{H}_X \hookrightarrow C(\overline{\Omega_X}; \mathbb{R}^d)$ and $\mathcal{H}_Y \hookrightarrow C(\overline{\Omega_Y}; \mathbb{R}^m)$ are continuous. Hence $\Phi^\dagger : \mathcal{H}_X \times \mathcal{H}_Y \to \mathbb{R}$ extends to a continuous map in the $C(\overline{\Omega_X}) \times C(\overline{\Omega_Y})$ topology. Denote this extension by $\Phi^\dagger$.

Denote the image of $\mathcal{K}$ under the embedding as:

$$\tilde{\mathcal{K}} = \{(f|_{\overline{\Omega_X}}, u|_{\overline{\Omega_Y}}) : (f, u) \in \mathcal{K}\} \subset C(\overline{\Omega_X}; \mathbb{R}^d) \times C(\overline{\Omega_Y}; \mathbb{R}^m). \tag{74}$$

This is compact by continuity of the embedding.

**Stage 2: Approximation by neural operator architecture.**

**Step 2a: Lifting.** We choose continuous lifting maps $P_X^\dagger, P_Y^\dagger$ (e.g., identity-plus-coordinate). By Assumption B.20, there exist neural networks $P_X, P_Y$ that approximate $P_X^\dagger, P_Y^\dagger$ uniformly on the compact ranges induced by $\tilde{\mathcal{K}}$.

**Step 2b: Finite-dimensional reduction of the energy.**

Since $\Phi^\dagger$ is continuous on the compact set $\tilde{\mathcal{K}} \subset C(\overline{\Omega_X}; \mathbb{R}^d) \times C(\overline{\Omega_Y}; \mathbb{R}^m)$, by Lemma B.21, for any $\epsilon > 0$ there exist points $\{x_1, \ldots, x_{N_X}\} \subset \overline{\Omega_X}$ and $\{y_1, \ldots, y_{N_Y}\} \subset \overline{\Omega_Y}$ with a continuous map $\Psi : \mathbb{R}^{N_X d + N_Y m} \to \mathbb{R}$ such that

$$\sup_{(f,u) \in \tilde{\mathcal{K}}} \left|\Phi^\dagger(f, u) - \Psi(f(x_1), \ldots, f(x_{N_X}), u(y_1), \ldots, u(y_{N_Y}))\right| < \frac{\epsilon}{4}. \tag{75}$$

Since $\Phi$ is continuous on the compact $\tilde{\mathcal{K}} \subset C(\overline{\Omega_X}; \mathbb{R}^d) \times C(\overline{\Omega_Y}; \mathbb{R}^m)$, by uniform continuity it can be approximated by point-evaluations of $f$ and $u$ on finite subsets.

By equicontinuity of $\tilde{\mathcal{K}}$ (compactness in the sup norm) and Lemma B.22, we can represent this functional via localized averages:

$$a_i(f) := \int_{\Omega_X} f(x) \eta_i(x)\, d\lambda_X(x) \approx f(x_i), \quad b_j(u) := \int_{\Omega_Y} u(y) \xi_j(y)\, d\lambda_Y(y) \approx u(y_j),$$

where $\eta_i, \xi_j$ are smooth bump functions with $\int \eta_i = 1$, $\int \xi_j = 1$.

Define the global feature vector

$$s(f,u) := (a_1(f), \dots, a_{N_X}(f), b_1(u), \dots, b_{N_Y}(u)) \in \mathbb{R}^{d_h}, \quad d_h := N_X d + N_Y m.$$

The energy depends only on $s(f,u)$, so we encode this in a constant latent field:

$$v^\dagger(f,u)(y) := s(f,u) \quad \forall y \in \overline{\Omega_Y}.$$

Define the energy density

$$\phi^\dagger(z) := \frac{1}{\lambda_Y(\Omega_Y)} \Psi(z), \quad \Phi^\dagger(y,z) := \phi^\dagger(z).$$

By Lemma B.22, the integral representation satisfies:

$$\int_{\Omega_Y} \Phi^\dagger(y, v^\dagger(f,u)(y))\, d\lambda_Y(y) = \Psi(a_1(f), \dots, b_{N_Y}(u)), \tag{76}$$

and therefore

$$\sup_{(f,u)\in\tilde{\mathcal{K}}} \left| \Phi^\dagger(f,u) - \int_{\Omega_Y} \Phi^\dagger(y, v^\dagger(f,u)(y))\, d\lambda_Y(y) \right| < \frac{\epsilon}{2}. \tag{77}$$

**Remark on the constant field:** The latent field $v^\dagger$ is constant in $y$ because $\Phi^\dagger$ depends on $(f,u)$ only through the aggregated features $s(f,u)$, not through pointwise spatial variation. This degenerate case is sufficient for universal approximation; if $\Phi^\dagger$ exhibits point-wise dependence, one can construct a spatially-varying $v^\dagger$ analogously using the techniques in Lemma B.22.

**Step 2c: Neural-operator universal approximation.**

By Lemma B.23, applied to the compact set $\tilde{\mathcal{K}}$ and target operator $\mathcal{G}^\dagger$, for any $\delta > 0$ there exists a depth-$T$ neural integral operator $v^{(T)}$ of the stated form such that

$$\sup_{(f,u)\in\tilde{\mathcal{K}}} \|v^{(T)}(f,u) - \mathcal{G}^\dagger(f,u)\|_\infty < \delta. \tag{78}$$

**Step 2d: Kernel parameterization by neural networks.**

By Assumption B.20, each continuous kernel $\kappa^{(t),\dagger}$ achieving the above can be approximated by neural network-parameterized kernels $\kappa_\theta^{(t)}$ such that:

$$\|\kappa_\theta^{(t)} - \kappa^{(t),\dagger}\|_\infty < \frac{\delta}{T \cdot C_{\text{stab}}}, \tag{79}$$

where $C_{\text{stab}}$ is a stability constant from Lemma B.9.

By the stability estimate (Lemma B.9), the error propagates as:

$$\sup_{(f,u)\in\tilde{K}} \|v_\theta^{(T)}(f,u) - v^{(T),\dagger}(f,u)\|_\infty < \delta. \tag{80}$$

**Stage 3: Energy head approximation.**

By the previous step, there exists $\Phi^\dagger$ and $v^\dagger$ such that

$$\sup_{(f,u)\in\tilde{\mathcal{K}}} \left| \Phi^\dagger(f,u) - \int_{\Omega_Y} \Phi^\dagger(y, v^\dagger(y))\, d\lambda_Y(y) \right| < \frac{\epsilon}{4}. \tag{81}$$

In our EBO architecture, the representation $v^{(T)}(y)$ already includes positional information via $(u(y), y)$ and positional encodings, so for $(f,u) \in \tilde{\mathcal{K}}$ we may view $\Phi^\dagger(y,\cdot)$ as a continuous function of the latent variable $z = v^{(T)}(y)$ on the compact range of $v^{(T)}$. Thus there exists a continuous $\phi^\dagger : \mathbb{R}^{d_h} \to \mathbb{R}$ such that

$$\Phi^\dagger(y, v^{(T)}(y)) = \phi^\dagger(v^{(T)}(y))$$

for all relevant $(f, u)$ and $y$.

By Assumption B.20, $\phi^\dagger$ can be approximated by a neural network $\phi_\theta$ such that:

$$\sup_{z\in[-M,M]^{d_h}} |\phi_\theta(z) - \phi^\dagger(z)| < \frac{\epsilon}{4\lambda_Y(\Omega_Y)}, \tag{82}$$

where $M = \sup_{(f,u)\in\tilde{K}} \|v^\dagger\|_\infty + \delta$.

**Stage 4: Combine all approximations.**

The EBO energy is:

$$\Phi_\theta(f, u) = \int_{\Omega_Y} \phi_\theta(v_\theta^{(T)}(y))\, d\lambda_Y(y). \tag{83}$$

For any $(f, u) \in \mathcal{K}$:

$$|\Phi_\theta(f, u) - \Phi^\dagger(f, u)| \tag{84}$$

$$\leq \underbrace{\left| \int_{\Omega_Y} \phi_\theta(v_\theta^{(T)}(y))\, d\lambda_Y - \int_{\Omega_Y} \phi_\theta(v^\dagger(y))\, d\lambda_Y \right|}_{(A)} \tag{85}$$

$$+ \underbrace{\left| \int_{\Omega_Y} \phi_\theta(v^\dagger(y))\, d\lambda_Y - \int_{\Omega_Y} \phi^\dagger(v^\dagger(y))\, d\lambda_Y \right|}_{(B)} \tag{86}$$

$$+ \underbrace{\left| \int_{\Omega_Y} \phi^\dagger(v^\dagger(y))\, d\lambda_Y - \Phi^\dagger(f, u) \right|}_{(C)}. \tag{87}$$

**Bound on (A):** By the Lipschitz property of $\phi_\theta$ (neural networks with bounded weights are Lipschitz), we have

$$(A) \leq L_{\phi_\theta} \int_{\Omega_Y} \|v_\theta^{(T)}(y) - v^\dagger(y)\|\, d\lambda_Y(y) \tag{88}$$

$$\leq L_{\phi_\theta} \lambda_Y(\Omega_Y) \cdot \delta. \tag{89}$$

Choose

$$\delta = \frac{\epsilon}{4L_{\phi_\theta}\lambda_Y(\Omega_Y)} \tag{90}$$

to obtain $(A) < \frac{\epsilon}{4}$.

