LEARNING MODULATED SPECTRAL-SPATIAL STATE-SPACE MODEL FOR SOLVING DYNAMIC PDES

Anonymous authors
Paper under double-blind review

ABSTRACT

Forecasting the evolution of systems governed by partial differential equations (PDEs) requires capturing interactions across scales and effectively conditioning on temporal history. We propose S^4-dyn, a spectral-spatial state-space neural operator that unifies cross-frequency modeling with history-conditioned spatial modulation. Its spectral-spatial backbone combines selective state-space modeling along bidirectional radial-angular scans of Fourier modes with local spatial processing, capturing complementary global and local interactions. A parallel modulation branch evolves historical observations in a latent spectral space and uses the resulting context to adaptively guide the forecasting backbone. Tri-plane factorization extends the selective state-space modeling in 2D Fourier space to 3D systems with reduced length of state-space sequence. We further establish an input-adaptive spatial kernel interpretation that characterizes the operator's cross-frequency interactions. Across 15 diverse 2D and 3D physical systems, S^4-dyn demonstrates strong dynamic prediction performance, achieving the lowest error in most evaluations on The Well and on PDEArena-NS, while remaining comparable to the strongest baseline on FNO-NS. Ablation studies support the effectiveness of combining structured spectral modeling with adaptive temporal conditioning for PDE forecasting.

1 INTRODUCTION

Learning a surrogate model for dynamic partial differential equations (PDEs) provides a data-driven alternative to efficiently solving complex dynamical systems. For autoregressive forecasting, the predicted solution is recursively applied for future predictions, and prediction errors may accumulate over long time steps rollouts. This is particularly challenging in dynamics with coexisting of large and small scale vortices, where cross-scale coupling affects future evolution of PDE solutions. Modeling these inter-scale interactions is thus essential for autoregressive forecasting.

Existing neural PDE solvers fundamentally differ in the choice of representation space or mechanisms to model nonlocal interactions. Fourier neural operators perform global spatial mixing in spectral space (Li et al., 2021), while attention-based architectures such as Transolver, CViT, and scOT learn nonlocal interactions over physical or latent representations (Wu et al., 2024; Wang et al., 2025; Herde et al., 2024). Other architectures introduce different interaction structures. For example, Flower method (Muser et al., 2026) constructs spatial feature interactions by warping domain coordinates through state-dependent spatial transformations. State-space models (SSMs) offer another route to long-range interaction (Gu et al., 2022; Gu & Dao, 2023), they have recently been introduced into neural operator learning through architectures such as LaMO (Tiwari et al., 2025) that applies SSMs over multidirectional raster scans of latent tokens, or AMO (Song & Jiang, 2026) that models frequency-domain transfer functions via adaptive Takenaka–Malmquist rational systems. While existing neural operators attempt to capture nonlocal spatial or scale dependencies in PDE solutions, they face key limitations in modeling cross-scale spectral coupling and effective temporal conditioning. Specifically, standard Fourier neural operators rely on local mode transformations, with limited explicit inter-mode interaction in spectral space. Existing spatial or spectral SSMs (e.g., LaMO, AMO) do not fully consider the sequence ordering of frequency radial-angular cascades of PDE solutions. Moreover, historical context lacks sophisticated design to modulate the prediction of future solutions, affecting the performance of longer prediction of PDE solutions.

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.

To address these challenges in cross-scale spectral coupling and temporal context modeling, we propose S^4**-dyn** (**S**pectral-**S**patial **S**tate-**S**pace Operator for **dyn**amic PDEs), a novel neural operator learning framework that unifies Fourier-domain state-space modeling with dynamic spatial feature modulation. Specifically, S^4-dyn constructs a multi-scale spectral sequence by partitioning Fourier modes into discrete radial rings from low to high frequencies via a novel Bidirectional Radial-Angular (BDRA) scan. This enables a selective state-space model to explicitly model cross-scale dependencies along a low-to-high radial ordering while modeling angular interactions along bidirectional paths. Based on this, multiple spectral-spatial state-space blocks are cascaded to build the main branch of S^4-dyn. Furthermore, to relieve context prediction drift during autoregressive predictions, a parallel Spatial Feature Modulation branch encodes historical solutions via spectral transform, and injects layer-wise Spatial-FiLM scale and shift fields into the main branch to adaptively modulate the multiple blocks in main branch. For PDEs with 3D domains, the spectral state-space model is extended by a Tri-Plane Factorized Spectral Module that projects the 3D spectral volumes onto three orthogonal 2D planes for BDRA processing, bypassing dense volumetric state-space sequence construction, which is both memory and computation intensive. We evaluate S^4-dyn on 15 diverse 2D and 3D physical systems, including 13 systems from The Well (Ohana et al., 2024) and two additional Navier–Stokes datasets (Li et al., 2021; Gupta & Brandstetter, 2023). On The Well, S^4-dyn achieves the lowest VRMSE on 10 of 13 systems for next-step prediction and 9 of 13 systems for both the 1:20 and 21:60 rollout steps. On the two additional Navier–Stokes datasets, S^4-dyn achieves the lowest relative L_2 error on PDEArena-NS and remains comparable to the strongest baseline on FNO-NS.

2 Related Work

Neural operators have demonstrated significant success in solving PDEs through data-driven approaches (Kovachki et al., 2023). DeepONet (Lu et al., 2021) employs a branch network that encodes the input function and a trunk network that encodes the query coordinates, whose outputs are combined to evaluate the solution. FNO (Li et al., 2021) performs global operator learning through spectral convolution. F-FNO (Tran et al., 2023) enhances model efficiency by employing factorization in the Fourier domain. U-NO (Rahman et al., 2023) and U-FNO (Wen et al., 2022) combine FNO and U-Net for multiscale problems. Transformer-based neural operators instead use attention to model nonlocal dependencies. However, standard Transformer architectures suffer from quadratic computational complexity, limiting their scalability to large-scale meshes. Models including OFormer (Li et al., 2023), GNOT (Hao et al., 2023), and ONO (Xiao et al., 2024) alleviate quadratic complexity by redesigning the attention mechanism itself. To avoid applying attention directly over massive mesh points, methods such as Transolver (Wu et al., 2024), UPT (Alkin et al., 2024), and GAOT (Wen et al., 2025) perform compact attention in the latent feature space.

Recent work improves temporal PDE forecasting through memory, physics-based consistency, or latent propagation. MemNO (Buitrago Ruiz et al., 2025) models the non-Markovian dynamics induced by partial observations by augmenting a Fourier neural operator with an S4 approximation of the Mori–Zwanzig memory term. PITA (Zhu et al., 2025) reduces rollout errors by aligning the governing equations inferred from autoregressive predictions with those discovered from observed trajectories. SSP (Lu et al., 2026) advances the retained low-frequency modes of a compact latent state to predict the next temporal window. Our method instead uses the evolved history as a conditioning signal that generates spatial modulation fields for a separate forecasting backbone.

Beyond attention-based architectures, state-space models have recently emerged as an alternative backbone for neural operators. SSM-based PDE solvers mainly define sequences in physical or latent space. LaMO (Tiwari et al., 2025) applies SSMs over multidirectional raster scans of latent tokens, while MNO (Cheng et al., 2026) processes physical-domain patches along bidirectional scan paths using S6 and Cross-S6 blocks. AMO (Song & Jiang, 2026) models frequency-domain transfer functions via adaptive Takenaka–Malmquist rational systems. FourierMamba (Li et al., 2025) is designed for image deraining using progressive and bilateral zigzag scans over amplitude and phase spectra of images. Unlike this image-centric approach, our approach addresses dynamic PDE prediction and explicitly organizes Fourier modes from coarse to fine spectral scales by partitioning them into increasing radial rings with bidirectional polar angle paths.

These studies highlight two complementary aspects of PDE forecasting: global feature mixing and temporal conditioning. Our S^4-dyn addresses them by organizing Fourier modes into radial rings from low to high frequencies and applying SSMs using the proposed bidirectional angular scans. In parallel, it evolves the observed history in a compact latent space and generates layer-wise spatial scale and shift fields to condition the forecasting backbone.

3 METHOD

We consider a dynamical partial differential equation defined over a spatial domain $\Omega \subset \mathbb{R}^d$:

$$\partial_t \mathbf{u} = \mathcal{N}[\mathbf{u}] \ (\mathbf{x} \in \Omega), \quad \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}) \ (\mathbf{x} \in \Omega), \quad \mathcal{B}[\mathbf{u}] = 0 \ (\mathbf{x} \in \partial\Omega). \tag{1}$$

Where $\mathbf{u}(\mathbf{x}, t) \in \mathbb{R}^{d_u}$ is the physical state, $\mathcal{N}$ is the governing differential operator, and $\mathcal{B}$ encodes the boundary conditions. We consider both 2D and 3D spatial domains. Let $\mathbf{u}_t = \mathbf{u}(\cdot, t)$ be the observed PDE solution at discrete time step t. We define the T_{in} history frames as $\mathbf{U}_t = (\mathbf{u}_{t-T_{\text{in}}+1}, \ldots, \mathbf{u}_t)$. The forecasting task is to learn a next timestep prediction operator $\mathcal{G}_\Theta$, parameterized by the trainable S^4-dyn model, such that

$$\widehat{\mathbf{u}}_{t+1} = \mathcal{G}_\Theta(\mathbf{U}_t). \tag{2}$$

For multi-step forecasting, the same operator is applied autoregressively: $\widehat{\mathbf{u}}_{t+s} = \mathcal{G}_\Theta(\widehat{\mathbf{U}}_{t+s-1}), s = 1, \ldots, T_{\text{ar}}$, where $\widehat{\mathbf{U}}_t$ is the current observed/estimated frames till time step t. After each prediction step, the newly predicted frame is appended while retaining the fixed length of history frames as T_{in}.

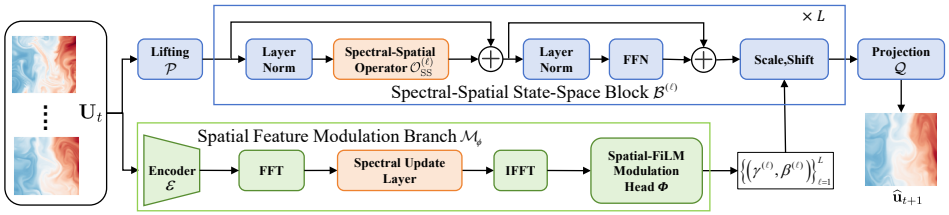

Figure 1: S^4-dyn architecture. Given the history $\mathbf{U}_t$, the Spectral-Spatial State-Space Branch predicts the next state through L stacked S^4-Blocks $\{\mathcal{B}^{(\ell)}\}_{\ell=1}^{L}$. In parallel, the Spatial Feature Modulation Branch $\mathcal{M}_\phi$ generates layer-wise modulation parameters $\{(\gamma^{(\ell)}, \beta^{(\ell)})\}_{\ell=1}^{L}$ to modulate the corresponding S^4-Blocks.

3.1 ARCHITECTURE OVERVIEW

We now introduce the forecasting operator $\mathcal{G}_\Theta$, implemented as S^4-dyn comprising two coupled branches, as shown in Fig. 1. The *Spectral-Spatial State-Space Branch* predicts the next frame from the observed history, while the *Spatial Feature Modulation Branch* encodes the history frames and generates layer-wise spatial feature modulation.

