SYMBOL-CONDITIONED ADAPTIVE FOURIER OPERATOR TRANSFORMER FOR LARGE-SCALE PDE PRE-TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Neural operators have emerged as a promising paradigm for solving PDEs and building PDE foundation models. However, existing architectures often suffer from spectral bias and struggle to capture high-frequency dynamics, limiting their capacity to model complex physical systems. To address these challenges, we introduce the Symbol-conditioned Adaptive Fourier Pre-training Operator Transformer (SAFPOT). First, we define multi-level frequency bands to represent heterogeneous spectral regions, enabling full-frequency learning. First, the multi-level frequency bands are defined to represent heterogeneous spectral regions, enabling full-frequency learning. Second, we design a novel Symbol-conditioned Adaptive Fourier (SAF) module, which dynamically modulates low-, mid-, and high-frequency bands via a frequency gate network. Together, these designs allow the model to adapt its spectral modeling strategy to the characteristics of each governing equation and effectively capture multi-scale physical dynamics. Finally, we pre-train models with parameters from 40M to 0.5B on 6 PDE datasets. In data-oriented evaluation, pre-trained SAFPOT with only 28M activated parameters surpasses fully fine-tuned baselines and further reduces L2RE by up to 71.75% after PDE-specific fine-tuning. In physics-oriented evaluation, SAFPOT reduces the average fRMSE by up to 59.4% compared with the strongest baseline and substantially reduces errors across each frequency band.

1 INTRODUCTION

Simulating real-world physical systems and problems in biology, thermodynamics, astrophysics, weather and climate modeling, and hydrodynamics, to name a few, often involves modeling and solving partial differential equations (PDEs). Neural operators (NOs) (Li et al., 2021; Lu et al., 2021; Kovachki et al., 2023) learn mappings between function spaces and provide a promising approach to efficiently solving complex physical systems. Fourier Neural Operator (FNO) (Li et al., 2021), which is one of the effective instantiations of a NO, learns the operator through Fourier-domain parameterization of the integral kernel. Despite their strong performance, these models are typically designed for a single PDE family and lack generalization across heterogeneous PDEs.

To overcome this task-specific paradigm, recent studies have extended large-scale pre-training to neural operators, leveraging diverse PDE datasets to learn generalizable foundation models. DPOT (Hao et al., 2024) introduces autoregressive denoising pre-training with a scalable Fourier-based architecture to learn transferable representations across diverse PDEs. Further, MoE-POT (Wang et al., 2025) employs a learnable routing mechanism to dynamically activate experts, enabling model capacity expansion without a proportional increase in computational cost. These advances move task-specific solvers toward unified multi-PDE learning, enabling them to acquire generalizable knowledge across diverse PDEs.

Unfortunately, existing architectures often suffer from spectral bias and struggle to capture high-frequency dynamics, limiting their capacity to model complex physical systems. Typically, Fourier-based methods retain only a fixed number of low-frequency modes while truncating high-frequency components (Kovachki et al., 2023; Kalimuthu et al., 2025; Yue et al., 2025). This static spectral truncation fundamentally caps the model's ability to resolve fine-scale structures. This limitation is

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

severely amplified in large-scale PDE pre-training: heterogeneous governing equations, coefficients, and boundary conditions induce fundamentally different spectral distributions across datasets (see Section 4.3). Moreover, the solution state alone may be insufficient to disambiguate the underlying dynamics and their associated spectral characteristics. Consequently, a model conditioned only on physical states must infer an appropriate frequency response, making it difficult to adapt spectral processing reliably across diverse PDE systems.

To tackle these challenges, we propose the **S**ymbol-conditioned **A**daptive **F**ourier **P**re-training **O**perator **T**ransformer (SAFPOT). First, rather than simply truncating frequencies, we define multi-level frequency bands representing heterogeneous spectral regions, enabling full-frequency learning. Second, we design a novel Symbol-conditioned Adaptive Fourier (SAF) module, which dynamically modulates these low-, mid-, and high-frequency bands via a frequency gate network. By fusing the evolving physical states with encoded PDE symbols, SAF explicitly bridges symbolic equation representations with frequency-domain learning. Together, these designs allow the model to adapt its spectral modeling strategy to the characteristics of each governing equation, effectively capturing multi-scale physical dynamics. Our contributions are summarized as follows:

- We propose SAFPOT, a pre-training operator model that unifies cross-PDE frequency learning. By defining multi-level frequency bands, SAFPOT preserves heterogeneous spectral regions and enables full-frequency learning across diverse PDEs.
- We design a novel SAF module, which dynamically modulates multi-level frequency bands via a frequency gate network. This design allows the model to adapt its spectral modeling strategy to the characteristics of each governing equation and effectively capture multi-scale physical dynamics.
- Extensive experiments demonstrate the effectiveness of SAFPOT. (1) Under data-oriented evaluation, pre-trained SAFPOT with the 28M activated parameter surpasses fully fine-tuned baselines. After PDE-specific fine-tuning, our model consistently achieves SOTA performance, reducing L2RE by up to 71.75%; (2) Under physics-oriented evaluation, our model substantially reduces errors across the low-, mid-, and high-frequency bands, lowering the average fRMSE by up to 59.4% over the best competing baseline.

2 Related work

Neural Operators and Spectral Modeling. NOs (Li et al., 2021; Lu et al., 2021; Kovachki et al., 2023; Liu et al., 2025) learn mappings between function spaces for PDE solving. FNO (Li et al., 2021) parameterizes integral kernels in the Fourier domain and has inspired extensive work on spectral operator learning (Guibas et al., 2022; Liu et al., 2023; Gao et al., 2025). Yet many FNO-based methods retain only a fixed set of low-frequency modes, limiting their ability to model broader spectral content. Recent studies address this limitation with more expressive spectral parameterizations. Amortized FNO (Xiao et al., 2024) introduces a learnable amortized parameterization of the Fourier kernel to capture high-frequency information without explicit spectral truncation. LOGLO-FNO (Kalimuthu et al., 2025) incorporates auxiliary branches for local spectral convolution and high-frequency feature propagation. SirenFNO (Shi et al., 2026) leverages sinusoidal representation networks for mode-wise kernel parameterization, enabling full-spectrum learning with a discretization-independent parameter count. Differently, our work explores symbol-conditioned adaptive spectral modeling to accommodate the diverse frequency characteristics of different equations.

PDE foundation models. Large-scale pre-training has been increasingly explored in PDE operator learning to develop foundation models with strong cross-equation transferability (Hao et al., 2024; McCabe et al., 2024; Wang et al., 2025). Recent studies have investigated different strategies for learning transferable representations across diverse PDE systems. MPP (McCabe et al., 2024) adopts an autoregressive pre-training framework for time-dependent PDE datasets. UPS (Shen et al., 2024) uses pre-trained large language models for cross-modal alignment to construct a unified multi-PDE solver. DPOT (Hao et al., 2024) introduces autoregressive denoising pre-training with a scalable Fourier-based architecture for transferable representations across PDEs. Further, MoE-POT (Wang et al., 2025) employs a routing mechanism to activate experts, enabling model capacity expansion without a proportional increase in computational cost. However, different PDEs exhibit distinct spectral characteristics that are not explicitly considered in existing approaches. SAFPOT leverages symbolic PDE information to adaptively modulate representations across different frequency bands.

3 METHOD

Problem formulation. We consider parameterized PDEs on $(\boldsymbol{x}, t) \in \Omega \times \mathcal{T} \subset \mathbb{R}^{d+1}$,

$$\frac{\partial \boldsymbol{u}}{\partial t} - \mathcal{F}[\boldsymbol{u}; \theta](x, t) = 0, \tag{1}$$
$$\boldsymbol{u}(x, 0) = \boldsymbol{u}^0(x),\ x \in \Omega, \qquad \mathcal{B}[\boldsymbol{u}](x, t) = 0,\ x \in \partial\Omega.$$

where $\boldsymbol{u}$ is a time-varying function, $\mathcal{F}[\boldsymbol{u}; \theta](x, t) = \mathcal{F}\big(t, x, \boldsymbol{u}, \partial_x \boldsymbol{u}, \partial_{xx} \boldsymbol{u}, \ldots; \theta\big)$ denotes a differential operator involving spatial derivatives, and $\theta \in \Theta$ denotes the parameters specifying the PDE type and coefficients. The functions $\boldsymbol{u}^0$ and $\mathcal{B}$ are the initial and boundary conditions, respectively.