**Bound on (B):** By the approximation of $\phi^\dagger$ by $\phi_\theta$:

$$(B) \leq \int_{\Omega_Y} |\phi_\theta(v^\dagger(y)) - \phi^\dagger(v^\dagger(y))|\, d\lambda_Y(y) \tag{91}$$

$$\leq \lambda_Y(\Omega_Y) \cdot \frac{\epsilon}{4\lambda_Y(\Omega_Y)} = \frac{\epsilon}{4}. \tag{92}$$

**Bound on (C):** By construction of $\phi^\dagger$:

$$(C) < \frac{\epsilon}{4}. \tag{93}$$

**Conclusion:**

$$|\Phi_\theta(f, u) - \Phi^\dagger(f, u)| < \frac{\epsilon}{4} + \frac{\epsilon}{4} + \frac{\epsilon}{4} < \epsilon. \tag{94}$$

Taking the supremum over $(f, u) \in \mathcal{K}$:

$$\sup_{(f,u)\in\mathcal{K}} |\Phi_\theta(f, u) - \Phi^\dagger(f, u)| < \epsilon. \tag{95}$$

This completes the proof. □

**Corollary B.25** (Density in Probability Measures)**.** *Under the conditions of Theorem 4.2, if $\Phi^\dagger$ induces a well-defined probability measure:*

$$\nu^\dagger(\cdot \mid f) \propto \exp(-\Phi^\dagger(f, \cdot)) \cdot \mu_0, \tag{96}$$

*where $\mu_0$ is a Gaussian reference measure on $\mathcal{H}_Y$, then the measures induced by approximating EBOs converge weakly:*

$$\nu_\theta(\cdot \mid f) \xrightarrow{w} \nu^\dagger(\cdot \mid f) \quad \text{as } \theta \to \theta^*. \tag{97}$$

*Proof.* **Step 1:** By Theorem 4.2, for any $\epsilon > 0$ and compact $K_u \subset \mathcal{H}_Y$:

$$\sup_{u \in K_u} |\Phi_\theta(f, u) - \Phi^\dagger(f, u)| < \epsilon. \tag{98}$$

**Step 2:** For the Radon-Nikodym derivatives:

$$\frac{d\nu_\theta}{d\mu_0}(u) = \frac{\exp(-\Phi_\theta(f, u))}{Z_\theta(f)}, \quad \frac{d\nu^\dagger}{d\mu_0}(u) = \frac{\exp(-\Phi^\dagger(f, u))}{Z^\dagger(f)}. \tag{99}$$

**Step 3:** On $K_u$:

$$\left| \frac{\exp(-\Phi_\theta(f, u))}{\exp(-\Phi^\dagger(f, u))} - 1 \right| \leq e^\epsilon - 1 \approx \epsilon \quad \text{for small } \epsilon. \tag{100}$$

**Step 4:** We assume that for all $f$ under consideration the normalizing constants $Z^\dagger(f)$ and $Z_\theta(f)$ are finite and that $\exp(-\Phi_\theta(f, u))$ is dominated by an integrable envelope uniformly in $\theta$. Under this integrability assumption, the normalizing constants satisfy

$$Z_\theta(f) = \int_{\mathcal{H}_Y} \exp(-\Phi_\theta(f, u))\, d\mu_0(u) \tag{101}$$

$$\to \int_{\mathcal{H}_Y} \exp(-\Phi^\dagger(f, u))\, d\mu_0(u) = Z^\dagger(f) \tag{102}$$

by dominated convergence.

**Step 5:** For any bounded continuous test function $\psi : \mathcal{H}_Y \to \mathbb{R}$:

$$\left| \int_{\mathcal{H}_Y} \psi(u)\, d\nu_\theta(u \mid f) - \int_{\mathcal{H}_Y} \psi(u)\, d\nu^\dagger(u \mid f) \right| \tag{103}$$

$$= \left| \int_{\mathcal{H}_Y} \psi(u) \left( \frac{d\nu_\theta}{d\mu_0}(u) - \frac{d\nu^\dagger}{d\mu_0}(u) \right) d\mu_0(u) \right| \tag{104}$$

$$\leq \|\psi\|_\infty \int_{\mathcal{H}_Y} \left| \frac{d\nu_\theta}{d\mu_0}(u) - \frac{d\nu^\dagger}{d\mu_0}(u) \right| d\mu_0(u) \tag{105}$$

$$\to 0 \quad \text{as } \epsilon \to 0. \tag{106}$$

This establishes weak convergence. □

# C. Synthetic function experiments

In this section, we provide a detailed summary of 1D function classes defined for EBO as well as training details.

## C.1. Function Class

We evaluate EBO on three function classes as defined in the following paragraphs.

**Cosine Functions.** We begin with a synthetic benchmark to validate our approach under controlled conditions and perform ablation studies. We generate functions of the form $f(x) = A\cos(2\pi\omega x + \phi)$, where amplitude $A \sim \mathcal{U}(0.5, 1.5)$, frequency $\omega \sim \mathcal{U}(4, 6)$, and phase $\phi \sim \mathcal{U}(0, 2\pi)$. Observations are sampled at irregular locations $\{x_i\}_{i=1}^N$ drawn uniformly from $[0, 1]$ and sorted. We initialize the initial function with Gaussian random field samples from a GP prior with RBF kernel.

**Damping.** We consider $y(x) = e^{-\alpha x}\sin(2\pi f x)$ with decay rate $\alpha \sim \mathcal{U}(3,6)$ and frequency $f \sim \mathcal{U}(8,16)$. The exponential envelope breaks translation invariance, coupling local oscillatory structure to global amplitude decay. Such non-stationary waveforms arise naturally in RLC circuits, seismology, and underdamped mechanical systems, where energy dissipation manifests as exponential attenuation of periodic motion.

**Izhikevich neuron.** We sample membrane potential trajectories from the Izhikevich spiking neuron model (Izhikevich, 2003), a two-dimensional dynamical system that balances biological plausibility with computational efficiency:

$$\frac{dv}{dt} = 0.04v^2 + 5v + 140 - u + I, \quad \frac{du}{dt} = a(bv - u), \tag{107}$$

with reset $v \leftarrow c$, $u \leftarrow u + d$ when $v \geq 30$ mV. Here $v$ is the membrane potential, $u$ is a recovery variable modeling slow ionic currents, and $I$ is the injected current. The four parameters $(a, b, c, d)$ determine the firing pattern: we sample regular spiking (RS), intrinsically bursting (IB), and fast spiking (FS) neuron types with $I \sim \mathcal{U}(5,15)$. Unlike the smooth function classes above, the resulting spike trains exhibit discontinuous resets, testing the model's capacity to learn distributions over non-smooth trajectories prevalent in computational neuroscience and brain-computer interfaces.

### C.2. Training Modes

We compare three training paradigms: direct prediction (Operator), energy-based minimization (EBO) and denoising (DDO). Table 4 summarizes the training configuration for different backbones.

*Table 4.* **Backbone-specific hyperparameters and parameter counts.** All architectures use `hidden_dim=128`, 4 layers, 4 heads, and the same coord encoding / token-concat conditioning.

| Parameter | Transformer | FactFormer | Galerkin |
|---|---|---|---|
| Hidden dim | 128 | 128 | 128 |
| # Layers | 4 | 4 | 4 |
| # Heads | 4 | 4 | 4 |
| Attention type | Vanilla MHA | Factorized | Galerkin (linear) |
| Special params | — | geotype=structured_1D | — |
| Backbone params | 476,544 | 370,049 | 615,169 |
| Pred. head params | 16,641 | 16,641 | 16,641 |
| Embedding/coord | 66,561 | 66,561 | 66,561 |
| **Total params** | **559,746** | **453,251** | **698,371** |

### C.3. Training Configuration

Table 5 summarizes the training configuration for EBO on the various functions datasets, and Table 6 summarizes the configuration of the shared optimizer and scheduler.