Let $\mathcal{P}$ and $\mathcal{Q}$ be the lifting and projection layers, with channel mappings $T_{\text{in}} d_u \to C$ and $C \to d_u$, where C is the hidden feature dimension. Let $\mathcal{B}^{(\ell)}$ denote the ℓ-th Spectral-Spatial State-Space Block (S^4-Block). Given $\mathbf{U}_t$,

$$\mathbf{h}^{(0)} = \mathcal{P}(\mathbf{U}_t), \quad \mathbf{h}^{(\ell)} = \mathcal{B}^{(\ell)}\big(\mathbf{h}^{(\ell-1)}; \gamma^{(\ell)}, \beta^{(\ell)}\big), \quad \widehat{\mathbf{u}}_{t+1} = \mathcal{Q}(\mathbf{h}^{(L)}). \tag{3}$$

Where the layer-wise modulation fields are generated from the same history by $\mathcal{M}_\phi$ as $\{(\gamma^{(\ell)}, \beta^{(\ell)})\}_{\ell=1}^{L} = \mathcal{M}_\phi(\mathbf{U}_t)$. Each S^4-Block follows a pre-normalized residual design, with the Spectral-Spatial Operator $\mathcal{O}_{\text{SS}}^{(\ell)}$ as its main feature-mixing module. Specifically, the ℓ-th S^4-Block is defined as

$$\begin{aligned} \mathbf{x}^{(\ell)} &= \mathrm{LN}_1^{(\ell)}\left(\mathbf{h}^{(\ell-1)}\right), \mathbf{a}^{(\ell)} = \mathcal{O}_{\text{SS}}^{(\ell)}\left(\mathbf{x}^{(\ell)}\right) + \mathbf{h}^{(\ell-1)}, \\ \widetilde{\mathbf{h}}^{(\ell)} &= \mathbf{a}^{(\ell)} + \mathrm{MLP}^{(\ell)}\left(\mathrm{LN}_2^{(\ell)}\left(\mathbf{a}^{(\ell)}\right)\right), \mathbf{h}^{(\ell)} = \left(1 + \gamma^{(\ell)}\right) \odot \widetilde{\mathbf{h}}^{(\ell)} + \beta^{(\ell)}, \end{aligned} \tag{4}$$

where $\mathrm{LN}_1^{(\ell)}$ and $\mathrm{LN}_2^{(\ell)}$ denote LayerNorm, $\mathrm{MLP}^{(\ell)}$ is the feed-forward network, and $\odot$ denotes element-wise multiplication. Spectral-Spatial Operator and Spatial Feature Modulation Branch are detailed in Sec. 3.2 and 3.4.

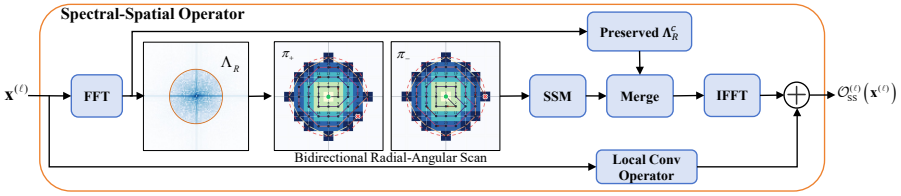

Figure 2: The Bidirectional Spectral State-Space Operator updates the retained Fourier modes in Λ_R through the Bidirectional Radial-Angular (BDRA) Scan and state-space modeling, while the complementary modes Λ_R^c are preserved. In parallel, the Local Convolutional Operator captures short-range spatial interactions. They are combined to produce $\mathcal{O}_{\mathrm{SS}}^{(\ell)}(\mathbf{x}^{(\ell)})$.

3.2 Spectral-Spatial Operator

In the ℓ-th S^4-Block, $\mathcal{O}_{\mathrm{SS}}^{(\ell)}$ takes $\mathbf{x}^{(\ell)}$ in Eq. 4 as input and combines the Bidirectional Spectral State-Space Operator $G_{\Lambda_d}^{(\ell)}$ with the Local Convolutional Operator $C^{(\ell)}$:

$$\mathcal{O}_{\mathrm{SS}}^{(\ell)}\left(\mathbf{x}^{(\ell)}\right) = G_{\Lambda_d}^{(\ell)}\left(\mathbf{x}^{(\ell)}\right) + C^{(\ell)}\left(\mathbf{x}^{(\ell)}\right), \tag{5}$$

where Λ_d denotes the retained spectral region, with $\Lambda_d = \Lambda_R$ for $d = 2$ and $\Lambda_d = \Lambda_{cub}$ for $d = 3$. We first describe $G_{\Lambda_R}^{(\ell)}$ for 2D fields; its extension to 3D fields via Tri-Plane Factorized Spectral Modeling is introduced in Sec. 3.3.

As illustrated in Figure 2, let $\mathcal{F}$ and $\mathcal{F}^{-1}$ denote the Fourier transform and its inverse, respectively. For the 2D operator $G_{\Lambda_R}^{(\ell)}$, we compute $\widehat{\mathbf{x}}^{(\ell)} = \mathcal{F}\left(\mathbf{x}^{(\ell)}\right) \in \mathbb{C}^{B\times C\times H\times W}$. Then we retain the spectral region $\Lambda_R = \{\mathbf{k} = (k_x, k_y) : \|\mathbf{k}\|_2 \le R\}$ and denote the corresponding Fourier coefficients by $\widehat{\mathbf{x}}_R^{(\ell)}$. We represent these complex coefficients in Cartesian form by concatenating their real and imaginary components along the feature dimension:

$$\widetilde{\mathbf{x}}_R^{(\ell)} = \left[\mathrm{Re}\left(\widehat{\mathbf{x}}_R^{(\ell)}\right), \mathrm{Im}\left(\widehat{\mathbf{x}}_R^{(\ell)}\right)\right]. \tag{6}$$

Bidirectional Radial-Angular (BDRA) Scan. Dynamic PDE fields often exhibit structures at multiple spatial scales, with lower spatial frequencies corresponding to larger-scale variations and higher frequencies to finer-scale variations. To provide a structured ordering of Fourier modes for state-space modeling, we group the retained two-dimensional modes into radial rings and arrange the rings in order of increasing radius. Within each ring, we traverse the modes in two opposite angular directions, yielding two sequences with a shared low-to-high radial progression. This radial-angular serialization provides the State-Space Model (SSM) with a structured sequence for modeling dependencies across spectral scales and complementary angular contexts. Specifically, we discretize the frequency magnitude $r_i = \|\mathbf{k}_i\|_2$ into concentric radial rings indexed by $q_i = \lfloor r_i + \frac{1}{2} \rfloor \in \{0, 1, \ldots, Q\}$, where Q denotes the outermost retained ring. For the q-th ring containing n_q modes, we construct an angle-sorted index sequence $\mathcal{I}_q^+ = (i_{q,1}, i_{q,2}, \ldots, i_{q,n_q})$ such that $\theta_{i_{q,1}} \le \theta_{i_{q,2}} \le \cdots \le \theta_{i_{q,n_q}}$, where $\theta_i = \mathrm{atan2}(k_{y,i}, k_{x,i})$. The reverse angular sequence is $\mathcal{I}_q^- = (i_{q,n_q}, \ldots, i_{q,2}, i_{q,1})$. When transitioning from ring q to ring $q+1$, the scan continues from i_{q,n_q} to $i_{q+1,1}$ for π_+, and from $i_{q,1}$ to $i_{q+1,n_{q+1}}$ for π_-. The two scan orders π_+ and π_- are formed by concatenating the ring sequences from the innermost ($q = 0$) to the outermost ($q = Q$) scale:

$$\pi_+ = \mathcal{I}_0^+ \| \mathcal{I}_1^+ \| \cdots \| \mathcal{I}_Q^+, \qquad \pi_- = \mathcal{I}_0^- \| \mathcal{I}_1^- \| \cdots \| \mathcal{I}_Q^-. \tag{7}$$

This yields the serialized spectral sequences $\widetilde{\mathbf{x}}_{R,\pm}^{(\ell)} = \widetilde{\mathbf{x}}_R^{(\ell)}[\pi_\pm]$. Both scan orders share the same low-to-high radial progression ($q = 0 \to Q$), while reversing the angular traversal within each radial scale to provide complementary directional contexts.

Bidirectional Spectral State-Space Operator. The two serialized sequences are modeled separately by parameterized State-Space Operators implemented with Mamba (Gu & Dao, 2023):

$$\mathbf{O}_{R,\pm}^{(\ell)} = \mathrm{SSM}_{\pm}^{(\ell)}\left(\widetilde{\mathbf{x}}_{R,\pm}^{(\ell)}\right). \tag{8}$$

The two outputs are then restored to their original Fourier locations according to π_+ and π_-, and their paired real and imaginary components are recombined into complex coefficients. The two directional outputs are summed to obtain the updated coefficients $\widehat{\mathbf{S}}_R^{(\ell)}$ on Λ_R. The updated coefficients replace the input spectrum within Λ_R, while the complementary coefficients remain unchanged. The complete $G_{\Lambda_R}^{(\ell)}$ is therefore

$$G_{\Lambda_R}^{(\ell)}\left(\mathbf{x}^{(\ell)}\right) = \mathcal{F}^{-1}\left[\begin{cases}\widehat{\mathbf{S}}_R^{(\ell)}(\mathbf{k}), & \mathbf{k}\in\Lambda_R,\\ \widehat{\mathbf{x}}^{(\ell)}(\mathbf{k}), & \mathbf{k}\notin\Lambda_R,\end{cases}\right] \tag{9}$$

Local Convolutional Operator. In parallel, $C^{(\ell)}$ performs local spatial processing directly on $\mathbf{x}^{(\ell)}$:

$$C^{(\ell)}\left(\mathbf{x}^{(\ell)}\right) = \mathrm{Conv}_2^{(\ell)}\left(\mathrm{GELU}\left(\mathrm{Conv}_1^{(\ell)}\left(\mathbf{x}^{(\ell)}\right)\right)\right). \tag{10}$$

The outputs of $G_{\Lambda_R}^{(\ell)}$ and $C^{(\ell)}$ are combined according to Eq. 5.

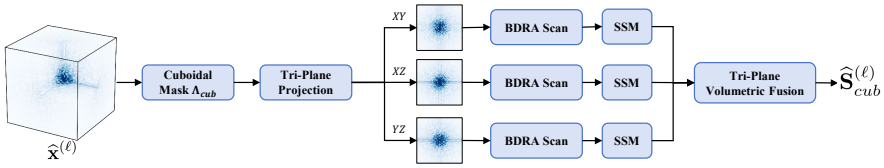

Figure 3: 3D Tri-plane factorized spectral modeling. The spectral cuboid is projected to form XY, XZ, and YZ planar representations, each of which is serialized by BDRA and processed by State-Space Operators. Then the plane-wise features are combined by Tri-Plane Volumetric Fusion.