Based on the above formulation, we represent a multi-PDE dataset as $\mathcal{D} = \bigcup_{k=1}^{K} \mathcal{D}_k$, where $\mathcal{D}_k = \{\boldsymbol{u}_i\}_{i=1}^{N_k}$ contains N_k solution trajectories generated by the k-th PDE. Each solution trajectory is discretized over a spatiotemporal mesh as $\boldsymbol{u}_i = \{\boldsymbol{u}_i^1, \ldots, \boldsymbol{u}_i^T\}$, where $\boldsymbol{u}_{i,j}^t = \{(x_j,\ \boldsymbol{u}_j^t) : x_j \in X_i\}$ for $1 \le t \le T$. Moreover, for each PDE, we construct an expression tree using $\mathcal{V}_k = \{\boldsymbol{u}, \partial_t \boldsymbol{u}, \partial_x \boldsymbol{u}, \partial_y \boldsymbol{u}, \partial_{xx} \boldsymbol{u}, \partial_{yy} \boldsymbol{u}, \partial_{xy} \boldsymbol{u}\}$, together with operators and equation coefficients. The tree is then linearized in Polish notation to obtain the symbolic sequence $\mathcal{S}_k$.

Our goal is to learn a neural operator $\mathcal{G}_w$ that jointly takes the previous T_{in} solution fields and the PDE symbolic representations $\mathcal{S}_k$ as inputs to predict the next solution state:

$$\hat{\boldsymbol{u}}_i^{t+1} = \mathcal{G}_w\left(\boldsymbol{u}_i^{t-T_{\text{in}}+1}, \ldots, \boldsymbol{u}_i^t; \mathcal{S}_k\right), \qquad \boldsymbol{u}_i \in \mathcal{D}_k, \tag{2}$$

where w denotes the learnable parameters, and future solution states are predicted autoregressively.

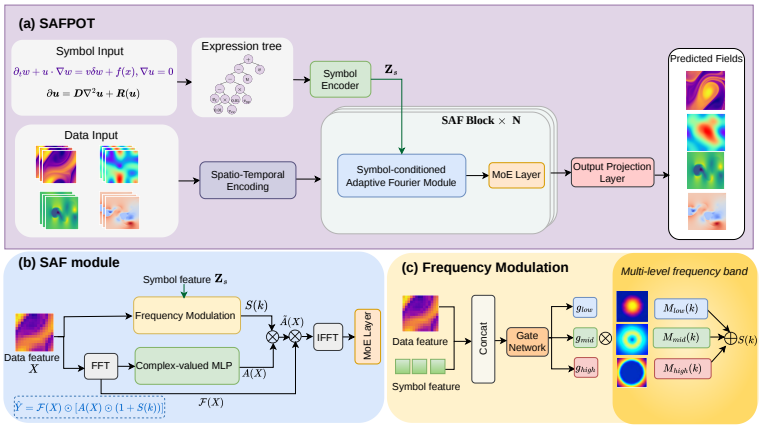


Figure 1: Overview architecture. Spatiotemporal solution fields and symbolic PDE representations are encoded by two parallel branches and processed through N SAF blocks. In each block, Fourier-transformed data features are modulated by a symbol-conditioned correction $\mathbf{S}(k)$ through a frequency gate network, followed by the MoE layer.

3.1 MODEL ARCHITECTURE

The key component of SAFPOT is the SAF module, which employs a frequency gate network to adaptively modulate multi-level frequency bands using joint data and symbolic representations. A visual overview is provided in Fig. 1. Next, we introduce the main components of the architecture and explain how they enable adaptive full-frequency modeling across heterogeneous PDEs.

Data branch. Given a sequence of historical solution fields $\boldsymbol{u}^{<T} \in \mathbb{R}^{B\times H\times W\times T\times C}$, each state $\boldsymbol{u}^t$ is divided into non-overlapping $p \times p$ patches and projected into a D-dimensional latent space: $Z_p^t = \mathcal{P}(\boldsymbol{u}^t) + E_{pos},\ \ 1 \le t \le T$. Here, $\mathcal{P}$ denotes the patch embedding operation and E_{pos} is the learnable positional embedding. The output $Z_p^t \in \mathbb{R}^{B\times H/p\times W/p\times D}$ is then aggregated across historical timesteps through a learnable temporal transformation: $X_{agg} = \sum_{t=1}^{T} W_t \cdot \boldsymbol{z}_p^t$, where $\boldsymbol{z}_p^t$

denotes a local latent feature in $Z_p^t = [\boldsymbol{z}_p^t]_{i,j}$. This temporal aggregation captures the evolution of local dynamics and produces a compact representation for subsequent spectral modeling.

Symbol branch. Given the symbolic representation $\mathcal{S}_k$, the equation is serialized into a Polish-notation token sequence and encoded by a transformer-based symbolic encoder. The resulting representation is subsequently fed into the symbol-conditioned frequency modulation module to generate PDE-specific spectral modulation response, enabling the spectral transformation to adapt to different governing equations.

SAF Block. The latent data representation is progressively updated through N stacked SAF blocks. Each block consists of a SAF module followed by an MoE layer. Under the symbol embedding, the SAF module performs an adaptive spectral transformation and reconstructs the resulting features in the spatial domain via an inverse Fourier transform. The MoE layer then further processes these latent features. Each MoE layer comprises two always-active shared experts and 16 routed experts; the router dynamically activates the top-4 routed experts for each input. After N SAF blocks, the final latent feature is mapped back to the physical space through an output projection layer to obtain the predicted future states. The detailed design of SAF is presented in the following section.

3.2 Symbol-conditioned Adaptive Fourier Modeling

To overcome spectral bias and achieve adaptive spectral modeling for heterogeneous PDEs, we propose the SAF module. It introduces state-driven spectral responses and symbol-conditioned frequency modulation into the standard Fourier layer.

Let $\mathbf{X}_l \in \mathbb{R}^{B\times H\times W\times D}$ denote the input feature of the l-th SAF block. For a translation-invariant kernel, the corresponding Fourier operator can be written as

$$(\mathcal{K}_\phi \mathbf{X}_l)(x) = \mathcal{F}^{-1}\left[R_\phi \cdot \mathcal{F}[\mathbf{X}_l]\right], \tag{3}$$

where $R_\phi(k) \in \mathbb{C}^{D\times D}$ is a frequency-dependent learnable transformation, and $\mathcal{F}/\mathcal{F}^{-1}$ denote the Fourier transform and its inverse. The key distinction lies in the construction of $R_\phi(k)$. Prior Fourier-based methods (Li et al., 2021; Hao et al., 2024; Wang et al., 2025) use input-independent spectral weights, whereas SAF derives $R_\phi(k)$ adaptively from the current Fourier features and further modulates it with the symbolic PDE representation.

Specifically, we first obtain the Fourier representation $\hat{\mathbf{X}}_l = \mathcal{F}(\mathbf{X}_l)$ via a 2D FFT. The state-driven spectral response is then dynamically generated as $\mathbf{A}_l = \Psi_\theta(\hat{\mathbf{X}}_l)$, where Ψ_θ is a learnable frequency-domain mapping parameterized by a two-layer complex-valued MLP:

$$\mathbf{A}_l = \mathbf{W}_2\,\sigma\Big(\mathbf{W}_1\hat{\mathbf{X}}_l + \mathbf{b}_1\Big) + \mathbf{b}_2, \tag{4}$$

where $\mathbf{W}_1$ and $\mathbf{W}_2$ are learnable complex-valued mappings, $\mathbf{b}_1$ and $\mathbf{b}_2$ are the corresponding biases, and σ denotes the activation function. For computational efficiency, we implement this mapping using a multi-head structure, detailed in Appendix B.1.