*Table 5.* Training mode hyperparameters (Operator/ EBO / DDO) shared across all 5 architectures.

| Parameter | Operator | EBO | DDO |
|---|---|---|---|
| Prediction method | One-shot MLP head | iterative EM | Reverse diffusion |
| Prediction head | 2-layer MLP | — | — |
| Pred. head LR multiplier | $10\times$ | — | — |
| Input initialization | Zero-valued | GRF samples from GP prior | GRF noise |
| Number of steps (training) | — | $T \sim \mathcal{U}[8, 64]$ | — |
| Step size $\alpha$ | — | 0.01 (fixed, non-learnable) | — |
| Time conditioning | none | none | timestep embedding |

**Multi-Resolution Training.** Models are trained on variable resolution grids $N \in \{32, 64, 128, 256\}$ with irregular sampling

*Table 6.* Shared optimizer and training schedule (all architectures, all training modes).

| Parameter | Value |
|---|---|
| Optimizer | AdamW |
| Learning rate | $5 \times 10^{-4}$ |
| Warmup steps | 5,000 |
| LR schedule | Cosine decay (min_lr_scale = 10) |
| Batch size | 64 |
| Resolution list | $\{16, 32, 64, 128, 256\}$ (multi-resolution) |
| Train iterations | 50,000 |
| Mixed precision | bfloat16 (AMP) |
| Coordinate encoding | SIREN ($\omega_0 = 30$, hidden_dim = 256, layers = 1) |
| Conditioning mode | Token concat |
| Seed | 42 |

**Algorithm 1** Iterative energy-minimization training step

**Require:** Parameters $\theta$, step size $h$, steps $K$, covariance operator $C_N$, energy-based operator $\Phi_\theta$.
1: Sample $(f, u^*)$.
2: Initialize $U_\theta^{(0)} \sim \mathcal{N}(0, C_N)$.
3: **for** $k = 0$ **to** $K - 1$ **do**
4: Compute scalar energy $E_k = \Phi_\theta(f, U_\theta^{(k)})$.
5: Compute $S_\theta^{(k)} = -C_N \nabla_{U_\theta^{(k)}} E_k$ by auto-differentiation with detachment.
6: Update $U_\theta^{(k+1)} = U_\theta^{(k)} + h S_\theta^{(k)}$
7: **end for**
8: Compute loss $\widehat{\mathcal{L}}(\theta) = \mathcal{J}\big(U_\theta^{(K)}, u^*; f\big)$.
9: Update $\theta$ using $\nabla_\theta \widehat{\mathcal{L}}^{(N)}(\theta)$.

locations. This approach improves generalization across different discretization scales and prevents the model from overfitting to specific grid structures.

**Context Masking.** We sample the context ratio uniformly $r \sim \mathcal{U}[0.05, 0.95]$ and employ mixed masking modes (random and block) to expose the model to diverse conditioning scenarios during training.

### C.4. Gaussian Process Kernel Selection

We employ function-specific GP kernels for function initialization, determined empirically as shown in Table 7.

*Table 7.* Optimal GP kernel configurations per function type

| Function Type | Kernel | $\nu$ | Length Scale |
|---|---|---|---|
| Oscillation | Matérn | 1.5 | 0.1 |
| Damping | Matérn | 2.5 | 0.1 |
| Izhikevich | Matérn | 0.5 | 0.1 |

The Matérn kernel smoothness parameter $\nu$ controls differentiability: $\nu = 0.5$ yields rough samples suitable for high-frequency components, $\nu = 1.5$ produces once-differentiable samples, and $\nu = 2.5$ generates smooth, twice-differentiable samples.

### C.5. Pseudo-algorithm for Training

We present the Algorithm 1 as the pseudo-algorithm for training EBO.

*Table 8.* **MSE under different GP-prior kernels.** Each row is a separately trained model with the same architecture and only the kernel changed. RBF, Matérn $\nu$=1.5, and Matérn $\nu$=2.5 runs trained to 50,000 iterations.

| Kernel | Damping | | | | | |
|---|---|---|---|---|---|---|
| | Super-Res | | | Forecasting | | |
| | 40% | 60% | 80% | 40% | 60% | 80% |
| RBF | 2.16e-04$_{\pm 2.16e-04}$ | 1.83e-04$_{\pm 1.97e-04}$ | 3.52e-04$_{\pm 2.29e-04}$ | 4.29e-04$_{\pm 5.92e-04}$ | 3.54e-04$_{\pm 5.70e-04}$ | 1.66e-04$_{\pm 1.97e-04}$ |
| Matérn $\nu$=1.5 | 3.22e-04$_{\pm 3.67e-04}$ | 2.79e-04$_{\pm 2.56e-04}$ | 5.58e-04$_{\pm 3.83e-04}$ | 4.12e-04$_{\pm 3.78e-04}$ | 2.66e-04$_{\pm 2.75e-04}$ | 2.40e-04$_{\pm 2.95e-04}$ |
| Matérn $\nu$=2.5 | 2.79e-04$_{\pm 2.87e-04}$ | 2.31e-04$_{\pm 2.14e-04}$ | 3.85e-04$_{\pm 2.54e-04}$ | 5.24e-04$_{\pm 7.00e-04}$ | 3.67e-04$_{\pm 4.42e-04}$ | 3.58e-04$_{\pm 5.00e-04}$ |

### C.6. Influence of Kernel Type on Performance

We make an ablation study on the influence of kernel types for the performance of EBO. As shown in Table 8, we see that for damped oscillations which are smooth, smoother kernels have stronger performance in both superresolution and forecasting, validating our assumption that GP prior kernels influence the smoothness of output functions.

## D. PDE Sparse Reconstruction Experiments

This section provides the dataset, training, inference, and evaluation details for the sparse PDE reconstruction experiments. We evaluate Darcy flow, Burgers, and Navier–Stokes at resolution 128 under random sparse-context masks.

### D.1. Task Setup

For each PDE sample, the model receives values at a subset of spatial locations and reconstructs the full solution field. We evaluate context ratios $\{20\%, 40\%, 60\%, 80\%\}$ in the main table and additional context ratios in appendix sweeps. The observed mask is denoted by $M$, where $M_i = 1$ indicates an observed context location and $M_i = 0$ indicates an unobserved location.

The primary metric is missing-region relative $L^2$ error,

$$\mathrm{rel}L^2_{\mathrm{miss}} = \frac{\|(1-M)(\hat{u}-u)\|_2}{\|(1-M)u\|_2 + \epsilon}. \tag{108}$$

This evaluates only the unknown region that must be reconstructed. For visualization, we use context insertion,

$$\hat{u}_{\mathrm{ctx}} = Mu + (1-M)\hat{u}, \tag{109}$$

so the displayed reconstructions preserve known observations.

### D.2. Datasets and Splits

*Table 9.* **Sparse PDE reconstruction datasets.** All main results use resolution 128.

| **Task** | **Resolution** | **Output field** | **Context mask** |
|---|---|---|---|
| Darcy flow | $128 \times 128$ | scalar solution field | random spatial mask |
| Navier–Stokes | $128 \times 128$ | scalar vorticity/field slice | random spatial mask |
| Burgers | 128 | 1D solution field | random 1D mask |

All comparisons use matched task, resolution, context ratio, seed, and test split. Operator and EBO reconstructions are evaluated on the same sparse masks whenever both predictions are available.

### D.3. Operator Baseline

The operator baseline is a context-aware sparse reconstructor. Its input contains observed values, an observed-mask channel, and coordinate channels. Unlike the initial unconstrained operator baseline, the corrected operator is trained to preserve

observed context values through an explicit context-consistency term. The loss is

$$\mathcal{L}_{\mathrm{op}} = \mathcal{L}_{\mathrm{miss}} + \lambda_{\mathrm{ctx}} \mathcal{L}_{\mathrm{ctx}}, \tag{110}$$

where

$$\mathcal{L}_{\mathrm{miss}} = \frac{\sum_i (1 - M_i)(\hat{u}_i - u_i)^2}{\sum_i (1 - M_i) + \epsilon}, \qquad \mathcal{L}_{\mathrm{ctx}} = \frac{\sum_i M_i (\hat{u}_i - u_i)^2}{\sum_i M_i + \epsilon}. \tag{111}$$

We use $\lambda_{\mathrm{ctx}} = 1$ for Darcy and Navier–Stokes. For Burgers, the final reported corrected operator uses the best validation-selected variant. Checkpoints are selected by validation missing-region relative $L^2$.

### D.4. EBO Inference

EBO reconstructs the missing field by iterative energy minimization. Starting from a Gaussian-reference initialization, the candidate field is updated by descending the learned conditional energy. Observed context values are clamped during refinement, so EBO maintains context consistency by construction during inference.

For the reported PDE results, EBO inference is selected using validation data only. We sweep deterministic inference with noise scale 0, step counts in $\{16, 32, 64, 96, 128\}$ when feasible, step-size multipliers in $\{0.1, 0.25, 0.5, 0.75, 1.0\}$, and aggregation rules including best-energy selection, posterior mean, and posterior median over chains. The test-set number reported in the main table uses the validation-selected inference configuration.