3.3 Tri-Plane Factorized Spectral Modeling for 3D Fields

Figure 3 illustrates the extension of $G_{\Lambda_d}^{(\ell)}$ to volumetric fields. Inspired by tri-plane representations in 3D vision (Chan et al., 2022), we form three orthogonal spectral planes for plane-wise state-space modeling. For $d = 3$, we compute the full Fourier spectrum $\widehat{\mathbf{x}}^{(\ell)} = \mathcal{F}(\mathbf{x}^{(\ell)}) \in \mathbb{C}^{B\times C\times H\times W\times D}$ and retain the cuboidal spectral region $\Lambda_{cub} = \{\mathbf{k} = (k_x, k_y, k_z) : |k_x| \le R_x,\ |k_y| \le R_y,\ |k_z| \le R_z\}$. Let $m_x = 2R_x + 1$, $m_y = 2R_y + 1$, and $m_z = 2R_z + 1$. We denote the retained coefficients by $\widehat{\mathbf{x}}_{cub}^{(\ell)} = \widehat{\mathbf{x}}^{(\ell)}\big|_{\Lambda_{cub}}$. Following Eq. 6, we represent them as

$$\widetilde{\mathbf{x}}_{cub}^{(\ell)} = \left[\mathrm{Re}\left(\widehat{\mathbf{x}}_{cub}^{(\ell)}\right), \mathrm{Im}\left(\widehat{\mathbf{x}}_{cub}^{(\ell)}\right)\right] \in \mathbb{R}^{B\times 2C\times m_x\times m_y\times m_z}. \tag{11}$$

Tri-Plane Projection. We project the retained 3D spectral representation onto the XY, XZ, and YZ planes by collapsing the corresponding orthogonal spectral axis:

$$\widetilde{\mathbf{x}}_{xy}^{(\ell)} = \mathcal{P}_{xy}^{(\ell)}\left(\widetilde{\mathbf{x}}_{cub}^{(\ell)}\right), \qquad \widetilde{\mathbf{x}}_{xz}^{(\ell)} = \mathcal{P}_{xz}^{(\ell)}\left(\widetilde{\mathbf{x}}_{cub}^{(\ell)}\right), \qquad \widetilde{\mathbf{x}}_{yz}^{(\ell)} = \mathcal{P}_{yz}^{(\ell)}\left(\widetilde{\mathbf{x}}_{cub}^{(\ell)}\right). \tag{12}$$

Specifically, $\mathcal{P}_{xy}^{(\ell)}$ applies a shared linear map $\mathbb{R}^{Cm_z}\to\mathbb{R}^C$ to the real and imaginary cross-slices at each (k_x, k_y) and concatenates the outputs. The XZ and YZ projections are defined analogously by collapsing the k_y and k_x dimensions, respectively. This gives $\widetilde{\mathbf{x}}_{xy}^{(\ell)} \in \mathbb{R}^{B\times 2C\times m_x\times m_y}$, $\widetilde{\mathbf{x}}_{xz}^{(\ell)} \in \mathbb{R}^{B\times 2C\times m_x\times m_z}$, and $\widetilde{\mathbf{x}}_{yz}^{(\ell)} \in \mathbb{R}^{B\times 2C\times m_y\times m_z}$. BDRA is then applied independently to each plane using its corresponding two-dimensional frequency coordinates.

Plane-Wise State-Space Modeling. Each projected plane preserves a two-dimensional frequency layout, allowing us to reuse the BDRA ordering in Sec. 3.2 without constructing a volumetric scan. For $p \in \{xy, xz, yz\}$, the two BDRA sequences are processed by independently parameterized State-Space Operators:

$$\mathbf{O}_{p,\pm}^{(\ell)} = \mathrm{SSM}_{p,\pm}^{(\ell)}\left(\widetilde{\mathbf{x}}_p^{(\ell)}[\pi_{p,\pm}]\right). \tag{13}$$

The outputs are restored to their planar Fourier locations, recombined into complex coefficients, and summed over the two directions to obtain $\widehat{\mathbf{S}}_p^{(\ell)}, p \in \{xy, xz, yz\}$.

Tri-Plane Volumetric Fusion. The three updated planar spectra are broadcast along their collapsed axes and summed to form the update of the retained 3D spectrum:

$$\widehat{\mathbf{S}}^{(\ell)}_{cub}(k_x,k_y,k_z)=\widehat{\mathbf{S}}^{(\ell)}_{xy}(k_x,k_y)+\widehat{\mathbf{S}}^{(\ell)}_{xz}(k_x,k_z)+\widehat{\mathbf{S}}^{(\ell)}_{yz}(k_y,k_z). \tag{14}$$

As in the 2D formulation, these coefficients replace the input spectrum within Λ_{cub}, while the complementary coefficients are preserved. The resulting 3D spectral operator is

$$G^{(\ell)}_{\Lambda_{cub}}\left(\mathbf{x}^{(\ell)}\right)=\mathcal{F}^{-1}\left[\begin{cases}\widehat{\mathbf{S}}^{(\ell)}_{cub}(\mathbf{k}), & \mathbf{k}\in\Lambda_{cub},\\ \widehat{\mathbf{x}}^{(\ell)}(\mathbf{k}), & \mathbf{k}\notin\Lambda_{cub}.\end{cases}\right]. \tag{15}$$

For a retained cuboid of size $m_x\times m_y\times m_z$, the tri-plane factorization reduces the sequence length per scan direction from $m_xm_ym_z$ to $m_xm_y+m_xm_z+m_ym_z$.

3.4 Spatial Feature Modulation Branch

The Spatial Feature Modulation Branch $\mathcal{M}_\phi$ generates layer-wise spatial modulation from a predicted next-step latent state. Specifically, it first predicts $\mathbf{z}_{t+1}$ from a compact spectral representation of the encoded history, and the Spatial-FiLM Modulation Head then maps $\mathbf{z}_{t+1}$ to the spatial scale and shift fields used in Eq. 4:

$$\mathcal{M}_\phi(\mathbf{U}_t)=\left\{\left(\gamma^{(\ell)},\beta^{(\ell)}\right)\right\}_{\ell=1}^{L}=\Phi(\mathbf{z}_{t+1}). \tag{16}$$

Where, Φ denotes the Spatial-FiLM Modulation Head. For 2D fields, $\gamma^{(\ell)}$ and $\beta^{(\ell)}$ are defined over $\mathbb{R}^{B\times C\times H\times W}$; for 3D fields, they are defined over $\mathbb{R}^{B\times C\times H\times W\times D}$.

To predict $\mathbf{z}_{t+1}$, a shared frame-wise encoder maps each observed state to $\mathbf{z}_\tau=E(\mathbf{u}_\tau)$, $\tau=t-T_{\text{in}}+1,\dots,t$. The encoded state has size $B\times C_z\times H_z\times W_z$ in 2D and $B\times C_z\times H_z\times W_z\times D_z$ in 3D. We apply $\mathcal{F}$ to each latent state and retain spectral region

$$\Lambda_z=\begin{cases}\left\{\mathbf{k}=(k_x,k_y):|k_x|\le R^z_x,\ |k_y|\le R^z_y\right\}, & d=2,\\ \left\{\mathbf{k}=(k_x,k_y,k_z):|k_x|\le R^z_x,\ |k_y|\le R^z_y,\ |k_z|\le R^z_z\right\}, & d=3.\end{cases} \tag{17}$$

The retained spectra are stacked in their temporal order:

$$\widehat{\mathbf{z}}_\tau=\mathcal{F}(\mathbf{z}_\tau),\qquad \widehat{\mathbf{z}}_{\tau,\Lambda_z}=\widehat{\mathbf{z}}_\tau|_{\Lambda_z},\qquad \widehat{\mathbf{Z}}_{t,\Lambda_z}=[\widehat{\mathbf{z}}_{t-T_{\text{in}}+1,\Lambda_z},\dots,\widehat{\mathbf{z}}_{t,\Lambda_z}]. \tag{18}$$

Unlike the radial support Λ_R used by BDRA, Λ_z maintains a regular Cartesian grid for the convolutional latent update.

The Spectral Update Layer predicts the next retained latent spectrum by combining a channel-wise temporal projection with a nonlinear spectral closure:

$$\widehat{\mathbf{z}}_{t+1,\Lambda_z}(\mathbf{k})=\sum_{j=0}^{T_{\text{in}}-1}\mathbf{w}_j\odot\widehat{\mathbf{z}}_{t-T_{\text{in}}+1+j,\Lambda_z}(\mathbf{k})+\mathcal{C}_\psi\left(\widehat{\mathbf{Z}}_{t,\Lambda_z}\right)(\mathbf{k}),\qquad \mathbf{k}\in\Lambda_z. \tag{19}$$

Where $\mathbf{w}_j$ denotes the learned channel-wise temporal weights shared across spectral locations, and $\mathcal{C}_\psi$ is implemented with local convolutions under the real–imaginary representation introduced in Sec. 3.2. The predicted coefficients update only the retained region Λ_z, while the complementary spectrum is preserved from the latest latent state:

$$\widehat{\mathbf{z}}_{t+1}(\mathbf{k})=\begin{cases}\widehat{\mathbf{z}}_{t+1,\Lambda_z}(\mathbf{k}), & \mathbf{k}\in\Lambda_z,\\ \widehat{\mathbf{z}}_t(\mathbf{k}), & \mathbf{k}\notin\Lambda_z,\end{cases}\qquad \mathbf{z}_{t+1}=\mathcal{F}^{-1}\left(\widehat{\mathbf{z}}_{t+1}\right), \tag{20}$$

The predicted latent state $\mathbf{z}_{t+1}$ is mapped by Φ in Eq. 16 to the layer-wise spatial modulation fields. For 2D fields, Φ upsamples $\mathbf{z}_{t+1}$ to $H\times W$ and directly predicts the scale and shift fields. For 3D fields, Φ follows the same tri-plane projection structure as Sec. 3.3, with independently parameterized projections applied to the spatial latent state $\mathbf{z}_{t+1}$:

$$\left\{\left(\gamma^{(\ell)}_p,\beta^{(\ell)}_p\right)\right\}_{\ell=1}^{L}=\text{Up}_p\left[\mathcal{P}^\Phi_p\left(\mathbf{z}_{t+1}\right)\right],\qquad p\in\{xy,xz,yz\}, \tag{21}$$

where $\mathcal{P}_p^{\Phi}$ projects the latent state onto plane p by collapsing the orthogonal spatial axis, and Up_p resizes the planar features to $H \times W$, $H \times D$, and $W \times D$ for $p = xy, xz, yz$, respectively. The three planar fields are then broadcast along their collapsed axes and combined as

$$\begin{aligned}\gamma^{(\ell)}(x,y,z) &= \gamma_{xy}^{(\ell)}(x,y) + \gamma_{xz}^{(\ell)}(x,z) + \gamma_{yz}^{(\ell)}(y,z),\\ \beta^{(\ell)}(x,y,z) &= \beta_{xy}^{(\ell)}(x,y) + \beta_{xz}^{(\ell)}(x,z) + \beta_{yz}^{(\ell)}(y,z).\end{aligned} \tag{22}$$

Thus, the 3D modulation is explicitly parameterized over $HW + HD + WD$ planar locations per channel and block, rather than over the full HWD volume.

Algorithms 1 and 2 in Appendix A summarize the computation of S^4-dyn. Algorithm 1 presents the Spatial Feature Modulation Branch and the Spectral-Spatial State-Space Branch for 2D fields. Algorithm 2 gives the tri-plane operations used by the Spectral-Spatial Operator and the Spatial Feature Modulation Branch for 3D fields.

3.5 Theoretical Analysis

We provide an operator-level interpretation of the spectral state-space modeling in Secs. 3.2 and 3.3.