By the convolution theorem, applying the generated spectral response to the input spectrum is equivalent to an input-dependent convolution in the spatial domain:

$$\mathcal{F}^{-1}\left[\mathbf{A}_l \odot \hat{\mathbf{X}}_l\right] = \boldsymbol{\kappa}_l(\mathbf{X}_l) * \mathbf{X}_l, \qquad \boldsymbol{\kappa}_l(\mathbf{X}_l) = \mathcal{F}^{-1}\left[\Psi_\theta(\mathcal{F}(\mathbf{X}_l))\right]. \tag{5}$$

Here, $\mathbf{A}_l$ can be interpreted as an input-dependent spectral filter. The corresponding spatial kernel is dynamically determined by the current physical state. However, the physical state alone may be insufficient to capture heterogeneous governing dynamics. We therefore incorporate equation-level information through the symbolic PDE representation to further condition the spectral response.

Symbol-conditioned frequency modulation. To represent heterogeneous spectral regions and enable full-frequency learning, SAF introduces multi-level frequency bands. Specifically, the low-, mid-, and high-frequency levels are instantiated as three soft frequency masks, which provide a smooth decomposition of the Fourier domain rather than hard spectral partitions. SAF then dynamically modulates these multi-level bands by predicting their coefficients through a frequency gate network.

Definition 1 (Soft frequency masks) *For a frequency location $k = (k_x, k_y)$, define the normalized radial frequency as $\rho(k) = \sqrt{k_x^2 + k_y^2}/(\max(\sqrt{k_x^2 + k_y^2}) + \epsilon)$. Let ρ_l and ρ_m denote the transition frequencies between adjacent bands, with $\rho_l < \rho_m$. The three soft frequency bands corresponding to low-, high-, and mid-frequency regions are given by*

$$M_{low}(k) = \sigma\left(\frac{\rho_l - \rho(k)}{\tau}\right), \; M_{high}(k) = \sigma\left(\frac{\rho(k) - \rho_m}{\tau}\right), \tag{6}$$
$$M_{mid}(k) = \sigma\left(\frac{\rho(k) - \rho_l}{\tau}\right)\sigma\left(\frac{\rho_m - \rho(k)}{\tau}\right),$$

where τ controls the smoothness of the transitions, and σ denotes the sigmoid function. Smaller values of $\rho(k)$ correspond to large-scale low-frequency components, while larger values indicate fine-scale high-frequency components.

Based on these frequency masks, SAF computes the modulation coefficients jointly from the current data features $\mathbf{X}_l$ and symbolic representation $\mathbf{Z}_s \in \mathbb{R}^{B\times L\times S}$:

$$[g_l^{low}, g_l^{mid}, g_l^{high}] = \lambda \tanh(\text{MLP}([\text{GAP}(\mathbf{X}_l), \mathbf{Z}_s])), \tag{7}$$

where $\text{GAP}(\cdot)$ denotes global average pooling, and λ controls the modulation magnitude.

Finally, the symbol-conditioned frequency correction is constructed by combining the conditioned coefficients with the soft frequency masks:

$$\mathbf{S}_l(k; \mathbf{X}_l, \mathbf{Z}_s) = g_l^{low} M_{low}(k) + g_l^{mid} M_{mid}(k) + g_l^{high} M_{high}(k). \tag{8}$$

In this way, $\mathbf{S}_l(k; \mathbf{X}_l, \mathbf{Z}_s)$ provides a smooth frequency correction, enabling our model to adaptively emphasize or suppress different spectral regions across heterogeneous PDEs.

Symbol-conditioned Adaptive Fourier module. With the symbol-conditioned correction $\mathbf{S}_l(k; \mathbf{X}_l, \mathbf{Z}_s)$, the spectral response $\mathbf{A}_l$ is further adjusted to obtain the final adaptive response:

$$\tilde{\mathbf{A}}_l(k) = \mathbf{A}_l(k) \odot [1 + \mathbf{S}_l(k; \mathbf{X}_l, \mathbf{Z}_s)]. \tag{9}$$

The factor $1 + \mathbf{S}_l$ provides a *symbol-conditioned frequency correction* of the *state-driven spectral response* $\mathbf{A}_l$. The resulting response $\tilde{\mathbf{A}}_l$ is then applied to the original spectrum $\hat{\mathbf{Y}}_l = \hat{\mathbf{X}}_l \odot \tilde{\mathbf{A}}_l(k)$ and the output feature is reconstructed through inverse Fourier transform $\mathbf{Y}_l = \mathcal{F}^{-1}(\hat{\mathbf{Y}}_l)$. Thus, $\tilde{\mathbf{A}}_l$ serves as the input-adaptive spectral transformation $R_\phi(k)$ in Eq. (9).

Overall, the unified Symbol-conditioned Adaptive Fourier modeling process can be summarized as

$$\mathbf{X}_l \xrightarrow{\mathcal{F}} \hat{\mathbf{X}}_l \xrightarrow{\Psi_\theta} \mathbf{A}_l \xrightarrow{\odot[1+\mathbf{S}_l(k;\mathbf{X}_l,\mathbf{Z}_s)]} \tilde{\mathbf{A}}_l \xrightarrow{\odot \hat{\mathbf{X}}_l} \hat{\mathbf{Y}}_l \xrightarrow{\mathcal{F}^{-1}} \mathbf{Y}_l. \tag{10}$$

3.3 Training objective

In this subsection, we introduce the training objective of SAFPOT, which consists of three components: prediction loss, alignment loss, and load balancing loss.

Prediction loss. For the prediction task, the loss function is defined as $\mathcal{L}_{\text{task}} = \sum_{t=1}^{T} \|\hat{\boldsymbol{u}}^t - \boldsymbol{u}^t\|_2^2$, where $\hat{\boldsymbol{u}}^t$ and $\boldsymbol{u}^t$ represent the prediction and ground truth, respectively.

Alignment loss. Inspired by symmetric cross-modal contrastive learning (Radford et al., 2021) and supervised multi-positive contrastive learning (Khosla et al., 2020), we introduce an alignment loss to promote consistency between the data and symbolic representations. First, we define the cross-modal similarity as $s_{ij} = \frac{(z_i^d)^\top z_j^s}{\tau}$, where z_i^d and z_j^s are the normalized data and symbolic representations, respectively; τ is a learnable temperature parameter. Different trajectories under the same PDE share the same governing dynamics. Accordingly, same-PDE pairs are treated as positives and cross-PDE pairs as negatives. Let $P(i)$ denote the positive set of sample i. The symmetric bidirectional alignment loss is defined as

$$\mathcal{L}_{\text{align}} = \frac{1}{2}(\mathcal{L}_{d\to s} + \mathcal{L}_{s\to d}) = -\frac{1}{2B}\sum_{i=1}^{B}\left[\log\frac{\sum_{j\in\mathcal{P}(i)}\exp(s_{ij})}{\sum_{j=1}^{B}\exp(s_{ij})} + \log\frac{\sum_{j\in\mathcal{P}(i)}\exp(s_{ji})}{\sum_{j=1}^{B}\exp(s_{ji})}\right], \tag{11}$$

where the first term aligns data to symbols, while the second performs the reverse alignment using the transposed similarity matrix. This alignment provides more informative symbolic conditions for frequency modulation.

Load balancing loss. To encourage uniform routing and balanced utilization, we employ an auxiliary load-balancing loss for the MoE layers. For the l-th layer, the importance of expert i is defined as the accumulated routing weight over the batch, $I_i^l = \sum_{b=1}^{B} w_{i,b}^l$, $\mathcal{L}_{\text{balance}}^l = \text{CV}\left(\{I_i^l\}_{i=1}^{N_r}\right)^2$, where $w_{i,b}^l$ denotes the routing weight assigned to expert i, and CV measures the variation of expert importance. The losses from all MoE layers are summed as $\mathcal{L}_{\text{moe}} = \sum_{l=1}^{L} \mathcal{L}_{\text{balance}}^l$.

The overall training objective is defined as

$$\mathcal{L} = \mathcal{L}_{\text{task}} + \lambda_{\text{align}}\mathcal{L}_{\text{align}} + \lambda_{\text{moe}}\mathcal{L}_{\text{moe}}, \tag{12}$$

where λ_{align} and λ_{moe} weight the alignment and load balancing losses, respectively.