*Table 10.* **EBO inference settings used in the PDE context sweep.**

| Task | Context | Selection | Steps | Step multiplier |
|---|---|---|---|---|
| Darcy | 5% | posterior mean | 16 | 1.0 |
| Darcy | 10% | posterior mean | 16 | 1.0 |
| Darcy | 20% | best-energy | 32 | 1.0 |
| Darcy | 40% | posterior mean | 16 | 1.0 |
| Darcy | 60% | posterior mean | 16 | 1.0 |
| Darcy | 80% | posterior mean | 16 | 1.0 |
| Burgers | 5% | best-energy | 96 | 1.0 |
| Burgers | 10% | best-energy | 96 | 0.25 |
| Burgers | 20% | posterior mean | 96 | 0.25 |
| Burgers | 40% | posterior median | 96 | 0.25 |
| Burgers | 60% | best-energy | 96 | 0.5 |
| Burgers | 80% | best-energy | 96 | 0.25 |
| Navier–Stokes | 5% | posterior median | 96 | 0.25 |
| Navier–Stokes | 10% | posterior median | 64 | 0.5 |
| Navier–Stokes | 20% | posterior median | 96 | 0.25 |
| Navier–Stokes | 40% | posterior median | 32 | 0.75 |
| Navier–Stokes | 60% | posterior median | 96 | 0.25 |
| Navier–Stokes | 80% | posterior median | 32 | 0.75 |

### D.5. Full Context Sweep

Table 11 reports the full sparse-reconstruction context sweep. EBO improves all Darcy settings, all Navier–Stokes settings, and the lower-context Burgers settings. The largest gains occur on Darcy, where EBO reduces missing-region relative $L^2$ from $0.174$ to $0.028$ at 10% context and from $0.108$ to $0.026$ at 20% context. Navier–Stokes is more challenging but still benefits from refinement, with reductions from $0.220$ to $0.125$ at 20% context and from $0.192$ to $0.113$ at 40% context.

### D.6. Context Consistency

Both methods are evaluated with the same known observations, but they enforce context consistency differently. The corrected operator receives the observed values and mask as input and is trained with the observed-context loss $\mathcal{L}_{\mathrm{ctx}}$. EBO

*Table 11.* **Full sparse PDE reconstruction context sweep at resolution 128.** We report missing-region relative $L^2$ error. Lower is better; bold marks the lower method per row.

| Task | Context | Operator | EBO | Reduction |
|---|---|---|---|---|
| Darcy | 5% | 0.284 | **0.078** | 72.6% |
| | 10% | 0.174 | **0.028** | 83.7% |
| | 20% | 0.108 | **0.026** | 76.0% |
| | 40% | 0.082 | **0.020** | 75.8% |
| | 60% | 0.055 | **0.019** | 65.9% |
| | 80% | 0.042 | **0.020** | 51.3% |
| Burgers | 5% | 0.474 | **0.152** | 67.9% |
| | 10% | 0.206 | **0.089** | 56.9% |
| | 20% | 0.175 | **0.062** | 64.3% |
| | 40% | 0.074 | **0.042** | 43.9% |
| | 60% | **0.037** | 0.048 | −30.1% |
| | 80% | **0.032** | 0.044 | −37.2% |
| Navier–Stokes | 5% | 0.378 | **0.216** | 42.8% |
| | 10% | 0.269 | **0.168** | 37.4% |
| | 20% | 0.220 | **0.125** | 43.2% |
| | 40% | 0.192 | **0.113** | 41.1% |
| | 60% | 0.179 | **0.116** | 35.1% |
| | 80% | 0.169 | **0.147** | 13.3% |

instead enforces the observed values during refinement by clamping the candidate field at locations where $M_i = 1$. Thus, for both methods the final displayed reconstruction is

$$\hat{u}_{\mathrm{ctx}} = Mu + (1 - M)\hat{u},$$

while the primary metric remains missing-region relative $L^2$. We report missing-region error because observed locations are already known at test time and should not dominate the evaluation.

This distinction is important for sparse reconstruction. The initial unconstrained operator baseline did not reliably preserve observed context values, which could create discontinuities after context insertion. The corrected operator removes this failure mode by adding the explicit context-consistency term. EBO avoids the same issue through hard clamping during the refinement trajectory.

**Burgers high-context behavior.** Burgers is the only PDE benchmark where the corrected operator outperforms EBO in the high-context regime. At 60% context, the operator obtains $0.037$ missing-region relative $L^2$ compared with EBO's $0.048$. At 80% context, the operator obtains $0.032$ compared with EBO's $0.044$. This behavior is consistent with the structure of the task: random-mask 1D Burgers reconstruction becomes substantially easier when most points are observed, and the context-aware supervised operator can exploit the dense context directly. We therefore interpret Burgers as showing a context-dependent regime: EBO is beneficial when the reconstruction remains underdetermined, while the corrected operator can be stronger when high-density observations make one-shot reconstruction sufficient.

### D.7. Interpretation by PDE

Darcy shows the clearest benefit from energy-based refinement. EBO improves every context level, with reductions above 50% for all six evaluated context ratios. The strongest relative gain is at 10% context, where error decreases from $0.174$ to $0.028$, corresponding to an $83.7\%$ reduction. Even at 80% context, where the missing region is small, EBO remains lower-error than the operator baseline.

Navier–Stokes also improves across all evaluated context ratios, but the gains are smaller than Darcy. At 20% and 40% context, EBO reduces missing-region relative $L^2$ by $43.2\%$ and $41.1\%$, respectively. At higher context ratios, the gap narrows: the reduction is $35.1\%$ at 60% and $13.3\%$ at 80%. This suggests that Navier–Stokes remains a harder reconstruction setting, and that the marginal benefit of iterative refinement decreases when the observed context becomes dense.

Burgers is the main context-dependent case. EBO improves the operator at 5%, 10%, 20%, and 40% context, with the 20% setting decreasing from $0.175$ to $0.062$. At 60% and 80% context, the corrected operator becomes better. We therefore use

Burgers as a diagnostic setting rather than an unconditional EBO win: when random 1D observations are dense, one-shot supervised reconstruction can be sufficient.

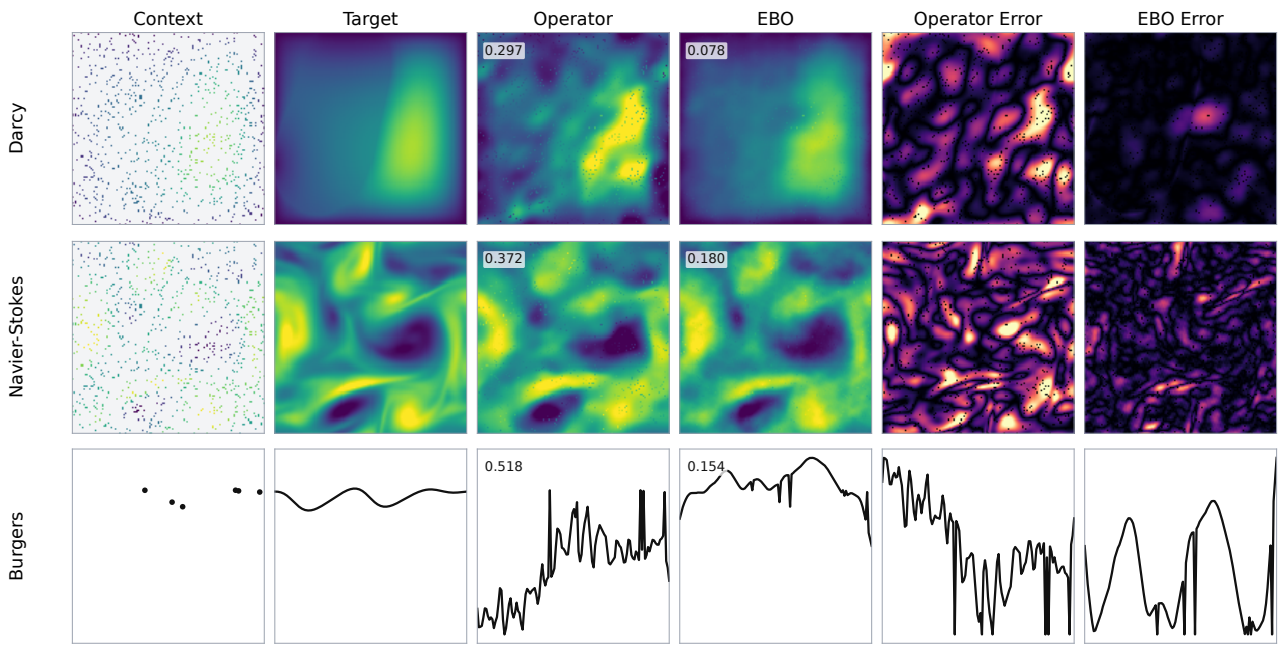


*Figure 9.* **Sparse PDE reconstruction at 5% context using the table-matched Operator and EBO sources.** Numeric annotations show missing-region relative $L^2$ for the displayed sample.