Theorem 3.1 (Spatial Kernel Representation of the Bidirectional Spectral State-Space Operator)**.** *Let $\mathbf{x}^{(\ell)} \in L^2(\mathbb{T}^d;\mathbb{R}^C)$, where $d \in \{2,3\}$. For each fixed input $\mathbf{x}^{(\ell)}$, the spatial action of $G_{\Lambda_d}^{(\ell)}$ admits the representation*

$$\left[G_{\Lambda_d}^{(\ell)}\left(\mathbf{x}^{(\ell)}\right)\right](\boldsymbol{\xi}) = \int_{\mathbb{T}^d} \mathcal{K}_{G,d}^{(\ell)}\left(\boldsymbol{\xi},\boldsymbol{\eta};\widehat{\mathbf{x}}^{(\ell)}\right)\mathbf{x}^{(\ell)}(\boldsymbol{\eta})\,d\boldsymbol{\eta}, \tag{23}$$

where the input-adaptive kernel decomposes as

$$\mathcal{K}_{G,d}^{(\ell)} = K_{\mathrm{SSM},d}^{(\ell)} + k_{\mathrm{out},d}\mathbf{I}_C. \tag{24}$$

$\mathbf{I}_C$ denotes the $C \times C$ identity matrix. The component induced by the State-Space Operators is

$$K_{\mathrm{SSM},d}^{(\ell)}\left(\boldsymbol{\xi},\boldsymbol{\eta};\widehat{\mathbf{x}}^{(\ell)}\right) = \frac{1}{(2\pi)^d}\sum_{\mathbf{k}\in\Lambda_d}\sum_{\mathbf{m}\in\Lambda_d}\mathbf{T}_d^{(\ell)}\left(\mathbf{k},\mathbf{m};\widehat{\mathbf{x}}^{(\ell)}\right)e^{i(\mathbf{k}\cdot\boldsymbol{\xi}-\mathbf{m}\cdot\boldsymbol{\eta})}, \tag{25}$$

$$k_{\mathrm{out},d}(\boldsymbol{\xi}-\boldsymbol{\eta}) = \delta_{\mathbb{T}^d}(\boldsymbol{\xi}-\boldsymbol{\eta}) - \frac{1}{(2\pi)^d}\sum_{\mathbf{k}\in\Lambda_d}e^{i\mathbf{k}\cdot(\boldsymbol{\xi}-\boldsymbol{\eta})}. \tag{26}$$

The input-dependent transform $\mathbf{T}_d^{(\ell)}(\mathbf{k},\mathbf{m};\widehat{\mathbf{x}}^{(\ell)})$ is characterized in Eq. 43. The term $k_{\mathrm{out},d}\mathbf{I}_C$ represents the identity action on the complementary Fourier modes preserved by $G_{\Lambda_d}^{(\ell)}$.

Theorem 3.1 connects the sequence-based spectral implementation to its cross-frequency operator action in physical space. BDRA structures cross-frequency interactions through a low-to-high radial progression with bidirectional angular traversals. The cross-frequency interactions in Fourier space induce an input-adaptive kernel in the physical domain. The proof is in Appendix B.

4 Experiments

We evaluate S^4-dyn on 15 diverse 2D and 3D physical systems, including 13 systems from The Well (Ohana et al., 2024) and two additional 2D Navier–Stokes datasets, FNO-NS (Li et al., 2021) and PDEArena-NS (Gupta & Brandstetter, 2023). These datasets include acoustic-wave, reaction–diffusion, shallow-water, and Navier–Stokes equations. Dataset descriptions are in Appendix C.

Training and evaluation. We train the one-step operator in Eq. 2 with relative L_2 loss and evaluate it autoregressively. Following Flower (Muser et al., 2026), we use four input frames for The Well and report VRMSE for next-step prediction and rollout steps 1:20 and 21:60, with the latter truncated at the final available target frame when necessary. FNO-NS and PDEArena-NS use ten input frames and report relative L_2 error over 10-step and 4-step rollouts, respectively. Checkpoint selection and further training details are provided in Appendix D.

Baselines. On The Well, we compare S^4-dyn with FNO (Li et al., 2021), CNU-Net (Ohana et al., 2024), scOT (Herde et al., 2024), and Flower (Muser et al., 2026). Their results are taken from

the published comparison in Flower. For FNO-NS and PDEArena-NS, we follow the comparison settings of AMO (Song & Jiang, 2026) and CViT (Wang et al., 2025), respectively. Flower is additionally trained by us under the corresponding settings.

Main Results. Table 1 reports the results on The Well (Ohana et al., 2024). S^4-dyn achieves the lowest VRMSE in 28 of the 39 dataset–horizon comparisons, including 10 of 13 next-step predictions and 9 of 13 cases in each rollout interval. Its advantage is particularly pronounced over steps 21:60, where it reduces the error of the second-best method by 75.4% on Acoustic Maze, 58.0% on Gray–Scott, 49.5% on 3D Rayleigh–Taylor, and 73.1% on TGC. Flower (Muser et al., 2026) remains stronger on Planet SWE and Rayleigh–Bénard. These cases are less naturally aligned with the Cartesian radial organization used by BDRA: Planet SWE is defined on a spherical grid, while Rayleigh–Bénard is periodic only horizontally and has a distinguished vertical direction. We further evaluate the model on two Navier–Stokes datasets in Appendix E. As reported in Tables 5 and 6, S^4-dyn obtains a relative L_2 error of 0.0276 on FNO-NS (Li et al., 2021), comparable to the reported AMO (Song & Jiang, 2026) result of 0.0278, and achieves 0.0085 on PDEArena-NS (Gupta & Brandstetter, 2023), compared with 0.0103 for Flower.

Table 1: VRMSE for $4 \rightarrow 1$ next-step prediction, 1:20 rollout, and 21:60 rollout. Baseline results are adopted from Flower (Muser et al., 2026). Best results are shown in **bold** and second-best results are underlined. En-dash (–) indicates that scOT does not support 3D data.

	Next-step					1:20 Rollout					21:60 Rollout				
Dataset	FNO	CNU-Net	scOT	Flower	S^4-dyn	FNO	CNU-Net	scOT	Flower	S^4-dyn	FNO	CNU-Net	scOT	Flower	S^4-dyn
2D Datasets															
Acoustic Maze	0.1454	0.0129	0.0361	0.0064	**0.0014**	0.4197	0.0874	0.1996	0.0489	**0.0169**	0.7058	0.2199	2.0202	0.1347	**0.0332**
Active Matter	0.1749	0.0650	0.1050	0.0249	**0.0126**	3.2862	1.7781	4.1055	1.3905	**0.5466**	1.8214	1.6465	2.0372	1.5251	**1.3731**
Gray-Scott	0.0372	0.0188	0.0673	**0.0102**	0.0219	0.9125	0.3059	0.4169	0.2074	**0.1271**	37.044	5.1671	4.5485	4.1788	**1.7534**
Planet SWE	0.0070	0.0027	0.0041	**0.0007**	0.0009	0.1316	0.0624	0.0518	**0.0187**	0.0401	0.5368	0.4846	0.2504	**0.1235**	0.2168
Rayleigh-Bénard	0.2104	0.2171	0.1863	**0.0807**	0.1225	39.038	12.507	5.6486	**2.1661**	2.5068	32.562	11.271	11.417	**3.7461**	6.6692
Shear Flow	0.0769	0.0594	0.1093	0.0463	**0.0218**	1.1245	0.7632	0.8930	0.2246	**0.1544**	14.821	7.4620	5.9953	1.2926	**0.8984**
TRL 2D	0.3250	0.2559	0.3555	0.1930	**0.1342**	1.2328	0.7051	0.8299	0.5491	**0.4115**	6.0891	1.3390	1.7280	1.0516	**0.9263**
3D Datasets															
MHD 64	0.3403	0.2062	–	0.1165	**0.0834**	1.3007	0.9534	–	0.7580	**0.7425**	1.6366	1.4299	–	1.2827	**1.2160**
Neutron Star Merger	0.4452	0.3391	–	0.3269	**0.3259**	**0.5980**	0.6529	–	0.6223	0.6109	**0.6209**	0.7938	–	0.7202	0.6617
Rayleigh-Taylor	0.1714	0.1351	–	0.0491	**0.0283**	3.0057	5.3894	–	0.5862	**0.0901**	3.3618	7.4448	–	1.4880	**0.7511**
Supernova Explosion	0.4326	0.4316	–	0.2888	**0.2592**	1.0162	2.2913	–	0.8113	**0.8038**	1.6350	13.615	–	2.0709	**1.6109**
TGC	0.2720	0.2113	–	0.1700	**0.0777**	3.2583	2.0510	–	1.2636	**0.4758**	3.8511	5.9142	–	3.7477	**1.0062**
TRL 3D	0.3261	0.3322	–	0.2073	**0.2051**	0.9203	0.7139	–	**0.6840**	0.9108	1.9969	**0.7885**	–	0.9881	1.2435

4.1 Ablation Study and Model Analysis

Component Ablations and Latent Spectrum Update. Table 2(a) reports ablations on Active Matter (256^2), first examining Spatial Feature Modulation Branch (SFM), $C^{(\ell)}$, $G^{(\ell)}_{\Lambda_R}$. Removing SFM increases the 1:20 and 21:60 rollout VRMSE from 0.5466 and 1.3731 to 0.8040 and 1.7271, showing the benefit of history-conditioned spatial modulation. Without $C^{(\ell)}$, the 1:20 rollout VRMSE increases to 1.5698, whereas removing $G^{(\ell)}_{\Lambda_R}$ increases the next-step and 21:60 VRMSE to 0.0333 and 1.9075. Thus, $C^{(\ell)}$ has a larger effect over the first 20 rollout steps, while $G^{(\ell)}_{\Lambda_R}$ is more important for next-step prediction and the 21:60 interval. Updating all latent Fourier modes in SFM yields nearly unchanged VRMSE, indicating that updating only Λ_z while preserving its complement is sufficient in this setting.

BDRA Scan Design. Table 2(a) then compares different scan paths over the same retained Fourier modes. Removing π_- increases the 1:20 rollout VRMSE from 0.5466 to 1.2004, while adding two high-to-low radial paths increases the parameter count from 6.59M to 8.14M but yields higher VRMSE at all three horizons. BDRA also yields lower VRMSE than the two- and four-direction raster scans of LaMO (Tiwari et al., 2025) and the bilateral and progressive zigzag scans of FourierMamba (Li et al., 2025). These results support a shared low-to-high radial progression with bidirectional angular traversals within each radial ring.

Spectral Operator Choice. The final block of Table 2(a) compares Mamba-based spectral modeling with FNO (Li et al., 2021), self-attention (Vaswani et al., 2017), and linear attention (Katharopoulos et al., 2020), with all variants using 6.59M parameters. The 1:20 rollout VRMSE increases from 0.5466 to 1.2507, 0.6954, and 0.6243. Among the tested spectral operators, the Mamba-based design gives the lowest errors across all three evaluation horizons.

Table 2: Ablation study of S^4-dyn. Lower VRMSE is better.

(a) Active Matter (256^2)

Variant	Params	1-Step ↓	1:20 ↓	21:60 ↓
S^4-dyn	6.59M	0.0126	0.5466	1.3731
w/o SFM	6.04M	0.0159	0.8040	1.7271
w/o $C^{(\ell)}$	3.04M	0.0298	1.5698	1.5724
w/o $G^{(\ell)}_{\Lambda_R}$	4.64M	0.0333	0.6603	1.9075
Full-spectrum update	6.59M	0.0128	0.5408	1.3809
w/o π_-	5.81M	0.0152	1.2004	1.6495
w/ radial reversal	8.14M	0.0187	0.6438	1.4554
2-dir raster scan	6.59M	0.0137	0.6847	1.5061
4-dir raster scan	8.14M	0.0166	0.6008	1.4599
FourierMamba zigzag	8.14M	0.0198	0.9678	1.4431
Mamba → FNO	6.59M	0.0130	1.2507	1.4978
Mamba → Attention	6.59M	0.0141	0.6954	1.4089
Mamba → Linear Attn	6.59M	0.0139	0.6243	1.3971

(b) 3D Rayleigh–Taylor (128^3)

Variant	Params	1-Step ↓	1:20 ↓	21:60 ↓
S^4-dyn	6.98M	0.0283	0.0901	0.7511
w/o SFM	6.18M	0.0298	0.0921	0.7686
Global FiLM	6.89M	0.0315	0.1828	1.2893

(c) Active Matter: R and L

Setting	Params	1-Step ↓	1:20 ↓	21:60 ↓
Retained radius (R)				
$R = 8$	6.59M	0.0208	1.0542	1.5467
$R = 16$	6.59M	0.0195	1.0407	1.4705
$R = 24$	6.59M	0.0126	0.5466	1.3731
$R = 32$	6.59M	0.0143	0.8641	1.4442
Number of S^4-blocks (L)				
$L = 4$	4.56M	0.0149	0.8722	1.5687
$L = 6$	6.59M	0.0126	0.5466	1.3731
$L = 8$	8.61M	0.0290	0.9822	1.4759

Spatial Modulation in 3D. Table 2(b) evaluates SFM on the 3D Rayleigh–Taylor dataset. Removing SFM only slightly increases VRMSE across all three evaluation horizons. We further compare with Global FiLM, where $\gamma^{(\ell)}$ and $\beta^{(\ell)}$ are channel-wise vectors shared across all spatial locations. Global FiLM increases the 1:20 rollout VRMSE from 0.0901 to 0.1828 and the 21:60 rollout VRMSE from 0.7511 to 1.2893, and performs worse than removing SFM.