4 Experiments

4.1 Experimental Setup

Datasets. We pre-train the model on 6 datasets from 3 benchmarks: FNO (Li et al., 2021), PDEBench (Takamoto et al., 2022), and CFDBench (Luo et al., 2023). These datasets involve a broad range of PDE types and parameters. The mathematical formulations of these PDEs are listed in Appendix A.3. To ensure consistency and compatibility across datasets, we apply padding and masking during preprocessing. Details are provided in Appendix A.2.

Baselines. We compare SAFPOT with two categories of baselines. (1) Small models include FNO (and Geo-FNO (Li et al., 2022) for datasets on irregular domains), U-Net (Ronneberger et al., 2015), FFNO (Tran et al., 2023), GK-Transformer (Cao, 2021), OFormer (Li et al., 2023), and GNOT (Hao et al., 2023). These models are trained and evaluated separately on each dataset. (2) Pre-trained models include NESTOR (Sun et al., 2026), the parameter-scaled DPOT-(T/S/M) (Hao et al., 2024) and MoE-POT-(T/S/M) (Wang et al., 2025). All pre-trained models use the same 6 datasets and evaluation settings.

Training and Evaluation. The hyperparameter settings are specified in Appendix A.1. All models are trained on four NVIDIA RTX 6000D GPUs, each with 80 GB memory. We evaluate data- and physics-oriented performance using L2RE and fRMSE, respectively. L2RE (Li et al., 2021) measures the relative difference between the predicted field $\hat{\boldsymbol{u}}^t$ and ground-truth field $\boldsymbol{u}^t$,

$$\text{L2RE} = \frac{||\hat{\boldsymbol{u}}^t - \boldsymbol{u}^t||_2}{||\boldsymbol{u}^t||_2}. \tag{13}$$

We quantify frequency-domain prediction error using fRMSE:

$$\text{fRMSE} = \frac{\sum_{k_{\min}}^{k_{\max}} \sqrt{|\mathcal{F}(\hat{\boldsymbol{u}}^t) - \mathcal{F}(\boldsymbol{u}^t)|^2}}{k_{\max} - k_{\min} + 1}, \tag{14}$$

where $\mathcal{F}$ is a discrete Fourier transform, and $k_{\min}$, $k_{\max}$ are the minimum and maximum indices in Fourier coordinates. Following the prior work (Hu et al., 2026), fRMSE is computed by applying a 3D FFT to the prediction and target fields, grouping them by frequency magnitude, and averaging within each group. We further partition the spectrum into low-, middle-, and high-frequency bands for fine-grained evaluation. Lower L2RE and fRMSE values indicate better performance.

4.2 Main Results

Data-oriented evaluation. The L2RE results on the pre-training test sets are detailed in Table 1. We compare various baseline models (first column) across different PDE datasets and parameter configurations (top row). Our analysis of these results is divided into two parts:

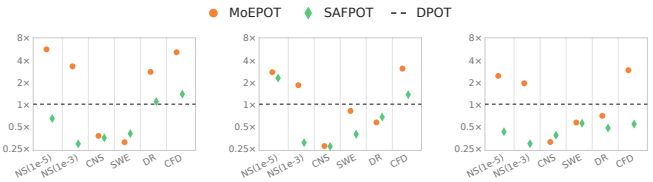
- *Pre-trained models.* SAFPOT achieves the best zero-shot performance on 5 out of 6 datasets, with significant improvements in L2RE compared to NESTOR, DPOT, and MoE-POT. Specifically,

Table 1: L2RE results under three experimental settings: single-dataset models trained independently on each dataset (top), pre-trained models under zero-shot evaluation (middle), and fine-tuned models (bottom). Best results within each setting are **bold**, with overall best results in **blue**.

	Model	Activated Params	FNO–ν		PDEBench			CFDBench	Average
			10^{-5}	10^{-3}	CNS	DR	SWE		
Small models	FNO	0.5M	0.156	0.0128	0.170	0.120	0.00440	**0.00761**	0.0785
	UNet	25M	0.198	0.0245	0.3570	0.0971	0.05210	0.20900	0.1563
	FFNO	1.2M	**0.121**	**0.0099**	0.0452	0.0571	0.01160	0.00714	0.0420
	GK–T	1.1M	0.134	0.0980	0.0366	0.0359	0.00692	0.01050	0.0537
	GNOT	2.2M	0.157	0.0125	0.0341	0.0311	**0.00678**	0.00877	0.0417
	Oformer	1.8M	0.1705	0.0104	**0.0205**	**0.0192**	0.00717	0.01020	**0.0397**
Pre-trained	NESTOR	13M	0.0674	0.00763	0.0159	**0.0184**	0.00449	0.00911	0.0205
	DPOT–T	7.5M	0.0958	0.01215	0.0213	0.0592	0.00451	0.00602	0.0332
	DPOT–S	30M	0.0679	0.00800	0.0235	0.0496	0.00304	0.00514	0.0262
	DPOT–M	122M	0.0604	0.00777	0.0215	0.0374	0.00265	0.00794	0.0229
	MoE-POT–T	17M	0.0737	0.00881	0.0108	0.0513	0.00405	0.00634	0.0258
	MoE-POT–S	90M	0.0633	0.00801	0.0127	0.0474	0.00314	0.00471	0.0232
	MoE-POT–M	288M	0.0584	0.00679	0.0100	0.0369	0.00267	0.00620	0.0202
	Ours–T	28M	0.0514	0.00538	0.0111	0.0309	0.00257	0.00461	0.0177
	Ours–S	140M	0.0461	**0.00510**	0.0101	0.0300	**0.00238**	**0.00404**	0.0163
	Ours–M	295M	**0.0446**	0.00512	**0.0092**	0.0264	0.00246	0.00559	**0.0156**
Fine-tuned	NESTOR	13M	0.0761	0.00779	0.0138	0.0266	0.00366	0.00685	0.0225
	DPOT–T	7.5M	0.0671	0.00680	0.0219	0.0273	0.00394	0.00426	0.0219
	DPOT–S	30M	0.0476	0.00526	0.0231	0.0223	0.00560	0.00346	0.0179
	DPOT–M	122M	0.0417	0.00560	0.0081	0.0190	0.00264	0.00680	0.0140
	MoE-POT–T	17M	0.0494	0.00592	0.0143	0.0247	0.00945	0.00476	0.0181
	MoE-POT–S	90M	0.0429	0.00501	0.0093	0.0226	0.00489	0.00326	0.0147
	MoE-POT–M	288M	0.0398	0.00466	0.0084	0.0184	0.00205	0.00509	0.0131
	Ours–T	28M	0.0419	0.00377	0.0072	0.0186	0.00267	0.00283	0.0128
	Ours–S	140M	0.0337	0.00339	0.0071	0.0185	0.00328	0.00303	0.0115
	Ours–M	295M	**0.0327**	**0.00330**	**0.0068**	**0.0138**	**0.00161**	**0.00279**	**0.0102**

we observe that Ours-M achieves the lowest average L2RE of 0.0156, representing a 22.8% reduction compared with the strongest pre-trained baseline, MoE-POT-M (0.0202). Compared with OFormer, the best-performing small model with an average L2RE of 0.0397, Ours-M reduces the average error by approximately 60.7%. Notably, even without any downstream fine-tuning, Ours-T achieves an average L2RE of 0.0177, slightly outperforming the fine-tuned MoE-POT-T (0.0181) and substantially outperforming the fine-tuned DPOT-T (0.0219). These results demonstrate that SAFPOT not only achieves strong cross-PDE generalization through pre-training, but also exhibits substantial potential for further improvement through downstream fine-tuning.

- *Fine-tuned models.* The results show that after 200 epochs of fine-tuning, our model achieves state-of-the-art performance across all 6 tasks, with Ours-M attaining an average L2RE of 0.0102, corresponding to a 22.1% reduction compared with MoE-POT-M (0.0131). The advantage becomes even more pronounced under scale-matched comparisons. For example, Ours-T reduces the L2RE from 0.00945 for MoE-POT-T to 0.00267, corresponding to a 71.75% reduction on PDEBench-SWE. Compared with training from scratch, fine-tuning on pretrained weights generally leads to better performance. This suggests that cross-PDE pre-training provides a stronger initialization by capturing transferable spatiotemporal and spectral patterns shared across different PDE systems, thereby reducing the burden of learning task-specific dynamics from scratch. The consistent gains after fine-tuning further indicate that these pre-trained representations remain sufficiently adaptable to downstream PDE-specific dynamics.