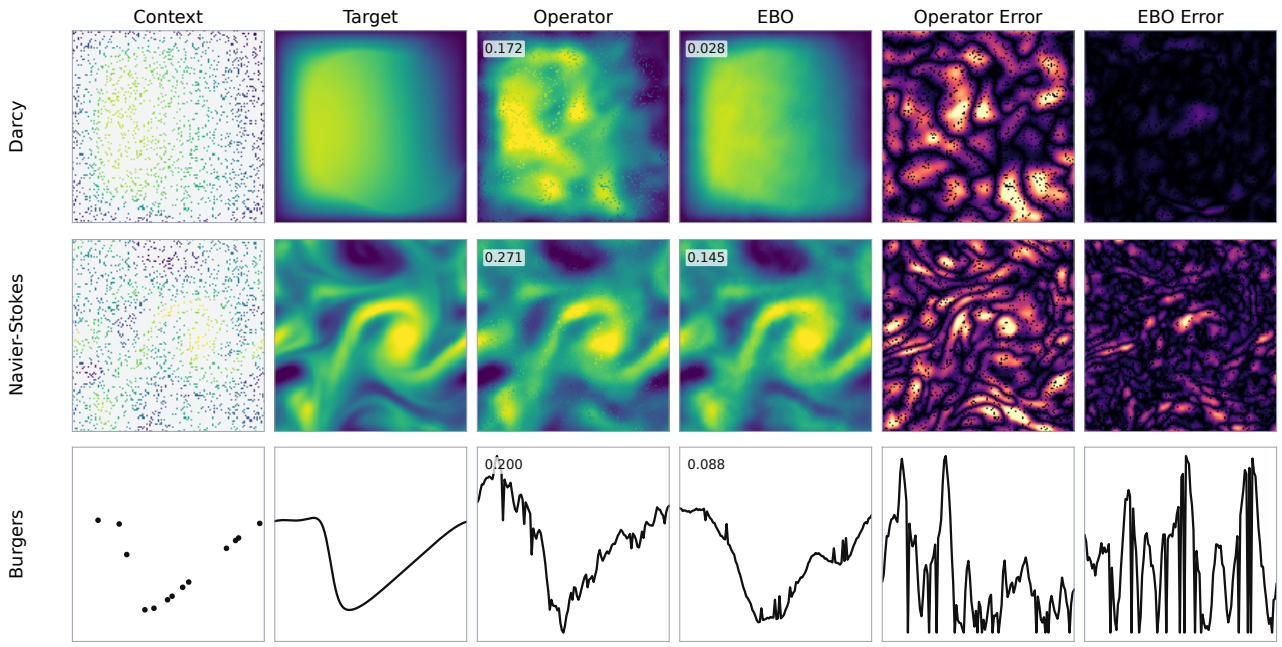


*Figure 10.* Sparse PDE reconstruction at 10% context using the table-matched Operator and EBO sources. Numeric annotations show missing-region relative $L^2$ for the displayed sample.

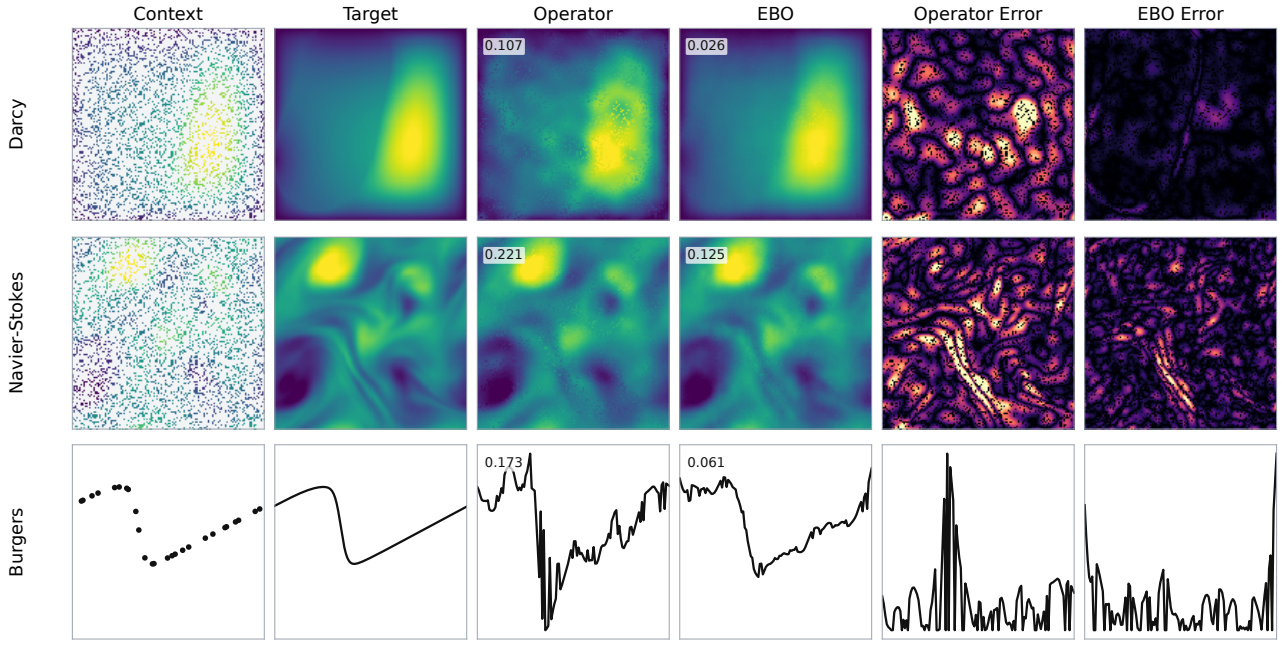


*Figure 11.* Sparse PDE reconstruction at 20% context using the table-matched Operator and EBO sources. Numeric annotations show missing-region relative $L^2$ for the displayed sample.

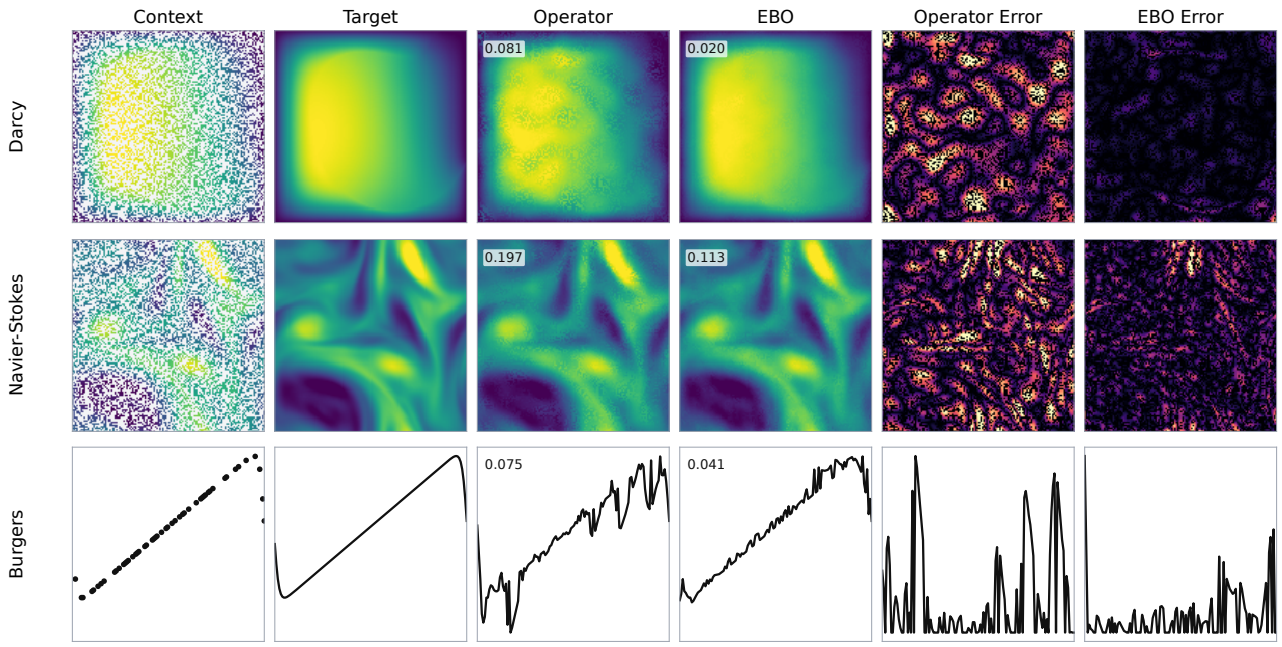


*Figure 12.* Sparse PDE reconstruction at 40% context using the table-matched Operator and EBO sources. Numeric annotations show missing-region relative $L^2$ for the displayed sample.