Hyperparameter Sensitivity. Table 2(c) shows the effects of the retained radius R defining Λ_R and the number L of S^4-blocks on the Active Matter (256^2) dataset. The 1:20 VRMSE decreases from 1.0542 at $R = 8$ to 0.5466 at $R = 24$, but increases to 0.8641 at $R = 32$. $L = 6$ produces 1:20 VRMSE of 0.5466, compared with 0.8722 for $L = 4$, and 0.9822 for $L = 8$. The same ordering is observed for next-step and 21:60 prediction, with $R = 24$ and $L = 6$ giving the lowest errors among the tested settings.

Computational Efficiency. Table 7 in Appendix F compares parameter count, next-timestep inference time, and peak GPU memory on Active Matter (256^2) and Rayleigh–Taylor (128^3). With 6.59M parameters in 2D and 6.98M in 3D, S^4-dyn is the smallest model in both settings. The current implementation is slower and has higher peak GPU memory usage than FNO, CNU-Net, and Flower, but remains faster than scOT on Active Matter.

Qualitative Analysis. Figure 4 shows predictions at rollout step 20 on 3D Rayleigh–Taylor. The error is computed as $E(\mathbf{u}, \widehat{\mathbf{u}}) = |\mathbf{u} - \widehat{\mathbf{u}}|$. S^4-dyn most faithfully preserves the bubble-and-spike morphology, with errors mainly confined to interface regions, whereas FNO, CNU-Net, and Flower exhibit larger deviations. Additional results and the learned γ and β fields are provided in Appendix G.

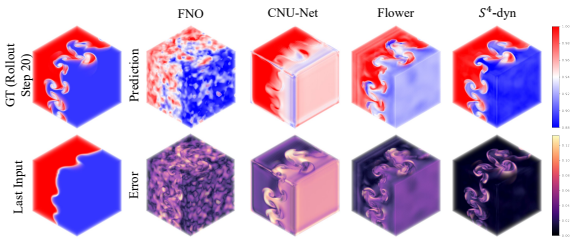

Figure 4: Rayleigh–Taylor density at rollout step 20.

5 Conclusion

We proposed S^4-dyn for dynamic PDE forecasting, combining the Spectral-Spatial State-Space Branch, Spatial Feature Modulation Branch, and Tri-Plane Factorized Spectral Modeling. Across 15 datasets, S^4-dyn achieves the lowest VRMSE on 10 of 13 The Well systems for next-step prediction and 9 of 13 for both rollout intervals, the lowest reported error on PDEArena-NS, and comparable performance to AMO on FNO-NS. Ablations validate the BDRA Scan, the spectral-spatial operators, and Spatial Feature Modulation. Future work will target irregular domains, broader PDE classes, and more efficient spectral-spatial implementations.

AI USE STATEMENT

In this work, we used generative AI tools to assist with language editing, including improving the clarity, grammar, and presentation of author-written text. Additionally, we used these tools to assist with the proof of Theorem 3.1. Generative AI was not used for other tasks requiring disclosure. All AI-assisted mathematical arguments and derivations were independently checked and verified by the authors for correctness, and all AI-assisted text was reviewed and revised by the authors. The authors made the final decisions regarding the scientific claims, mathematical arguments, and presentation, and take full responsibility for the content of this work.

REFERENCES

Benedikt Alkin, Andreas Fürst, Simon Schmid, Lukas Gruber, Markus Holzleitner, and Johannes Brandstetter. Universal physics transformers: A framework for efficiently scaling neural operators. In *Advances in Neural Information Processing Systems*, 2024.

Ricardo Buitrago Ruiz, Tanya Marwah, Albert Gu, and Andrej Risteski. On the benefits of memory for modeling time-dependent pdes. In *International Conference on Learning Representations*, 2025.

Shuhao Cao. Choose a transformer: Fourier or galerkin. In *Advances in Neural Information Processing Systems*, 2021.

Eric R. Chan, Connor Z. Lin, Matthew A. Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J. Guibas, Jonathan Tremblay, Sameh Khamis, Tero Karras, and Gordon Wetzstein. Efficient geometry-aware 3D generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Chun-Wun Cheng, Jiahao Huang, Yi Zhang, Guang Yang, Carola-Bibiane Schönlieb, and Angelica I. Aviles-Rivero. Mamba neural operator: Who wins? transformers vs. state-space models for pdes. *Journal of Computational Physics*, 548:114567, 2026.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations*, 2022.

Gaurav Gupta, Xiongye Xiao, and Paul Bogdan. Multiwavelet-based operator learning for differential equations. In *Advances in Neural Information Processing Systems*, 2021.

Jayesh K Gupta and Johannes Brandstetter. Towards multi-spatiotemporal-scale generalized pde modeling. *Transactions on Machine Learning Research*, 2023.

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*, 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 *International Conference on Machine Learning*, 2024.

Maximilian Herde, Bogdan Raonić, Tobias Rohner, Roger Käppeli, Roberto Molinaro, Emmanuel de Bézenac, and Siddhartha Mishra. Poseidon: Efficient foundation models for pdes. In *Advances in Neural Information Processing Systems*, 2024.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International Conference on Machine Learning*, 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.

Dong Li, Yidi Liu, Xueyang Fu, Jie Huang, Senyan Xu, Qi Zhu, and Zheng-Jun Zha. Fouriermamba: Fourier learning integration with state space models for image deraining. In *International Conference on Machine Learning*, 2025.

Zijie Li, Kazem Meidani, and Amir Barati Farimani. Transformer for partial differential equations' operator learning. *Transactions on Machine Learning Research*, 2023.

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.

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.

Xiaoxiao Lu, Ye Yuan, and Jiahao Shi. Stable long-horizon pde forecasting via latent structured spectral propagators. *arXiv preprint arXiv:2605.10154*, 2026.

Huakun Luo, Haixu Wu, Hang Zhou, Lanxiang Xing, Yichen Di, Jianmin Wang, and Mingsheng Long. Transolver++: An accurate neural solver for pdes on million-scale geometries. In *International Conference on Machine Learning*, 2025.

Till Muser, Alexandra Spitzer, Matti Lassas, Maarten V. de Hoop, and Ivan Dokmanić. Flowers: A warp drive for neural pde solvers. In *International Conference on Machine Learning*, 2026.

Ruben Ohana, Michael McCabe, Lucas Meyer, Rudy Morel, Fruzsina J Agocs, Miguel Beneitez, Marsha Berger, Blakesley Burkhart, Stuart B. Dalziel, Drummond B. Fielding, et al. The well: a large-scale collection of diverse physics simulations for machine learning. In *Advances in Neural Information Processing Systems*, 2024.

Md Ashiqur Rahman, Zachary E. Ross, and Kamyar Azizzadenesheli. U-no: U-shaped neural operators. *Transactions on Machine Learning Research*, 2023.

Zeyuan Song and Zheyu Jiang. Adaptive mamba neural operators. In *International Conference on Learning Representations*, 2026.

Karn Tiwari, Niladri Dutta, N M Anoop Krishnan, and Prathosh A P. Latent mamba operator for partial differential equations. In *International Conference on Machine Learning*, 2025.

Alasdair Tran, Alexander Mathews, Lexing Xie, and Cheng Soon Ong. Factorized fourier neural operators. In *International Conference on Learning Representations*, 2023.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

Sifan Wang, Jacob H. Seidman, Shyam Sankaran, Hanwen Wang, George J. Pappas, and Paris Perdikaris. Cvit: Continuous vision transformer for operator learning. In *International Conference on Learning Representations*, 2025.

Tian Wang and Chuang Wang. Latent neural operator for solving forward and inverse pde problems. In *Advances in Neural Information Processing Systems*, 2024.

Gege Wen, Zongyi Li, Kamyar Azizzadenesheli, Anima Anandkumar, and Sally M Benson. U-fno—an enhanced fourier neural operator-based deep-learning model for multiphase flow. *Advances in Water Resources*, 163:104180, 2022.

Shizheng Wen, Arsh Kumbhat, Levi Lingsch, Sepehr Mousavi, Yizhou Zhao, Praveen Chandrashekar, and Siddhartha Mishra. Geometry aware operator transformer as an efficient and accurate neural surrogate for pdes on arbitrary domains. In *Advances in Neural Information Processing Systems*, 2025.

Haixu Wu, Tengge Hu, Huakun Luo, Jianmin Wang, and Mingsheng Long. Solving high-dimensional pdes with latent spectral models. In *International Conference on Machine Learning*, 2023.

Haixu Wu, Huakun Luo, Haowen Wang, Jianmin Wang, and Mingsheng Long. Transolver: A fast transformer solver for pdes on general geometries. In *International Conference on Machine Learning*, 2024.

Zipeng Xiao, Zhongkai Hao, Bokai Lin, Zhijie Deng, and Hang Su. Improved operator learning by orthogonal attention. In *International Conference on Machine Learning*, 2024.

Congcong Zhu, Xiaoyan Xu, Jiayue Han, and Jingrun Chen. Physics-informed temporal alignment for auto-regressive pde foundation models. In *International Conference on Machine Learning*, 2025.

A ALGORITHMIC SUMMARY

Algorithms 1 and 2 summarize the forward computation of S^4-dyn. Algorithm 1 presents the complete 2D procedure, including history-conditioned spatial modulation and the layer-wise Spectral-Spatial State-Space updates. Algorithm 2 gives the tri-plane extension for 3D fields.