(a) Tiny (b) Small (c) Medium

Figure 2: Normalized fRMSE across six datasets and three model sizes. The dashed line denotes the size-matched DPOT baseline (=1); lower values indicate better performance.

Physics-oriented evaluation. The fRMSE prediction performance of DPOT, MoE-POT, and SAFPOT in the frequency domain is shown in Fig. 2. SAFPOT achieves the lowest fRMSE on 3/6, 3/6, and 5/6 datasets at the Tiny, Small, and Medium scales, respectively. The frequency-domain advantage becomes increasingly consistent as the model size grows. At the Medium scale, our model reduces the normalized fRMSE to $0.29\times$–$0.55\times$ that of DPOT across all datasets, while MoE-POT achieves the lowest fRMSE only on CNS. MoE-POT, in contrast, exhibits larger cross-dataset variations, particularly on FNO-ν (10^{-5}) and CFDBench. Notably, SAFPOT attains the lowest average fRMSE at all model sizes, reaching 1.54×10^{-3}, 2.04×10^{-3}, and 9.10×10^{-4} for SAFPOT-T, SAFPOT-S, and SAFPOT-M, respectively. Compared with the strongest size-matched baseline at each scale, these results correspond to average fRMSE reductions of approximately 50.5%, 24.7%, and 59.4%. Detailed fRMSE results are provided in Appendix C.2. Overall, these results demonstrate that SAFPOT achieves strong frequency-domain generalization across heterogeneous PDE systems, with its advantage becoming more pronounced at larger scales.

4.3 Band-wise Frequency Analysis

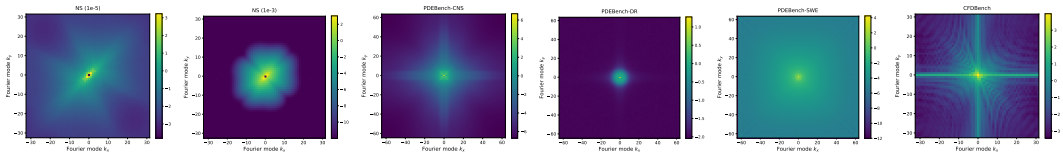


Figure 3: Spectrograms of 6 PDE datasets with distinct spectral distributions.

We analyze the spectrograms of 6 PDEs and compare medium-sized pre-trained operators across low-, mid-, and high-frequency bands using fRMSE. The results are shown in Fig. 3 and Table 2.

Spectral distributions across PDEs. NS (10^{-3}) and DR have compact, low-frequency-dominated spectra. NS (10^{-5}) exhibits a broader and diagonally elongated distribution. CNS and CFDBench contain strong axis-aligned components, whereas SWE combines a sharp low-frequency peak with weak, diffuse components.

Low-, mid-, and high-frequency fRMSE. SAFPOT achieves the lowest fRMSE on four of the six datasets in the low-frequency band and on five of the six datasets in both the mid- and high-frequency bands. It also obtains the lowest average fRMSE in all three bands, with values of 1.09×10^{-3}, 7.76×10^{-4}, and 8.09×10^{-4}, respectively.

In summary, our model achieves robust performance across all three frequency bands and datasets with distinct spectral characteristics. The results highlight the advantages of SAF in capturing heterogeneous frequency components across PDE systems through spectral adaptation guided by PDE-specific symbolic information.

4.4 Ablation Experiments

We conduct ablation studies using the Tiny configuration by removing key components from the full model and report the average L2RE in Table 3.

Impact of frequency modulation. Removing frequency modulation causes the largest performance drop, increasing the average error from 0.0177 to 0.0210. This result highlights the importance of frequency-domain adaptation.

Table 2: Low-, Mid-, and High-frequency fRMSE comparison of different models on 6 PDEs.

	Method	FNO-ν		PDEBench			CFDBench	Average
		10^{-5}	10^{-3}	CNS	DR	SWE		
Low	DPOT	2.48×10^{-3}	8.90×10^{-4}	1.22×10^{-2}	1.08×10^{-3}	7.62×10^{-4}	1.14×10^{-3}	3.08×10^{-3}
	MoE-POT	5.52×10^{-3}	1.73×10^{-3}	$\mathbf{2.81 \times 10^{-3}}$	7.04×10^{-4}	$\mathbf{3.77 \times 10^{-4}}$	2.63×10^{-3}	2.30×10^{-3}
	Ours	$\mathbf{9.12 \times 10^{-4}}$	$\mathbf{2.59 \times 10^{-4}}$	4.00×10^{-3}	$\mathbf{4.51 \times 10^{-4}}$	4.09×10^{-4}	$\mathbf{5.07 \times 10^{-4}}$	$\mathbf{1.09 \times 10^{-3}}$
Mid	DPOT	3.94×10^{-3}	5.27×10^{-4}	4.60×10^{-3}	1.67×10^{-4}	3.34×10^{-4}	1.13×10^{-3}	1.78×10^{-3}
	MoE-POT	9.63×10^{-3}	9.96×10^{-4}	$\mathbf{1.84 \times 10^{-3}}$	1.21×10^{-4}	2.41×10^{-4}	3.20×10^{-3}	2.67×10^{-3}
	Ours	$\mathbf{1.67 \times 10^{-3}}$	$\mathbf{1.51 \times 10^{-4}}$	1.96×10^{-3}	$\mathbf{1.04 \times 10^{-4}}$	$\mathbf{2.12 \times 10^{-4}}$	$\mathbf{5.57 \times 10^{-4}}$	$\mathbf{7.76 \times 10^{-4}}$
High	DPOT	3.99×10^{-3}	1.77×10^{-4}	4.62×10^{-3}	9.94×10^{-5}	1.96×10^{-4}	8.38×10^{-4}	1.65×10^{-3}
	MoE-POT	1.01×10^{-2}	3.52×10^{-4}	$\mathbf{2.10 \times 10^{-3}}$	7.10×10^{-5}	1.37×10^{-4}	3.15×10^{-3}	2.65×10^{-3}
	Ours	$\mathbf{1.81 \times 10^{-3}}$	$\mathbf{5.41 \times 10^{-5}}$	2.23×10^{-3}	$\mathbf{5.82 \times 10^{-5}}$	$\mathbf{1.17 \times 10^{-4}}$	$\mathbf{5.79 \times 10^{-4}}$	$\mathbf{8.09 \times 10^{-4}}$

Impact of symbol branch. Removing the symbol branch results in an increase of 0.0007 in the average L2RE. This degradation indicates that symbolic information is not merely an isolated auxiliary feature, but contributes to the model through its interaction with frequency-domain representations.

Table 3: Ablation study of different components.

Method	FNO-ν		PDEBench			CFDBench	Average
	10^{-5}	10^{-3}	CNS	DR	SWE		
w/o frequency modulation	0.0621	0.00690	0.0116	0.0387	0.00287	0.00407	0.0210
w/o symbol branch	0.0542	0.00550	0.0109	0.0333	0.00266	0.00400	0.0184
w/o alignment loss	0.0534	0.00589	0.0111	0.0322	0.00263	0.00421	0.0182
Ours	0.0514	0.00538	0.0111	0.0309	0.00257	0.00461	0.0177

Impact of the alignment loss. Removing the align loss leads to an increase of 0.0005 in the average L2RE. Although its contribution is smaller compared to other modules, it yields a modest improvement in average L2RE.

Table 4: L2RE results on downstream tasks. Best results are in **bold**.