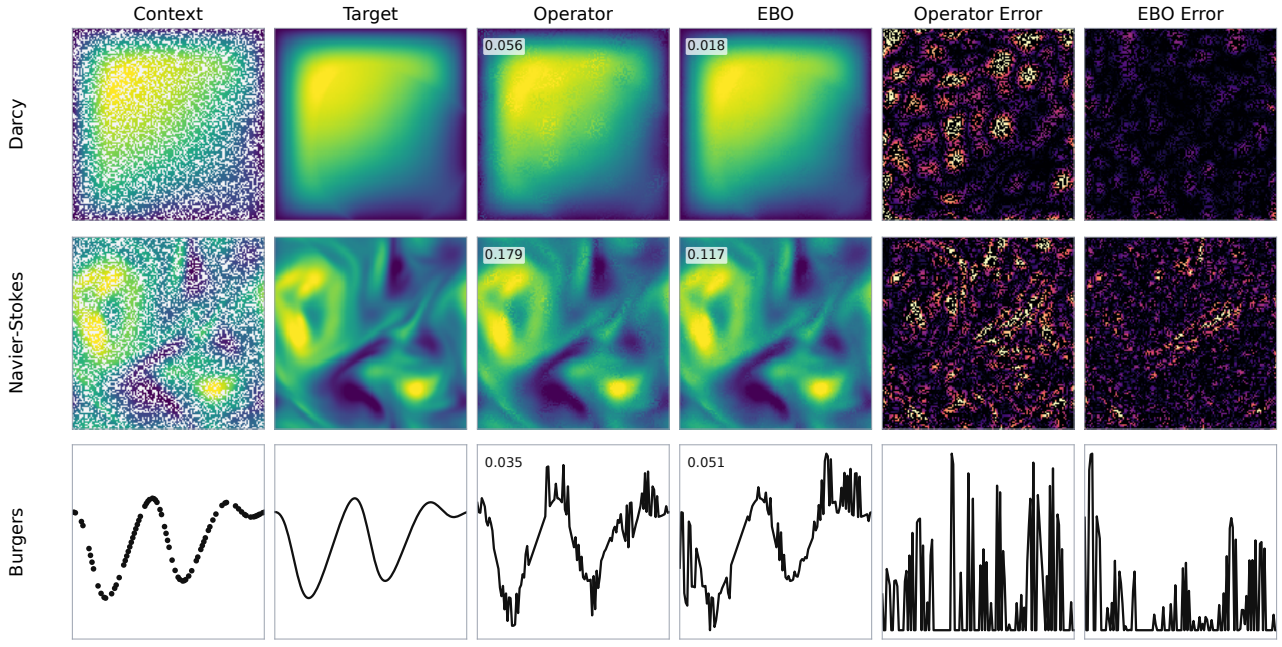


*Figure 13.* Sparse PDE reconstruction at 60% context using the table-matched Operator and EBO sources. Numeric annotations show missing-region relative $L^2$ for the displayed sample.

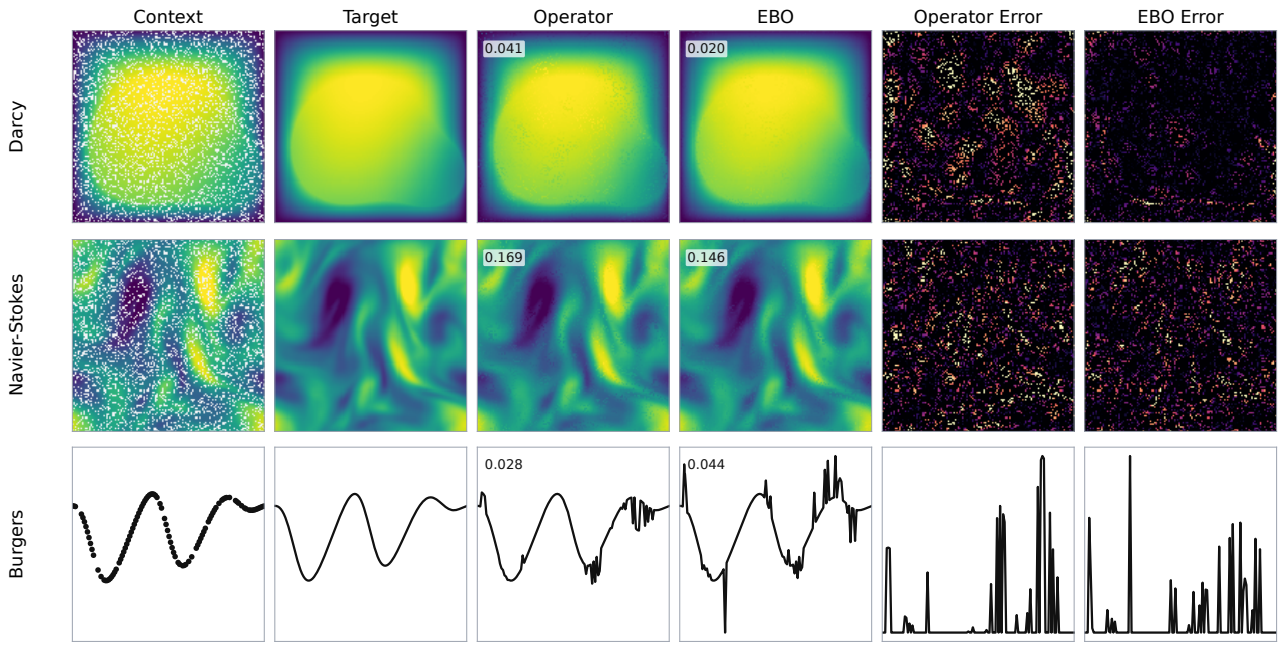


*Figure 14.* Sparse PDE reconstruction at 80% context using the table-matched Operator and EBO sources. Numeric annotations show missing-region relative $L^2$ for the displayed sample.

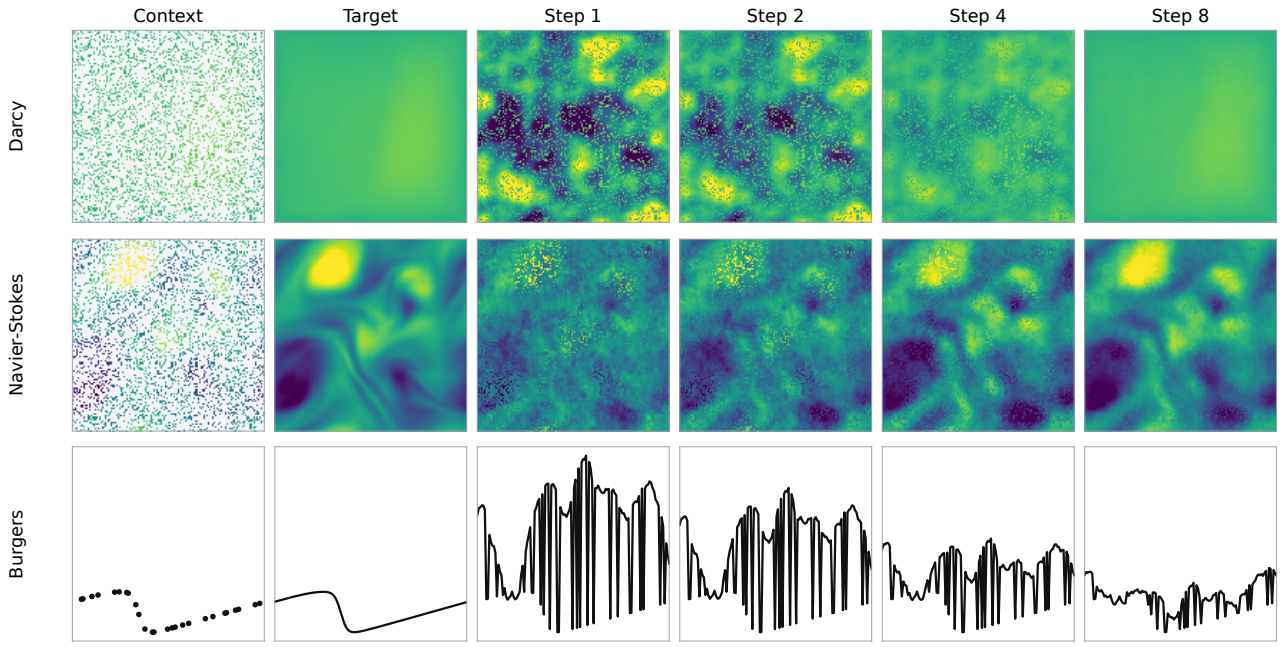


*Figure 15.* EBO energy-minimization reconstruction trajectory at 20% context. Rows show Darcy, Navier–Stokes, and Burgers. Columns show the sparse context, target, and EBO reconstructions after 1, 2, 4, and 8 energy-minimization steps.