Algorithm 1 Computation of S^4-dyn for 2D fields

Require: History $\mathbf{U}_t = (\mathbf{u}_{t-T_{\text{in}}+1}, \dots, \mathbf{u}_t)$ and number of S^4-Blocks L
Ensure: Next-state prediction $\widehat{\mathbf{u}}_{t+1}$

Spatial Feature Modulation Branch $\mathcal{M}_\phi$
for $\tau = t - T_{\text{in}} + 1, \dots, t$ **do**
  $\mathbf{z}_\tau \leftarrow E(\mathbf{u}_\tau)$
  $\widehat{\mathbf{z}}_\tau \leftarrow \mathcal{F}(\mathbf{z}_\tau)$
  $\widehat{\mathbf{z}}_{\tau,\Lambda_z} \leftarrow \widehat{\mathbf{z}}_\tau\big|_{\Lambda_z}$
end for
$\widehat{\mathbf{Z}}_{t,\Lambda_z} \leftarrow [\widehat{\mathbf{z}}_{t-T_{\text{in}}+1,\Lambda_z}, \dots, \widehat{\mathbf{z}}_{t,\Lambda_z}]$ ▷ Eq. 18
Compute $\widehat{\mathbf{z}}_{t+1,\Lambda_z}$ with the Spectral Update Layer ▷ Eq. 19
Reconstruct $\mathbf{z}_{t+1}$ by updating Λ_z, preserving its complement, and applying $\mathcal{F}^{-1}$ ▷ Eq. 20
$\{(\gamma^{(\ell)}, \beta^{(\ell)})\}_{\ell=1}^{L} \leftarrow \Phi(\mathbf{z}_{t+1})$ ▷ Eq. 16

Spectral-Spatial State-Space Branch
$\mathbf{h}^{(0)} \leftarrow \mathcal{P}(\mathbf{U}_t)$
Construct the BDRA orders π_+ and π_- on Λ_R ▷ Eq. 7
for $\ell = 1, \dots, L$ **do**
  $\mathbf{x}^{(\ell)} \leftarrow \mathrm{LN}_1^{(\ell)}(\mathbf{h}^{(\ell-1)})$
  $\widehat{\mathbf{x}}^{(\ell)} \leftarrow \mathcal{F}(\mathbf{x}^{(\ell)})$
  $\widehat{\mathbf{x}}_R^{(\ell)} \leftarrow \widehat{\mathbf{x}}^{(\ell)}\big|_{\Lambda_R}$
  $\widetilde{\mathbf{x}}_R^{(\ell)} \leftarrow [\mathrm{Re}(\widehat{\mathbf{x}}_R^{(\ell)}), \mathrm{Im}(\widehat{\mathbf{x}}_R^{(\ell)})]$
  for $s \in \{+, -\}$ **do**
    $\mathbf{O}_{R,s}^{(\ell)} \leftarrow \mathrm{SSM}_s^{(\ell)}(\widetilde{\mathbf{x}}_R^{(\ell)}[\pi_s])$ ▷ Eq. 8
  end for
  Restore the two outputs to their Fourier locations, recombine their real–imaginary components, and sum them as $\widehat{\mathbf{S}}_{\mathrm{R}}^{(\ell)}$
  Construct $G_{\Lambda_R}^{(\ell)}(\mathbf{x}^{(\ell)})$ by updating Λ_R, preserving its complement, and applying $\mathcal{F}^{-1}$ ▷ Eq. 9
  $\mathbf{c}^{(\ell)} \leftarrow \mathrm{Conv}_2^{(\ell)}(\mathrm{GELU}(\mathrm{Conv}_1^{(\ell)}(\mathbf{x}^{(\ell)})))$ ▷ Eq. 10
  $\mathbf{a}^{(\ell)} \leftarrow G_{\Lambda_R}^{(\ell)}(\mathbf{x}^{(\ell)}) + \mathbf{c}^{(\ell)} + \mathbf{h}^{(\ell-1)}$
  $\widetilde{\mathbf{h}}^{(\ell)} \leftarrow \mathbf{a}^{(\ell)} + \mathrm{MLP}^{(\ell)}(\mathrm{LN}_2^{(\ell)}(\mathbf{a}^{(\ell)}))$
  $\mathbf{h}^{(\ell)} \leftarrow (1 + \gamma^{(\ell)}) \odot \widetilde{\mathbf{h}}^{(\ell)} + \beta^{(\ell)}$ ▷ Eq. 4
end for
$\widehat{\mathbf{u}}_{t+1} \leftarrow \mathcal{Q}(\mathbf{h}^{(L)})$ ▷ Eq. 3
return $\widehat{\mathbf{u}}_{t+1}$

Algorithm 2 Tri-plane extension of S^4-dyn for 3D fields

Require: Predicted latent state $\mathbf{z}_{t+1}$ and normalized features $\{\mathbf{x}^{(\ell)}\}_{\ell=1}^{L}$
Ensure: $\{G^{(\ell)}_{\Lambda_{cub}}(\mathbf{x}^{(\ell)}), \gamma^{(\ell)}, \beta^{(\ell)}\}_{\ell=1}^{L}$

Spatial Feature Modulation Branch $\mathcal{M}_\phi$
for $p \in \{xy, xz, yz\}$ **do**
  $\{(\gamma_p^{(\ell)}, \beta_p^{(\ell)})\}_{\ell=1}^{L} \leftarrow \mathrm{Up}_p[\mathcal{P}_p^{\Phi}(\mathbf{z}_{t+1})]$ ▷ Eq. 21
end for
for $\ell = 1, \ldots, L$ **do**
  $\gamma^{(\ell)} \leftarrow \gamma_{xy}^{(\ell)} + \gamma_{xz}^{(\ell)} + \gamma_{yz}^{(\ell)}$
  $\beta^{(\ell)} \leftarrow \beta_{xy}^{(\ell)} + \beta_{xz}^{(\ell)} + \beta_{yz}^{(\ell)}$ ▷ Eq. 22
end for

Tri-Plane Factorized Spectral Modeling
for $\ell = 1, \ldots, L$ **do**
  $\widehat{\mathbf{x}}^{(\ell)} \leftarrow \mathcal{F}(\mathbf{x}^{(\ell)})$
  $\widehat{\mathbf{x}}_{cub}^{(\ell)} \leftarrow \widehat{\mathbf{x}}^{(\ell)}\big|_{\Lambda_{cub}}$
  $\widetilde{\mathbf{x}}_{cub}^{(\ell)} \leftarrow [\mathrm{Re}(\widehat{\mathbf{x}}_{cub}^{(\ell)}), \mathrm{Im}(\widehat{\mathbf{x}}_{cub}^{(\ell)})]$
  for $p \in \{xy, xz, yz\}$ **do**
    $\widetilde{\mathbf{x}}_{p}^{(\ell)} \leftarrow \mathcal{P}_p^{(\ell)}(\widetilde{\mathbf{x}}_{cub}^{(\ell)})$ ▷ Eq. 12
    for $s \in \{+, -\}$ **do**
      $\mathbf{O}_{p,s}^{(\ell)} \leftarrow \mathrm{SSM}_{p,s}^{(\ell)}(\widetilde{\mathbf{x}}_{p}^{(\ell)}[\pi_{p,s}])$ ▷ Eq. 13
    end for
    Restore the two outputs to their planar Fourier locations, recombine their real–imaginary components, and sum them as $\widehat{\mathbf{S}}_p^{(\ell)}$
  end for
  $\widehat{\mathbf{S}}_{cub}^{(\ell)}(k_x, k_y, k_z) \leftarrow \widehat{\mathbf{S}}_{xy}^{(\ell)}(k_x, k_y) + \widehat{\mathbf{S}}_{xz}^{(\ell)}(k_x, k_z) + \widehat{\mathbf{S}}_{yz}^{(\ell)}(k_y, k_z)$ ▷ Eq. 14
  Construct $G^{(\ell)}_{\Lambda_{cub}}(\mathbf{x}^{(\ell)})$ by updating Λ_{cub}, preserving its complement, and applying $\mathcal{F}^{-1}$ ▷ Eq. 15
  Continue the remaining S^4-Block operations as in Algorithm 1, using $G^{(\ell)}_{\Lambda_{cub}}(\mathbf{x}^{(\ell)})$ in place of $G^{(\ell)}_{\Lambda_R}(\mathbf{x}^{(\ell)})$
end for

B Kernel Analysis of the Bidirectional Spectral State-Space Operator

This section derives a spatial-kernel representation of the Bidirectional Spectral State-Space Operator $G^{(\ell)}_{\Lambda_d}$ at the level of its selective state-space recurrence. Mamba implements this recurrence in the network; its input/output projections, depthwise convolution, activation, and gate are outside the analysis. The Local Convolutional Operator $C^{(\ell)}$ remains the parallel branch in Eq. 5. We use a periodic continuum representation whose finite-grid realization corresponds to the FFT.

Theorem B.1 (Spatial Kernel Representation of the Bidirectional Spectral State-Space Operator)**.** *Let $\mathbb{T}^d = [0, 2\pi]^d$, where $d \in \{2, 3\}$, and let $\mathbf{x}^{(\ell)} \in L^2(\mathbb{T}^d; \mathbb{R}^C)$ be the input feature field of $G^{(\ell)}_{\Lambda_d}$. Its Fourier coefficients are*

$$\widehat{\mathbf{x}}^{(\ell)}(\mathbf{k}) = \frac{1}{(2\pi)^d} \int_{\mathbb{T}^d} \mathbf{x}^{(\ell)}(\boldsymbol{\eta}) e^{-i\mathbf{k}\cdot\boldsymbol{\eta}}\, d\boldsymbol{\eta}, \qquad \mathbf{x}^{(\ell)}(\boldsymbol{\xi}) = \sum_{\mathbf{k}\in\mathbb{Z}^d} \widehat{\mathbf{x}}^{(\ell)}(\mathbf{k}) e^{i\mathbf{k}\cdot\boldsymbol{\xi}}. \tag{27}$$

The inverse Fourier series is understood in $L^2(\mathbb{T}^d)$. Since $\mathbf{x}^{(\ell)}$ is real-valued,

$$\widehat{\mathbf{x}}^{(\ell)}(-\mathbf{k}) = \overline{\widehat{\mathbf{x}}^{(\ell)}(\mathbf{k})}. \tag{28}$$

Let

$$\Lambda_d = \begin{cases} \Lambda_R, & d = 2, \\ \Lambda_{cub}, & d = 3, \end{cases} \tag{29}$$

where Λ_R and Λ_{cub} are the retained spectral regions defined in Secs. 3.2 and 3.3.

For a fixed input $\widehat{\mathbf{x}}^{(\ell)}$, let $\mathbf{T}_d^{(\ell)}(\mathbf{k}, \mathbf{m}; \widehat{\mathbf{x}}^{(\ell)})$ denote the mode-to-mode transfer from input Fourier mode $\mathbf{m}$ to output Fourier mode $\mathbf{k}$ induced by the BDRA Scan and the corresponding State-Space Operators. For $d = 3$, this transfer additionally includes the Tri-Plane Projection and Tri-Plane Volumetric Fusion.

Then the spatial action of $G_{\Lambda_d}^{(\ell)}$ admits the input-adaptive kernel representation

$$\left[G_{\Lambda_d}^{(\ell)}\left(\mathbf{x}^{(\ell)}\right)\right](\boldsymbol{\xi}) = \int_{\mathbb{T}^d} \mathcal{K}_{G,d}^{(\ell)}\left(\boldsymbol{\xi}, \boldsymbol{\eta}; \widehat{\mathbf{x}}^{(\ell)}\right) \mathbf{x}^{(\ell)}(\boldsymbol{\eta})\, d\boldsymbol{\eta}, \tag{30}$$

where

$$\mathcal{K}_{G,d}^{(\ell)} = K_{\mathrm{SSM},d}^{(\ell)} + k_{\mathrm{out},d}\mathbf{I}_C, \tag{31}$$

and $\mathbf{I}_C$ denotes the $C \times C$ identity matrix. The component induced by the State-Space Operators is

$$K_{\mathrm{SSM},d}^{(\ell)}\left(\boldsymbol{\xi}, \boldsymbol{\eta}; \widehat{\mathbf{x}}^{(\ell)}\right) = \frac{1}{(2\pi)^d} \sum_{\mathbf{k}\in\Lambda_d} \sum_{\mathbf{m}\in\Lambda_d} \mathbf{T}_d^{(\ell)}\left(\mathbf{k}, \mathbf{m}; \widehat{\mathbf{x}}^{(\ell)}\right) e^{i(\mathbf{k}\cdot\boldsymbol{\xi} - \mathbf{m}\cdot\boldsymbol{\eta})}. \tag{32}$$

The complementary spectral modes induce

$$k_{\mathrm{out},d}(\boldsymbol{\xi} - \boldsymbol{\eta}) = \delta_{\mathbb{T}^d}(\boldsymbol{\xi} - \boldsymbol{\eta}) - \frac{1}{(2\pi)^d} \sum_{\mathbf{k}\in\Lambda_d} e^{i\mathbf{k}\cdot(\boldsymbol{\xi}-\boldsymbol{\eta})}. \tag{33}$$

For a fixed input, and hence fixed transfer coefficients, $K_{\mathrm{SSM},d}^{(\ell)}$ is a finite Fourier sum and is continuous in $(\boldsymbol{\xi}, \boldsymbol{\eta})$. The Dirac term in $k_{\mathrm{out},d}$ is understood in the distributional sense and represents the identity action on the complementary Fourier modes preserved by $G_{\Lambda_d}^{(\ell)}$.