Dataset	Geo-FNO	U-Net	FFNO	DPOT		MoE-POT		Ours	
				Scratch	Pre-train	Scratch	Pre-train	Scratch	Pre-train
NS (10^{-4})	0.107	0.413	0.220	0.0599	0.0264	0.0480	0.0160	0.0257	**0.0089**
CNS $(1, 0.01)$	0.0813	0.0827	0.390	0.0521	0.0398	0.0381	0.0307	0.0159	**0.0116**
PDEArena	0.154	0.167	0.161	0.111	0.0621	0.137	0.0618	0.0839	**0.0378**

4.5 Downstream Tasks Experiments

To evaluate the transferability and generalization of our pre-trained model, we conduct downstream experiments on three unseen PDEs. We initialize the downstream models with the pre-trained SAFPOT weights. All models are trained from scratch or fine-tuned for 500 epochs. As shown in Table 4, SAFPOT trained from scratch already achieves competitive performance across all three tasks, demonstrating the effectiveness of the proposed architecture for operator learning. The pre-trained SAFPOT also achieves the best performance on all three downstream tasks, including PDEArena with a substantially different equation structure. These results indicate that pre-training enables the model to learn more transferable and effective representations. Moreover, it maintains strong predictive performance even on substantially different downstream tasks, fully showcasing its advantage in capturing PDE characteristics.

5 Conclusion

In this paper, we propose SAFPOT for large-scale PDE pre-training. We define multi-level frequency bands for full-frequency learning and design the SAF module to adapt spectral responses by jointly leveraging physical states and symbolic PDE information. Our model effectively captures heterogeneous spectral characteristics, achieving robust full-frequency prediction and downstream transferability across diverse PDE systems. These results highlight SAFPOT as a scalable foundation for large-scale physical modeling.

AI USE STATEMENT

During the preparation of this manuscript, we used generative AI tools to assist with drafting, language refinement, grammar, and readability. We did not use generative AI tools to generate experimental results or formulate scientific claims. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

REPRODUCIBILITY STATEMENT

We provide detailed model configurations, training hyperparameters, data preprocessing procedures, and dataset settings in the Appendix. Additional implementation details and supplementary experiments are also included to facilitate reproducibility.

REFERENCES

Shuhao Cao. Choose a Transformer: Fourier or Galerkin. In *Advances in Neural Information Processing Systems*, volume 34, pp. 24924–24940. Curran Associates, Inc., 2021.

Damai Dai, Chengqi Deng, Chenggang Zhao, RX Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Yu Wu, et al. DeepSeekMoE: Towards ultimate expert specialization in mixture-of-experts language models. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 1280–1297, 2024.

Wenhan Gao, Ruichen Xu, Yuefan Deng, and Yi Liu. Discretization-invariance? on the discretization mismatch errors in neural operators. In *The Thirteenth International Conference on Learning Representations*, 2025.

John Guibas, Morteza Mardani, Zongyi Li, Andrew Tao, Anima Anandkumar, and Bryan Catanzaro. Adaptive fourier neural operators: Efficient token mixers for transformers. In *International Conference on Learning Representations*, 2022.

Zhongkai Hao, Zhengyi Wang, Hang Su, Chengyang Ying, Yinpeng Dong, Songming Liu, Ze Cheng, Jian Song, and Jun Zhu. GNOT: A general neural operator transformer for operator learning. In *International conference on machine learning*, pp. 12556–12569. PMLR, 2023.

Zhongkai Hao, Chang Su, Songming Liu, Julius Berner, Chengyang Ying, Hang Su, Anima Anandkumar, Jian Song, and Jun Zhu. DPOT: Auto-regressive denoising operator transformer for large-scale PDE pre-training. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 17616–17635. PMLR, 2024.

Peiyan Hu, Haodong Feng, Hongyuan Liu, Tongtong Yan, Wenhao Deng, Tianrun Gao, Rong Zheng, Haoren Zheng, Chenglei Yu, Chuanrui Wang, Kaiwen Li, Zhi-Ming Ma, Dezhi Zhou, Xingcai Lu, Dixia Fan, and Tailin Wu. Realpdebench: A benchmark for complex physical systems with real-world data. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=y3oHMcoItR. Oral Presentation.

Marimuthu Kalimuthu, David Holzmüller, and Mathias Niepert. LOGLO-FNO: Efficient learning of local and global features in fourier neural operators. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 18661–18673, 2020.

Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Learning maps between function spaces with applications to PDEs. *Journal of Machine Learning Research*, 24(89):1–97, 2023.

Zhijie Li, Wenhui Peng, Zelong Yuan, and Jianchun Wang. Fourier neural operator approach to large eddy simulation of three-dimensional turbulence. *Theoretical and Applied Mechanics Letters*, 12 (6):100389, 2022.

Zijie Li, Kazem Meidani, and Amir Barati Farimani. Transformer for partial differential equations' operator learning. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=EPPqt3uERT.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations*, 2021.

Ning Liu, Siavash Jafarzadeh, and Yue Yu. Domain agnostic fourier neural operators. In *Advances in Neural Information Processing Systems*, volume 36, pp. 47438–47450, 2023. doi: 10.52202/075280-2053.

Shengjun Liu, Yu Yu, Ting Zhang, Hanchao Liu, Xinru Liu, and Deyu Meng. Architectures, variants, and performance of neural operators: A comparative review. *Neurocomputing*, 648:130518, 2025.

Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3(3):218–229, 2021.

Yining Luo, Yingfa Chen, and Zhen Zhang. Cfdbench: A large-scale benchmark for machine learning methods in fluid dynamics. *arXiv preprint arXiv:2310.05963*, 2023.

Michael McCabe, Bruno Régaldo-Saint Blancard, Liam Holden Parker, Ruben Ohana, Miles D. Cranmer, Alberto Bietti, Michael Eickenberg, Siavash Golkar, Géraud Krawezik, François Lanusse, Mariel Pettee, Tiberiu Tesileanu, Kyunghyun Cho, and Shirley Ho. Multiple physics pretraining for spatiotemporal surrogate models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 119301–119335, 2024. doi: 10.52202/079017-3791.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PMLR, 2021.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2017.

Junhong Shen, Tanya Marwah, and Ameet Talwalkar. UPS: Efficiently building foundation models for PDE solving via cross-modal adaptation. *Transactions on Machine Learning Research*, 2024.

Pengqing Shi, Jie Yin, Stephen Tierney, and Junbin Gao. Sirenfno: Efficient and full frequency learning of fourier neural operators. In *Proceedings of the 35th International Joint Conference on Artificial Intelligence (IJCAI-ECAI 2026)*, 2026.

Dengdi Sun, Xiaoya Zhou, Xiao Wang, Hao Si, Wanli Lyu, Jin Tang, and Bin Luo. NESTOR: A nested MOE-based neural operator for large-scale PDE pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6147–6156, 2026.

Makoto Takamoto, Timothy Praditia, Raphael Leiteritz, Daniel MacKinlay, Francesco Alesiani, Dirk Pflüger, and Mathias Niepert. PDEBench: An extensive benchmark for scientific machine learning. In *Advances in Neural Information Processing Systems*, volume 35, pp. 1596–1611, 2022.

Alasdair Tran, Alexander Mathews, Lexing Xie, and Cheng Soon Ong. Factorized fourier neural operators. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=tmIiMPl4IPa.

Hong Wang, Haiyang Xin, Jie Wang, Xuanze Yang, Fei Zha, Huanshuo Dong, and Yan Jiang. Mixture-of-experts operator transformer for large-scale PDE pre-training. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-1057.

Zipeng Xiao, Siqi Kou, Zhongkai Hao, Bokai Lin, and Zhijie Deng. Amortized fourier neural operators. In *Advances in Neural Information Processing Systems*, volume 37, pp. 115001–115020, 2024.

Xihang Yue, Yi Yang, and Linchao Zhu. Holistic physics solver: Learning PDEs in a unified spectral-physical space. In *Proceedings of the 42nd International Conference on Machine Learning*, 2025.