# E. TUSZ Dataset and Preprocessing

The Temple University Hospital Seizure Corpus (TUSZ) v2.0.3 (Shah et al., 2018) is the largest publicly available annotated EEG dataset for seizure detection, containing clinical EEG recordings from over 600 patients with expert annotations for seizure onset/offset times and type classifications. Automated seizure detection is clinically significant for continuous EEG monitoring in intensive care units, where timely identification of seizure activity can guide treatment decisions and improve patient outcomes.

We adopt a one-class learning paradigm: the model is trained exclusively on background (non-seizure) EEG segments and learns to model the dynamics of normal brain activity. At inference time, seizure segments—which the model has never seen during training—produce higher energy scores because they deviate from the learned normal distribution. This formulation naturally frames seizure detection as anomaly detection. We use the official train/dev/eval splits provided with the corpus.

## E.1. Seizure Types

The TUSZ corpus employs a hierarchical annotation taxonomy that distinguishes between focal (partial) and generalized seizure types, as well as specific clinical manifestations. Table 12 summarizes 9 seizure type annotations used in the dataset.

*Table 12.* TUSZ seizure type annotations.

| Label | Type | Description |
|---|---|---|
| bckg | Background | Normal, non-seizure activity |
| fnsz | Focal Non-Specific | Focal seizure, unclassified subtype |
| gnsz | Generalized Non-Specific | Generalized seizure, unclassified subtype |
| spsz | Simple Partial | Focal seizure without impaired awareness |
| cpsz | Complex Partial | Focal seizure with impaired awareness |
| absz | Absence | Brief generalized seizure with staring |
| tnsz | Tonic | Sustained muscle stiffening |
| tcsz | Tonic-Clonic | Stiffening followed by rhythmic jerking |
| mysz | Myoclonic | Brief, shock-like muscle jerks |

## E.2. Electrode Montage

We use the standard TCP (Temporal Central Parasagittal) bipolar montage with 22 differential channels. Bipolar montages compute the voltage difference between adjacent electrode pairs, which offers several advantages for clinical EEG analysis: they reduce common-mode artifacts, emphasize local electrical activity by attenuating distant sources, and provide better spatial localization of abnormal discharges. The TCP montage organizes electrodes into anatomically meaningful chains—temporal chains capture activity from the temporal lobes (important for temporal lobe epilepsy), parasagittal chains cover the sensorimotor cortex, and central chains bridge the hemispheres. Files missing any required electrodes are excluded from preprocessing.

*Table 13.* 22-channel TCP bipolar montage organization.

| Chain | Channels | Count |
|---|---|---|
| Left Temporal | FP1-F7, F7-T3, T3-T5, T5-O1 | 4 |
| Right Temporal | FP2-F8, F8-T4, T4-T6, T6-O2 | 4 |
| Central | A1-T3, T3-C3, C3-CZ, CZ-C4, C4-T4, T4-A2 | 6 |
| Left Parasagittal | FP1-F3, F3-C3, C3-P3, P3-O1 | 4 |
| Right Parasagittal | FP2-F4, F4-C4, C4-P4, P4-O2 | 4 |
| **Total** | | **22** |

### E.3. Preprocessing Pipeline

Our preprocessing pipeline consists of frequency filtering, normalization, and segmentation stages designed to prepare clinical EEG data for training the model.

**Frequency Filtering.** Raw EEG signals contain artifacts and noise outside the frequency range of interest for learning neural dynamics. We apply a two-stage filtering process:

- **Bandpass filter (0.5–70 Hz)**: A 4th-order IIR Butterworth filter removes DC drift and slow baseline wander (below 0.5 Hz) as well as high-frequency noise and muscle artifacts (above 70 Hz). The 0.5 Hz lower cutoff preserves slow EEG rhythms relevant to seizure activity while rejecting electrode drift.
- **Notch filter (60 Hz)**: A 4th-order IIR Butterworth notch filter attenuates power line interference, which is ubiquitous in clinical recordings and can obscure EEG features.

**Normalization and Segmentation.** We apply per-file z-score normalization *before* segmentation, computing channel-wise statistics over the entire recording: $\tilde{x}_c(t) = (x_c(t) - \mu_c)/(\sigma_c + \epsilon)$. This approach preserves the relative amplitude relationships between segments from the same recording, which is important because seizure activity often manifests as amplitude changes relative to baseline.

The preprocessed signal is then segmented using non-overlapping windows of $T = 12$s duration with stride $S = 12$s. The 12-second window length provides sufficient temporal context to capture seizure morphology while keeping segment sizes manageable. Segments are labeled based on annotation overlap: a segment receives a seizure label if the maximum overlap with any seizure annotation exceeds the threshold, and background otherwise. Segments with ambiguous overlap are excluded.

*Table 14.* Preprocessing pipeline for TUSZ data.

| Step | Operation | Details |
|---|---|---|
| 1 | Frequency filtering | Bandpass 0.5–70 Hz (IIR Butterworth order 4); Notch 60 Hz |
| 2 | Channel extraction | Compute 22 bipolar channels; standardize names by removing reference suffixes (e.g., "-LE", "-REF") |
| 3 | Normalization | Per-file z-score: $\tilde{x}_c(t) = (x_c(t) - \mu_c)/(\sigma_c + \epsilon)$ applied before segmentation |
| 4 | Segmentation | Sliding window: $T = 12$s duration, $S = 12$s stride (non-overlapping) |
| 5 | Label assignment | Seizure label if annotation overlap $\geq$ threshold; otherwise background |
| 6 | Coordinates | Time coordinates $\mathbf{t} = \{0, \Delta t, \ldots, T\}$ with $\Delta t = 1/f_s$ |

### E.4. Output Format and Statistics

*Table 15.* Preprocessed segment format.

| Field | Shape | Description |
|---|---|---|
| `data` | $(22, N)$ | Normalized EEG values |
| `coords` | $(N,)$ | Time coordinates in seconds |
| `label` | scalar | Integer label (0=bckg, 1–8=seizure) |
| `label_name` | string | Seizure type name |
| `sfreq` | scalar | Sampling frequency (Hz) |
| `n_samples` | scalar | Number of samples $N$ |

Due to the variability of medical measurement setups, each preprocessed segment has a different sampling frequency (250–1000 Hz) across recordings. Since we cut segments based on a fixed time window (12s), the number of samples $N$

differs across segments. For example, at 250 Hz, a 12s segment contains 3000 samples; at 1000 Hz, it contains 12000 samples.

*Table 16.* Sequence length statistics (samples per 12-second segment).

| Split | Mean | Std | Median | Min | Max |
|---|---|---|---|---|---|
| Train | 3323 | 1404 | 3072 | 1500 | 12000 |
| Dev | 3355 | 1514 | 3072 | 1500 | 12000 |
| Eval | 3346 | 1561 | 3072 | 1536 | 12000 |

### E.5. Training Configuration

Table 17 summarizes the training configuration for EBO on the TUSZ dataset.

*Table 17.* EBO training configuration for seizure detection.

| Parameter | Value |
|---|---|
| Input channels | 22 |
| Hidden size / Depth / Heads | 256 / 4 / 4 |
| Learning rate | $1 \times 10^{-4}$ |
| Batch size | 4 |
| Position encoding | SIREN |
| EM steps | 1–32 (randomized during training) |
| EM step size | 0.01 (learnable, $100\times$ LR multiplier) |
| GRF kernel | Matérn ($\nu = 0.5$, $\ell = 0.1$, $\sigma = 1.0$) |
| Noise annealing | Cosine schedule |

## F. LOB Dataset and Preprocessing

Limit Order Book (LOB) data captures the full depth of buy and sell orders in financial markets, providing a comprehensive view of market microstructure at the tick level. LOB data is particularly valuable for understanding market dynamics because it reveals not just executed trades but also the latent supply and demand at various price levels.

We preprocess high-frequency LOB snapshots into fixed-format segments suitable for energy-based anomaly detection, with realized volatility (RV) as the target for regime shift detection. Unlike the TUSZ dataset where we employ one-class learning (training only on normal data), for LOB we use an *unsupervised* paradigm: the model trains on all market segments without label information, learning to model the joint distribution of LOB features. At inference time, we examine whether the model's energy correlates with future realized volatility—high energy indicating that the current market state deviates from typical patterns and may precede volatile periods.