Proof. **Step 1: State-Space Operator.**

Consider one BDRA scan path. We use $s \in \{+, -\}$ for the two-dimensional operator and $s = (p, r)$, where $p \in \{xy, xz, yz\}$ and $r \in \{+, -\}$, for the three-dimensional operator. Let $\{\overline{\mathbf{A}}_{s,i}^{(\ell)}, \overline{\mathbf{B}}_{s,i}^{(\ell)}, \mathbf{C}_{s,i}^{(\ell)}\}_{i=1}^{N_s}$ and $\mathbf{D}_s^{(\ell)}$ denote the discretized coefficients of the corresponding State-Space Operator after its input sequence is fixed.

For a serialized real–imaginary input sequence $\mathbf{v} = (\mathbf{v}_1, \ldots, \mathbf{v}_{N_s})^T$, the state-space update with initial state $\mathbf{h}_0 = \mathbf{0}$ is

$$\mathbf{h}_i = \overline{\mathbf{A}}_{s,i}^{(\ell)}\mathbf{h}_{i-1} + \overline{\mathbf{B}}_{s,i}^{(\ell)}\mathbf{v}_i, \qquad \mathbf{o}_i = \mathbf{C}_{s,i}^{(\ell)}\mathbf{h}_i + \mathbf{D}_s^{(\ell)}\mathbf{v}_i, \qquad i = 1, \ldots, N_s. \tag{34}$$

For $j < i$, define the ordered product

$$\prod_{q=j+1}^{i} \overline{\mathbf{A}}_{s,q}^{(\ell)} := \overline{\mathbf{A}}_{s,i}^{(\ell)}\overline{\mathbf{A}}_{s,i-1}^{(\ell)} \cdots \overline{\mathbf{A}}_{s,j+1}^{(\ell)}, \tag{35}$$

with the empty product for $j = i$ defined as the identity matrix. Expanding the state recursively gives

$$\mathbf{h}_i = \sum_{j=1}^{i} \left(\prod_{q=j+1}^{i} \overline{\mathbf{A}}_{s,q}^{(\ell)} \right) \overline{\mathbf{B}}_{s,j}^{(\ell)}\mathbf{v}_j. \tag{36}$$

Substituting Eq. 36 into Eq. 34 yields

$$\mathbf{o}_i = \sum_{j=1}^{i} \mathbf{H}_{s;i,j}^{(\ell)}\mathbf{v}_j, \tag{37}$$

where

$$\mathbf{H}^{(\ell)}_{s;i,j} = \begin{cases} \mathbf{C}^{(\ell)}_{s,i}\left(\prod_{q=j+1}^{i} \overline{\mathbf{A}}^{(\ell)}_{s,q}\right)\overline{\mathbf{B}}^{(\ell)}_{s,j}, & 1 \le j < i \le N_s, \\ \mathbf{C}^{(\ell)}_{s,i}\overline{\mathbf{B}}^{(\ell)}_{s,i} + \mathbf{D}^{(\ell)}_{s}, & 1 \le j = i \le N_s, \\ \mathbf{0}, & 1 \le i < j \le N_s. \end{cases} \tag{38}$$

Thus, once the input-dependent coefficients are fixed, the State-Space Operator is represented by the block lower-triangular matrix $\mathbf{H}^{(\ell)}_s$. Its lower-triangular structure follows from the order of the corresponding BDRA sequence.

Step 2: BDRA Scan and spectral mode coupling.

Define the real–imaginary representation

$$\mathcal{R}(\mathbf{z}) = \begin{bmatrix} \mathrm{Re}(\mathbf{z}) \\ \mathrm{Im}(\mathbf{z}) \end{bmatrix}. \tag{39}$$

For $d = 2$, let $\sigma_\pm : \Lambda_R \to \{1, \dots, |\Lambda_R|\}$ be the bijections induced by the BDRA orders $\pi_\pm$. After the two State-Space outputs are restored to their Fourier locations and summed, the updated coefficients on Λ_R satisfy

$$\begin{aligned} \mathcal{R}\left(\widehat{\mathbf{S}}^{(\ell)}_R(\mathbf{k})\right) = &\sum_{\mathbf{m}\in\Lambda_R} \mathbf{H}^{(\ell)}_{+;\sigma_+(\mathbf{k}),\sigma_+(\mathbf{m})}\mathcal{R}\left(\widehat{\mathbf{x}}^{(\ell)}(\mathbf{m})\right) \\ &+ \sum_{\mathbf{m}\in\Lambda_R} \mathbf{H}^{(\ell)}_{-;\sigma_-(\mathbf{k}),\sigma_-(\mathbf{m})}\mathcal{R}\left(\widehat{\mathbf{x}}^{(\ell)}(\mathbf{m})\right), \qquad \mathbf{k}\in\Lambda_R. \end{aligned} \tag{40}$$

The corresponding transfer on the retained real–imaginary representation is

$$\widetilde{\mathbf{T}}^{(\ell)}_2(\mathbf{k},\mathbf{m}) = \mathbf{H}^{(\ell)}_{+;\sigma_+(\mathbf{k}),\sigma_+(\mathbf{m})} + \mathbf{H}^{(\ell)}_{-;\sigma_-(\mathbf{k}),\sigma_-(\mathbf{m})}, \qquad \mathbf{k},\mathbf{m}\in\Lambda_R. \tag{41}$$

For $d = 3$, let $\mathbf{P}^{(\ell)}_p$ denote the matrix representation of the Tri-Plane Projection $\mathcal{P}^{(\ell)}_p$, let $\mathbf{\Pi}_{p,\pm}$ denote the block permutation induced by $\pi_{p,\pm}$, and let $\mathbf{L}_p$ denote the broadcast operation used in the Tri-Plane Volumetric Fusion. After vectorizing the retained cuboid and the three planar representations, Eqs. 12–14 give

$$\widetilde{\mathbf{T}}^{(\ell)}_3 = \sum_{p\in\{xy,xz,yz\}} \mathbf{L}_p\left(\mathbf{\Pi}^{-1}_{p,+}\mathbf{H}^{(\ell)}_{p,+}\mathbf{\Pi}_{p,+} + \mathbf{\Pi}^{-1}_{p,-}\mathbf{H}^{(\ell)}_{p,-}\mathbf{\Pi}_{p,-}\right)\mathbf{P}^{(\ell)}_p. \tag{42}$$

For either $d = 2$ or $d = 3$, the real transfers in Eqs. 41 and 42 act on the concatenated real and imaginary components of the retained Fourier coefficients. After these components are recombined, the resulting dependence on a complex coefficient may involve both $\widehat{\mathbf{x}}^{(\ell)}(\mathbf{m})$ and $\overline{\widehat{\mathbf{x}}^{(\ell)}(\mathbf{m})}$. Since $\mathbf{x}^{(\ell)}$ is real-valued, Eq. 28 gives

$$\overline{\widehat{\mathbf{x}}^{(\ell)}(\mathbf{m})} = \widehat{\mathbf{x}}^{(\ell)}(-\mathbf{m}).$$

Moreover, both retained regions are symmetric about the zero-frequency mode: if $\mathbf{k} \in \Lambda_d$, then $-\mathbf{k} \in \Lambda_d$, so all conjugate contributions can be re-indexed over the retained full spectrum. We therefore denote the resulting input-dependent transfer from input mode $\mathbf{m}$ to output mode $\mathbf{k}$ by

$$\mathbf{T}^{(\ell)}_d\left(\mathbf{k},\mathbf{m};\widehat{\mathbf{x}}^{(\ell)}\right).$$

Accordingly, the updated retained coefficient at mode $\mathbf{k}$ is

$$\sum_{\mathbf{m}\in\Lambda_d} \mathbf{T}^{(\ell)}_d\left(\mathbf{k},\mathbf{m};\widehat{\mathbf{x}}^{(\ell)}\right)\widehat{\mathbf{x}}^{(\ell)}(\mathbf{m}), \qquad \mathbf{k}\in\Lambda_d. \tag{43}$$

By the retained-spectrum replacement in Eqs. 9 and 15, the spatial output of $G^{(\ell)}_{\Lambda_d}$ is

$$\left[G^{(\ell)}_{\Lambda_d}\left(\mathbf{x}^{(\ell)}\right)\right](\boldsymbol{\xi}) = \mathcal{F}^{-1}\left[\begin{cases} \sum_{\mathbf{m}\in\Lambda_d} \mathbf{T}^{(\ell)}_d\left(\mathbf{k},\mathbf{m};\widehat{\mathbf{x}}^{(\ell)}\right)\widehat{\mathbf{x}}^{(\ell)}(\mathbf{m}), & \mathbf{k}\in\Lambda_d, \\ \widehat{\mathbf{x}}^{(\ell)}(\mathbf{k}), & \mathbf{k}\notin\Lambda_d \end{cases}\right](\boldsymbol{\xi}). \tag{44}$$

Step 3: Inverse Fourier transform.