APPENDIX

A EXPERIMENTAL DETAILS

A.1 DETAILS OF PRE-TRAINING AND FINE-TUNING

Pre-training. We developed three model variants of different scales, namely SAFPOT-Tiny, SAFPOT-Small, and SAFPOT-Medium (detailed configurations summarized in Table 5). For the pre-training stage, we adopted a learning rate of 1×10^{-3} and used a one-cycle learning rate schedule over 1000 epochs, with the first 200 epochs as the warm-up phase. We optimized the model using Adam with a weight decay of 1×10^{-6} and $\beta_1 = \beta_2 = 0.9$. Training was conducted using four NVIDIA RTX 6000D GPUs, with a total batch size of 20. The dataset weight was uniformly set to $w = 1$ across all datasets. We set the patch size to 8. During training, we used T = 10 time steps to predict the next frame, aligning with the original settings of most datasets.

Fine-tuning. In this stage, we applied a learning rate of 1×10^{-3} with a one-cycle schedule over 200 epochs (the first 40 epochs reserved for warm-up). The key component of our model is the SAF module, which learns adaptive spectral representations conditioned on both the input states and governing PDEs. During fine-tuning, the pre-trained SAF layers are further optimized on each target dataset, allowing the learned spectral representations to adapt to the dataset-specific frequency characteristics while preserving the transferable knowledge acquired during pre-training. The remaining model parameters are jointly updated to facilitate task-specific adaptation.

Table 5: Model configurations for different SAFPOT sizes.

Size	Attention dim	MLP dim	Layers	Heads	Model size	Activated size
Tiny	512	512	4	4	40M	28M
Small	1024	1024	6	8	200M	140M
Medium	1024	2048	8	8	498M	295M

A.2 DATA PREPROCESSING AND SAMPLING

Data Padding and Masking. PDE datasets often differ in the number of physical variables, spatial resolution, and domain geometry. If we directly sample from the raw data, we obtain batches with substantial variation in sample size, resulting in imbalanced computational workloads and lower efficiency in modern multi-GPU training. Following prior work (Hao et al., 2024), we standardize the spatial and channel dimensions of all datasets. First, the spatial resolution is set to $H \times W = 128 \times 128$, which is used by many of the datasets considered in this work. Datasets with lower resolutions are upsampled to H via interpolation, while those with higher resolutions are randomly downsampled or interpolated to H. Second, all datasets are padded along the channel dimension to match the maximum number of variables across datasets, with unused channels filled with a constant value (set to 1 in our experiments). For datasets with irregular geometric shapes, we use an additional mask channel that encodes the geometric configuration of each PDE instance. This ensures consistent representation across datasets while preserving unique structural information.

Table 6 further summarizes the training and test set sizes adopted for pre-training and downstream evaluation.

Table 6: Train and test set sizes of the PDE datasets used for pre-training and downstream evaluation.

Setting	Dataset	Training set size	Test set size
Pre-training	NS(10^{-5})	1,000	200
	NS (10^{-3})	1,000	200
	CNS $(0.1, 0.01)$	9,000	200
	SWE	900	60
	DR	900	60
	CFDBench	9,000	1,000
Downstream	NS (10^{-4})	2,000	200
	CNS $(1, 0.01)$	2,000	200
	PDEArena	2,000	200

Noise Injection. During pre-training, we perturb the input sequences with Gaussian noise to improve optimization stability. Specifically, for $t \leq T$, denote $\boldsymbol{u}^{<t} = (\boldsymbol{u}^0, \ldots, \boldsymbol{u}^{t-1})$ and the noise as $\boldsymbol{\epsilon} \sim \mathcal{N}\left(0,\, \epsilon \left\|\boldsymbol{u}^{<t}\right\| I\right)$. We then feed $\boldsymbol{u}^{<t} + \boldsymbol{\epsilon}$ into the model.

A.3 Detailed Information of Datasets

The PDE datasets used in our experiments are summarized below.

FNO-ν. This dataset describes the temporal evolution of the vorticity field $w(x,t)$ in two-dimensional incompressible fluid, where $(x,t) \in [0,1]^2 \times [0,T]$. The dynamics are governed by the two-dimensional Navier–Stokes equations in the vorticity form:

$$\partial_t w + u \cdot \nabla w = \nu \Delta w + f(x), \qquad \nabla \cdot u = 0,$$

where u denotes the velocity field, Δ is the Laplace operator, ν represents the viscosity coefficient, and $f(x)$ is the external forcing term. We consider three viscosity settings: $\nu \in \{10^{-3}, 10^{-4}, 10^{-5}\}$.

PDEBench-CNS. This dataset describes the temporal evolution of two-dimensional compressible fluid flows over the spatiotemporal domain $(x,t) \in [0,1]^2 \times [0,T]$. Each sample consists of the velocity field $\boldsymbol{u}(x,t)$, pressure field $p(x,t)$, and density field $\rho(x,t)$. The dynamics are governed by the compressible Navier–Stokes equations:

$$\begin{aligned}
\partial_t \rho + \nabla \cdot (\rho \boldsymbol{u}) &= 0, \\
\rho\big(\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u}\big) &= -\nabla p + \eta \Delta \boldsymbol{u} + \big(\zeta + \frac{\eta}{3}\big) \nabla \big(\nabla \cdot \boldsymbol{u}\big), \\
\partial_t \left(\frac{3}{2} p + \frac{\rho u^2}{2}\right) &= -\nabla \cdot \left(\left(\varepsilon + p + \frac{\rho u^2}{2}\right) \boldsymbol{u} - \boldsymbol{u} \cdot \boldsymbol{\sigma}'\right).
\end{aligned}$$

Here, η denotes dynamic shear viscosity and ζ is bulk viscosity.

PDEBench-SWE. This dataset describes the temporal evolution of the water depth field $h(x,t)$ over the spatiotemporal domain $(x,t) \in [-1,1]^2 \times [0,5]$. The dynamics are governed by the two-dimensional shallow-water equations:

$$\begin{aligned}
\partial_t h + \nabla \cdot (h \boldsymbol{u}) &= 0, \\
\partial_t (h \boldsymbol{u}) + \nabla \cdot \left(\frac{1}{2} h \boldsymbol{u}^2 + \frac{1}{2} g_r h^2\right) &= -g_r h \nabla b.
\end{aligned}$$

PDEBench-DR. This dataset describes the temporal evolution of the density fields $\boldsymbol{u}(x,t)$ over the spatiotemporal domain $(x,t) \in [-2.5, 2.5]^2 \times [0,1]$. The dynamics are governed by the two-dimensional diffusion–reaction (DR) systems:

$$\partial_t \boldsymbol{u} = D \nabla^2 \boldsymbol{u} + R(\boldsymbol{u}).$$

PDEArena. This dataset describes the temporal evolution of the velocity field $\boldsymbol{u}(x,t)$, pressure field $p(x,t)$, and density field $\rho(x,t)$ over the spatiotemporal domain $(x,t) \in [0,32]^2 \times [0,24]$. The dynamics are governed by the incompressible Navier-Stokes equation:

$$\partial_t \boldsymbol{u} + (\boldsymbol{u} \cdot \nabla)\boldsymbol{u} = -\nabla p + \nu \Delta \boldsymbol{u}, \qquad \nabla \cdot \boldsymbol{u} = 0.$$

CFDBench. This dataset describes the temporal evolution of incompressible or weakly compressible flows in irregular geometries. We need to predict the velocity field $\boldsymbol{u}(x,t)$ and pressure field $p(x,t)$ over computational domains with complex boundaries. The dynamics are governed by the following equations:

$$\begin{aligned} \partial_t(\rho \boldsymbol{u}) + \nabla \cdot (\rho \boldsymbol{u}^2) &= -\nabla p + \nabla \cdot \mu \left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T \right), \\ \nabla \cdot (\rho \boldsymbol{u}) &= 0. \end{aligned}$$

B Detailed architecture

B.1 Multi-head Structure for the Spectral Mixer

For computational efficiency, the complex-valued spectral mapping Ψ_θ in the SAF module is implemented using a grouped multi-head structure. Rather than applying a single dense complex-valued MLP to the entire channel dimension, we partition the Fourier representation into h groups and process them independently. Given the Fourier representation $\hat{\mathbf{X}}_l$, we divide its channel dimension into h groups:

$$\hat{\mathbf{X}}_l = \mathrm{Concat}\left(\hat{X}_l^1, \hat{X}_l^2, \ldots, \hat{X}_l^h \right), \tag{15}$$

where each group $\hat{X}_l^i$ contains $d = D/h$ channels. Each group is transformed independently by a two-layer complex-valued MLP:

$$A_l^i = W_{2,i}\, \sigma\left(W_{1,i} \hat{X}_l^i + b_{1,i} \right) + b_{2,i}, \qquad i = 1, \ldots, h, \tag{16}$$

where $W_{1,i}$ and $W_{2,i}$ denote the learnable complex-valued mappings for the i-th group, and $b_{1,i}$ and $b_{2,i}$ are the corresponding biases. The resulting group-wise spectral responses are concatenated along the channel dimension:

$$\mathbf{A}_l = \mathrm{Concat}\left(A_l^1, A_l^2, \ldots, A_l^h \right). \tag{17}$$

This grouped formulation allows different channel subspaces to learn distinct spectral transformations while reducing the computational and memory cost of the complex-valued mapping. Since the groups are independent, their transformations can also be computed in parallel.