### F.1. Data Collection

We collected high-frequency limit order book data from the Korea Exchange (KRX) futures market, specifically focusing on KOSDAQ150 futures front-month contracts. The KOSDAQ150 futures contract is a derivative instrument based on the KOSDAQ150 index, which tracks 150 representative stocks listed on the KOSDAQ market. The data was gathered using a custom-developed collection program designed to capture the complete market microstructure at tick-level granularity.

The dataset spans two temporally separated periods: a training set from July 2, 2024 to December 12, 2024 (approximately 5.5 months), and a test set from January 17, 2025 to March 13, 2025 (approximately 2 months). This temporal separation ensures out-of-sample evaluation and prevents information leakage from future market conditions into the training process.

We focused exclusively on the regular trading session (8:45–15:35 KST), excluding the opening and closing auction periods. The auction mechanisms employ different price discovery processes (call auction vs. continuous matching) that would introduce heterogeneity in the order book dynamics, making them unsuitable for our analysis of continuous market behavior.

Our collection program recorded complete limit order book snapshots at every market event, capturing the full depth of the order book whenever the market state changed. The recorded events include new limit order submissions, order cancellations, market order arrivals, and trade executions. This event-driven approach preserves the complete information flow of the market, capturing both the evolution of latent liquidity (through limit order submissions and cancellations) and realized demand (through market orders and executions). For each event, we recorded the top 5 price levels on both the bid and ask sides of the order book, along with the corresponding order quantities at each level.

### F.2. Feature Channels

We extract 21 feature channels from raw LOB data, capturing price levels, order volumes, and returns. The features are designed to be scale-invariant and comparable across different assets and time periods.

**Price Features (Channels 0–9).** Price features encode the distance of each order book level from the mid-price, normalized by the tick size $\tau = 10$ and log-transformed:

$$\text{Bid Price}^{(i)} = \ln\left(\frac{|P_{\text{mid}} - P_{\text{bid}}^{(i)}|}{\tau} + 1\right) \tag{112}$$

$$\text{Ask Price}^{(i)} = \ln\left(\frac{|P_{\text{ask}}^{(i)} - P_{\text{mid}}|}{\tau} + 1\right) \tag{113}$$

where $P_{\text{mid}} = (P_{\text{bid}}^{(1)} + P_{\text{ask}}^{(1)})/2$ is the mid-price. The log transformation compresses the dynamic range of price distances, making features from different market conditions (tight vs. wide spreads) more comparable. The tick size normalization ($\tau = 10$) converts absolute price differences to tick units.

**Volume Features (Channels 10–19).** Volume features capture the liquidity available at each price level using log-transformed order amounts:

$$\text{Amount}^{(i)} = \ln(A^{(i)} + 1) \tag{114}$$

The log transformation is essential because order sizes span multiple orders of magnitude (from single-lot retail orders to large institutional blocks), and raw volumes would dominate the feature space.

**Return Feature (Channel 20).** The log return captures price momentum at the tick level:

$$r_t = \ln\left(\frac{P_{\text{mid}}^{(t)}}{P_{\text{mid}}^{(t-1)}}\right), \quad P_{\text{mid}} = (P_{\text{bid}}^{(1)} + P_{\text{ask}}^{(1)})/2 \tag{115}$$

Log returns are preferred over simple returns because they are additive over time and approximately symmetric for small price changes, providing a stationary transformation of the non-stationary price process.

*Table 18.* LOB feature channels (21 total). Tick size $\tau = 10$.

| Channels | Name | Formula |
|---|---|---|
| 0–4 | Bid Price (1–5) | $\ln(\lvert P_{\text{mid}} - P_{\text{bid}}^{(i)}\rvert/\tau + 1)$ |
| 5–9 | Ask Price (1–5) | $\ln(\lvert P_{\text{ask}}^{(i)} - P_{\text{mid}}\rvert/\tau + 1)$ |
| 10–14 | Bid Amount (1–5) | $\ln(A_{\text{bid}}^{(i)} + 1)$ |
| 15–19 | Ask Amount (1–5) | $\ln(A_{\text{ask}}^{(i)} + 1)$ |
| 20 | Log Return | $\ln(P_{\text{mid}}^{(t)}/P_{\text{mid}}^{(t-1)})$ |

### F.3. Target: Realized Volatility

For each segment, we compute the Realized Volatility (RV) of the *next* 10-second time window as the target label:

$$\text{RV} = \sqrt{\sum_{k=1}^{K} r_k^2} \tag{116}$$

where $r_k = \ln(P_{\text{mid}}^{(k)}/P_{\text{mid}}^{(k-1)})$ are the log returns in the future window, and $K$ is the number of events. High RV indicates volatile periods (potential regime shifts), while low RV indicates calm markets.

It is important to note that our energy-based model does not directly predict RV as a regression target. Instead, the model learns the distribution of LOB feature sequences in an unsupervised manner, and we evaluate whether the learned energy function correlates with future volatility. The hypothesis is that market states preceding high-volatility periods may exhibit distinctive patterns (e.g., order book imbalances, unusual spread dynamics) that the energy model assigns higher energy to, as these states deviate from typical market conditions.

### F.4. Preprocessing Pipeline

*Table 19.* LOB preprocessing pipeline.

| Step | Operation | Details |
|---|---|---|
| 1 | Time parsing | Convert timestamp (HHMMSSCC format) to seconds from midnight |
| 2 | Trading hours filter | Keep events within 8:45–15:35 (valid trading session) |
| 3 | Sliding window | $T = 10$s window duration, $S = 1$s stride; variable events per window |
| 4 | Minimum events filter | Discard windows with $< 10$ events |
| 5 | Feature extraction | Compute 21 channels per event (Table 18) |
| 6 | Target computation | RV of next 10s window |
| 7 | Coordinate generation | Relative time $t - t_{\text{start}}$ for each event |

The LOB preprocessing pipeline converts raw tick data into fixed-format segments suitable for neural network training.

**Time parsing and filtering.** Raw timestamps are stored in HHMMSSCC format (hours, minutes, seconds, centiseconds). We parse these to seconds from midnight and filter to keep only events within the core trading session (8:45–15:35), excluding the opening and closing auction periods where price discovery mechanisms differ from continuous trading.

**Sliding window segmentation.** We use overlapping windows with $T = 10$s duration and $S = 1$s stride. The shorter stride creates more training samples and provides finer temporal resolution for detecting regime transitions. Windows with fewer than 10 events are discarded to ensure sufficient information content.

**Variable sequence lengths.** Each window contains a variable number of events depending on market activity—quiet periods may have only 10–20 events per 10 seconds, while active periods can have hundreds. This natural variation is preserved in our preprocessing, and the model handles variable-length inputs through subsampling to a fixed length at training time.

### F.5. Output Format and Statistics

*Table 20.* Preprocessed LOB segment format.

| Field | Shape | Description |
|---|---|---|
| `data` | $(21, N)$ | Feature channels (log-transformed) |
| `coords` | $(N,)$ | Relative time coordinates (seconds) |
| `target` | scalar | RV of next 10s window |
| `n_samples` | scalar | Number of events $N$ in window |
| `n_future` | scalar | Number of events in future window |
| `window_start` | scalar | Absolute start time (seconds) |

### F.6. Training Configuration

Table 22 summarizes the training configuration for EBO on the LOB dataset.

*Table 21.* Dataset and sequence length statistics (events per 10-second window).

| Split | Segments | Mean | Std | Median | Min | Max |
|---|---|---|---|---|---|---|
| Train | 2,517,168 | 71.4 | 55.8 | 56 | 10 | 1230 |
| Test | 824,883 | 70.3 | 57.4 | 54 | 10 | 867 |

*Table 22.* EBO training configuration for LOB volatility detection.

| Parameter | Value |
|---|---|
| Input channels | 21 (all LOB features) |
| Hidden size / Depth / Heads | 256 / 4 / 4 |
| Learning rate | $1 \times 10^{-5}$ |
| Batch size | 16 |
| Epochs | 3 |
| Position encoding | SIREN |
| EM steps | 1–32 (randomized during training) |
| EM step size | 0.01 (learnable, $10\times$ LR multiplier) |
| GRF kernel | Matérn ($\nu = 1.5$, $\ell = 0.1$, $\sigma = 1.0$) |
| Noise annealing | Cosine schedule |

### F.7. Evaluation Metrics

For volatility regime detection, we evaluate the model's ability to identify market states that precede high-volatility periods using two metrics:

**Energy–Volatility Correlation.** We compute Spearman correlation coefficients between energy and future realized volatility. Spearman correlation is robust to outliers and captures monotonic relationships. A positive correlation indicates that the model assigns higher energy to market states preceding volatile periods.

**Decile Analysis.** We partition test segments into 5 groups based on their future energy percentiles (0–20%, 20–40%, ..., 80–100%) and examine the distribution of future RV within each decile using box plots.