B.2 Mixture-of-Experts Layer

We use MoE-POT as a primary baseline and adopt its MoE architecture (Wang et al., 2025). Specifically, each MoE layer consists of a set of shared experts and routed experts, together with a router that dynamically selects a subset of routed experts for each input. Both the experts and the routing network are implemented with convolutional layers. Given the input feature $\mathbf{Y}_l$ at the l-th block, the router produces routing logits $R_l(\mathbf{Y}_l(x)) = G_l(\mathbf{Y}_l(x)) \in \mathbb{R}^{N_r}$, where N_r denotes the number of routed experts (we set $N_r = 16$). The corresponding routing probabilities are obtained through

$$w_l(\mathbf{Y}_l(x)) = \mathrm{Softmax}\left(R_l(\mathbf{Y}_l(x)) \right). \tag{18}$$

Only the K experts with the largest routing probabilities are activated, yielding

$$\mathrm{TopK}\left(w_l(\mathbf{Y}_l(x)) \right) = \left\{ (i_k, w_l^{i_k}) \right\}_{k=1}^{K}, \tag{19}$$

where i_k is the index of the selected routed expert and $w_l^{i_k}$ denotes its normalized routing weight.

Let $\mathbf{E}_l^s = \{E_{l(s)}^1, \ldots, E_{l(s)}^{N_s}\}$ and $\mathbf{E}_l^r = \{E_{l(r)}^1, \ldots, E_{l(r)}^{N_r}\}$ denote the shared and routed expert sets, respectively. The shared experts are activated for every input, whereas only the Top-K routed experts are selected by the router. Each shared or routed expert is implemented as a convolutional subnetwork that takes $\mathbf{Y}_l(x)$ as input and produces an output feature map of the same shape. The output of the MoE layer (Dai et al., 2024; Shazeer et al., 2017) is then computed as

$$\mathbf{Y}_{l+1}(x) = \frac{1}{N_s} \sum_{i=1}^{N_s} E_{l(s)}^i(\mathbf{Y}_l(x)) + \sum_{k=1}^{K} w_l^{i_k} E_{l(r)}^{i_k}(\mathbf{Y}_l(x)). \tag{20}$$

C SUPPLEMENTARY EXPERIMENTS

C.1 SCALING EXPERIMENTS

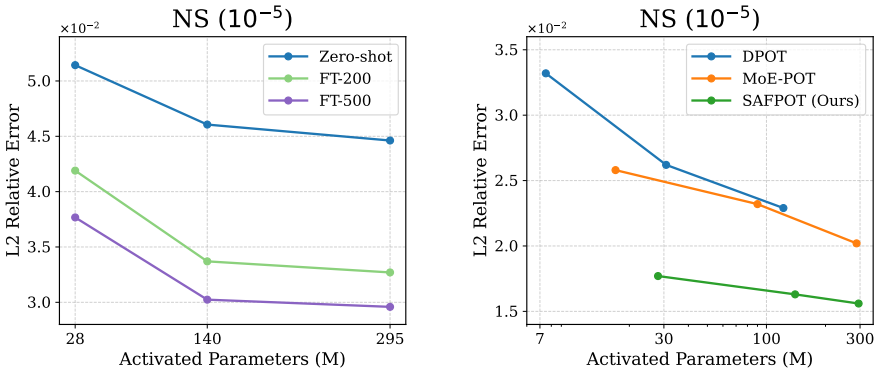


Figure 4: Scaling analysis on NS (10^{-5}). Left: L2 relative errors under different numbers of fine-tuning epochs. Right: zero-shot L2 relative errors across different models.

Model scalability is a key property of large-scale pre-trained models. Here, we conduct scaling experiments on NS (10^{-5}) to evaluate scalability. This dataset represents a low-viscosity regime that exhibits rich multi-scale turbulence and is particularly relevant to our goal of adaptive full-frequency modeling. As shown in Fig. 4, we observe that the zero-shot test error consistently decreases as the model size increases, approximately following a scaling law. From the left figure, the model consistently benefits from increasing model capacity under both zero-shot and fine-tuning settings, while fine-tuning further improves performance across all scales. From the right figure, our model maintains lower zero-shot errors than DPOT and MoE-POT across different model scales. Notably, SAFPOT with only 28M activated parameters achieves a lower zero-shot error than the 288M MoE-POT, highlighting its strong parameter efficiency.

C.2 DETAILED FRMSE RESULTS

Table 7: Physics-oriented prediction errors (fRMSE) of DPOT, MoE-POT, and SAFPOT across six PDE benchmarks. Lower is better. The best result within each model scale is shown in **bold**.

Method	FNO-ν		PDEBench			CFDBench	Average
	10^{-5}	10^{-3}	CNS	DR	SWE		
DPOT-T	4.47×10^{-3}	8.49×10^{-4}	1.11×10^{-2}	$\mathbf{4.24 \times 10^{-4}}$	7.02×10^{-4}	$\mathbf{1.10 \times 10^{-3}}$	3.11×10^{-3}
MoE-POT-T	2.49×10^{-2}	2.78×10^{-3}	4.12×10^{-3}	1.17×10^{-3}	$\mathbf{2.13 \times 10^{-4}}$	5.61×10^{-3}	6.46×10^{-3}
SAFPOT-T	$\mathbf{2.85 \times 10^{-3}}$	$\mathbf{2.45 \times 10^{-4}}$	$\mathbf{3.88 \times 10^{-3}}$	4.60×10^{-4}	2.79×10^{-4}	1.51×10^{-3}	$\mathbf{1.54 \times 10^{-3}}$
DPOT-S	$\mathbf{3.12 \times 10^{-3}}$	5.48×10^{-4}	1.19×10^{-2}	5.88×10^{-4}	3.55×10^{-4}	$\mathbf{9.68 \times 10^{-4}}$	2.92×10^{-3}
MoE-POT-S	8.48×10^{-3}	9.94×10^{-4}	3.22×10^{-3}	$\mathbf{3.32 \times 10^{-4}}$	2.87×10^{-4}	2.95×10^{-3}	2.71×10^{-3}
SAFPOT-S	7.06×10^{-3}	$\mathbf{1.64 \times 10^{-4}}$	$\mathbf{3.18 \times 10^{-3}}$	3.93×10^{-4}	$\mathbf{1.38 \times 10^{-4}}$	1.31×10^{-3}	$\mathbf{2.04 \times 10^{-3}}$
DPOT-M	3.38×10^{-3}	5.31×10^{-4}	7.63×10^{-3}	4.31×10^{-4}	4.48×10^{-4}	1.02×10^{-3}	2.24×10^{-3}
MoE-POT-M	8.18×10^{-3}	1.02×10^{-3}	$\mathbf{2.33 \times 10^{-3}}$	2.99×10^{-4}	2.52×10^{-4}	2.95×10^{-3}	2.51×10^{-3}
SAFPOT-M	$\mathbf{1.42 \times 10^{-3}}$	$\mathbf{1.54 \times 10^{-4}}$	2.88×10^{-3}	$\mathbf{2.04 \times 10^{-4}}$	$\mathbf{2.46 \times 10^{-4}}$	$\mathbf{5.46 \times 10^{-4}}$	$\mathbf{9.10 \times 10^{-4}}$