Expanding the inverse Fourier transform in Eq. 44 gives

$$\left[G_{\Lambda_d}^{(\ell)}\left(\mathbf{x}^{(\ell)}\right)\right](\boldsymbol{\xi}) = \sum_{\mathbf{k}\in\Lambda_d}\sum_{\mathbf{m}\in\Lambda_d} \mathbf{T}_d^{(\ell)}\left(\mathbf{k},\mathbf{m};\widehat{\mathbf{x}}^{(\ell)}\right)\widehat{\mathbf{x}}^{(\ell)}(\mathbf{m})e^{i\mathbf{k}\cdot\boldsymbol{\xi}} + \sum_{\mathbf{k}\notin\Lambda_d}\widehat{\mathbf{x}}^{(\ell)}(\mathbf{k})e^{i\mathbf{k}\cdot\boldsymbol{\xi}}. \tag{45}$$

Substituting the Fourier coefficients from Eq. 27 into the retained-spectrum term gives

$$\begin{aligned}&\sum_{\mathbf{k}\in\Lambda_d}\sum_{\mathbf{m}\in\Lambda_d} \mathbf{T}_d^{(\ell)}\left(\mathbf{k},\mathbf{m};\widehat{\mathbf{x}}^{(\ell)}\right)\widehat{\mathbf{x}}^{(\ell)}(\mathbf{m})e^{i\mathbf{k}\cdot\boldsymbol{\xi}}\\ &= \sum_{\mathbf{k}\in\Lambda_d}\sum_{\mathbf{m}\in\Lambda_d} \mathbf{T}_d^{(\ell)}\left(\mathbf{k},\mathbf{m};\widehat{\mathbf{x}}^{(\ell)}\right)\left[\frac{1}{(2\pi)^d}\int_{\mathbb{T}^d}\mathbf{x}^{(\ell)}(\boldsymbol{\eta})e^{-i\mathbf{m}\cdot\boldsymbol{\eta}}\,d\boldsymbol{\eta}\right]e^{i\mathbf{k}\cdot\boldsymbol{\xi}}.\end{aligned} \tag{46}$$

For a fixed input $\widehat{\mathbf{x}}^{(\ell)}$, the transfer coefficients are fixed with respect to the integration variable $\boldsymbol{\eta}$. Since Λ_d is finite, the finite sums and the integral can be exchanged:

$$\begin{aligned}&\sum_{\mathbf{k}\in\Lambda_d}\sum_{\mathbf{m}\in\Lambda_d} \mathbf{T}_d^{(\ell)}\left(\mathbf{k},\mathbf{m};\widehat{\mathbf{x}}^{(\ell)}\right)\widehat{\mathbf{x}}^{(\ell)}(\mathbf{m})e^{i\mathbf{k}\cdot\boldsymbol{\xi}}\\ &= \int_{\mathbb{T}^d}\left[\frac{1}{(2\pi)^d}\sum_{\mathbf{k}\in\Lambda_d}\sum_{\mathbf{m}\in\Lambda_d} \mathbf{T}_d^{(\ell)}\left(\mathbf{k},\mathbf{m};\widehat{\mathbf{x}}^{(\ell)}\right)e^{i(\mathbf{k}\cdot\boldsymbol{\xi}-\mathbf{m}\cdot\boldsymbol{\eta})}\right]\mathbf{x}^{(\ell)}(\boldsymbol{\eta})\,d\boldsymbol{\eta}.\end{aligned} \tag{47}$$

The bracketed expression is $K_{\mathrm{SSM},d}^{(\ell)}(\boldsymbol{\xi},\boldsymbol{\eta};\widehat{\mathbf{x}}^{(\ell)})$ in Eq. 32.

Step 4: Preserved complementary modes.

For the complementary Fourier modes,

$$\begin{aligned}\sum_{\mathbf{k}\notin\Lambda_d}\widehat{\mathbf{x}}^{(\ell)}(\mathbf{k})e^{i\mathbf{k}\cdot\boldsymbol{\xi}} &= \int_{\mathbb{T}^d}\left[\sum_{\mathbf{k}\notin\Lambda_d}\frac{1}{(2\pi)^d}e^{i\mathbf{k}\cdot(\boldsymbol{\xi}-\boldsymbol{\eta})}\right]\mathbf{x}^{(\ell)}(\boldsymbol{\eta})\,d\boldsymbol{\eta}\\ &= \int_{\mathbb{T}^d}\left[\delta_{\mathbb{T}^d}(\boldsymbol{\xi}-\boldsymbol{\eta}) - \frac{1}{(2\pi)^d}\sum_{\mathbf{k}\in\Lambda_d}e^{i\mathbf{k}\cdot(\boldsymbol{\xi}-\boldsymbol{\eta})}\right]\mathbf{x}^{(\ell)}(\boldsymbol{\eta})\,d\boldsymbol{\eta}.\end{aligned} \tag{48}$$

The bracketed expression is $k_{\mathrm{out},d}(\boldsymbol{\xi}-\boldsymbol{\eta})$ in Eq. 33. Combining Eqs. 47 and 48 gives Eq. 30 with the decomposition in Eq. 31. This completes the proof. □

For $\mathbf{k},\mathbf{m}\in\Lambda_R$, let $q_{\mathbf{k}} = \lfloor\|\mathbf{k}\|_2 + \frac{1}{2}\rfloor$ and $q_{\mathbf{m}} = \lfloor\|\mathbf{m}\|_2 + \frac{1}{2}\rfloor$ denote their radial ring indices. Since both π_+ and π_- traverse the rings from inner to outer, $q_{\mathbf{m}} > q_{\mathbf{k}}$ implies $\sigma_\pm(\mathbf{m}) > \sigma_\pm(\mathbf{k})$. Together with the lower-triangular structure of $\mathbf{H}_\pm^{(\ell)}$, Eq. 41 gives

$$\widetilde{\mathbf{T}}_2^{(\ell)}(\mathbf{k},\mathbf{m}) = \mathbf{0}, \qquad q_{\mathbf{m}} > q_{\mathbf{k}}.$$

Thus, through the State-Space Operator, a mode in an inner ring does not depend on modes from outer rings under the BDRA ordering. Within each ring, π_+ and π_- traverse the modes in opposite angular directions.

C Dataset Details

We evaluate S^4-dyn on 15 diverse 2D and 3D physical systems. Table 3 summarizes their spatial resolution, number of physical fields, number of temporal steps per trajectory, and total number of trajectories. For datasets from The Well (Ohana et al., 2024), we use the official training, validation, and test partitions distributed with the benchmark, without re-splitting the trajectories. For FNO-NS and PDEArena-NS, we follow the dataset splits and evaluation settings of AMO (Song & Jiang, 2026) and CViT (Wang et al., 2025), respectively, and report results on their corresponding test sets.

Table 3: Summary of the datasets used in our experiments. N_{fields} counts each scalar-, vector-, or tensor-valued physical field as one field. N_{steps} and N_{traj} denote the number of temporal steps per trajectory and the total number of trajectories, respectively.

Dataset	Dim.	Resolution	N_{fields}	N_{steps}	N_{traj}
The Well: 2D systems					
Acoustic Maze	2D	256×256	4	100	8,000
Active Matter	2D	256×256	4	81	360
Gray–Scott	2D	128×128	2	1,001	1,200
Planet SWE	2D	256×512	2	1,008	120
Rayleigh–Bénard	2D	512×128	3	200	1,750
Shear Flow	2D	128×256	3	200	1,120
TRL 2D	2D	128×384	3	101	90
The Well: 3D systems					
MHD 64	3D	64^3	3	100	100
Neutron Star Merger	3D	$192 \times 128 \times 66$	8	181	8
Rayleigh–Taylor	3D	128^3	2	120	45
Supernova Explosion	3D	128^3	4	59	1,000
TGC	3D	64^3	4	50	2,700
TRL 3D	3D	$128 \times 128 \times 256$	3	101	90
Additional Navier–Stokes systems					
FNO-NS	2D	64×64	1	20	1,200
PDEArena-NS	2D	128×128	2	14	7,800

Acoustic Maze (Ohana et al., 2024). This dataset models the propagation of acoustic waves through materials with spatially varying density:

$$\frac{\partial p}{\partial t} + K(x,y)\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) = 0, \qquad \frac{\partial u}{\partial t} + \frac{1}{\rho(x,y)}\frac{\partial p}{\partial x} = 0, \qquad \frac{\partial v}{\partial t} + \frac{1}{\rho(x,y)}\frac{\partial p}{\partial y} = 0. \quad (49)$$

Where, ρ is the material density, (u, v) is the velocity, p is the pressure, and K is the bulk modulus. In the Maze configuration, the walls have $\rho = 10^6$, the paths have $\rho = 3$, and the initial pressure field contains one to six randomly placed high-pressure rings. The data contain velocity, pressure, material density, and sound-speed fields on a 256×256 grid.

Active Matter (Ohana et al., 2024). This dataset describes active particles immersed in a Stokes fluid. The particle distribution $\Psi(\mathbf{x}, \mathbf{p}, t)$ follows the Smoluchowski equation

$$\frac{\partial \Psi}{\partial t} + \nabla_{\mathbf{x}} \cdot (\dot{\mathbf{x}}\Psi) + \nabla_{\mathbf{p}} \cdot (\dot{\mathbf{p}}\Psi) = 0, \quad (50)$$

coupled to the Stokes equations

$$-\Delta \mathbf{u} + \nabla P = \nabla \cdot \mathbf{\Sigma}, \qquad \nabla \cdot \mathbf{u} = 0. \quad (51)$$

The data contain concentration, velocity, orientation-tensor, and strain-rate-tensor fields. The simulations use a periodic square domain of length $L = 10$ with 256^2 spatial modes.

Gray–Scott (Ohana et al., 2024). The Gray–Scott reaction–diffusion system describes the concentrations of two chemical species, A and B:

$$\frac{\partial A}{\partial t} = \delta_A \Delta A - AB^2 + f(1 - A), \qquad \frac{\partial B}{\partial t} = \delta_B \Delta B + AB^2 - (f + k)B. \quad (52)$$

Where, f and k are the feed and kill rates, while δ_A and δ_B are the diffusion constants. The simulations use $\delta_A = 0.00002$ and $\delta_B = 0.00001$ on the doubly periodic domain $[-1, 1]^2$, discretized by a 128×128 Fourier grid. Six choices of (f, k) produce glider, bubble, maze, worm, spiral, and spot patterns.

Planet SWE (Ohana et al., 2024). This dataset solves the rotating, forced, hyperviscous spherical shallow-water equations

$$\frac{\partial \mathbf{u}}{\partial t} = -\mathbf{u} \cdot \nabla \mathbf{u} - g\nabla h - \nu \nabla^4 \mathbf{u} - 2\mathbf{\Omega} \times \mathbf{u}, \qquad \frac{\partial h}{\partial t} = -H\nabla \cdot \mathbf{u} - \nabla \cdot (h\mathbf{u}) - \nu \nabla^4 h + F. \quad (53)$$

The physical fields are the horizontal velocity $\mathbf{u}$ and surface height h. Initial conditions are derived from the 500-hPa level of ERA5, and the time-dependent forcing F introduces daily and annual seasonality. The fields are represented on a 256×512 equiangular spherical grid.

Rayleigh–Bénard (Ohana et al., 2024). This dataset considers a two-dimensional horizontally periodic fluid heated from below and cooled from above. Its buoyancy b, velocity $\mathbf{u} = (u_x, u_z)$, and pressure p satisfy

$$\frac{\partial b}{\partial t} - \kappa \Delta b = -\mathbf{u} \cdot \nabla b, \qquad \frac{\partial \mathbf{u}}{\partial t} - \nu \Delta \mathbf{u} + \nabla p - b\mathbf{e}_z = -\mathbf{u} \cdot \nabla \mathbf{u}. \tag{54}$$

The thermal diffusivity κ and viscosity ν are determined by the Rayleigh and Prandtl numbers. The dataset contains buoyancy, pressure, and velocity fields on a 512×128 grid.

Shear Flow (Ohana et al., 2024). This dataset models a two-dimensional periodic incompressible shear flow governed by

$$\frac{\partial \mathbf{u}}{\partial t} - \nu \Delta \mathbf{u} + \nabla p = -\mathbf{u} \cdot \nabla \mathbf{u}. \tag{55}$$

A passive tracer s follows

$$\frac{\partial s}{\partial t} - D \Delta s = -\mathbf{u} \cdot \nabla s. \tag{56}$$

The viscosity and tracer diffusivity are $\nu = \mathrm{Reynolds}^{-1}$ and $D = (\mathrm{Reynolds} \times \mathrm{Schmidt})^{-1}$. The data contain tracer, velocity, and pressure fields on a 128×256 grid.

TRL 2D (Ohana et al., 2024). The two-dimensional turbulent radiative layer dataset solves the fluid equations with a radiative cooling term:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0, \qquad \frac{\partial (\rho \mathbf{v})}{\partial t} + \nabla \cdot (\rho \mathbf{v}\mathbf{v} + P) = 0, \tag{57}$$

$$\frac{\partial E}{\partial t} + \nabla \cdot ((E + P)\mathbf{v}) = -\frac{E}{t_{\mathrm{cool}}}, \qquad E = \frac{P}{\gamma - 1}, \quad \gamma = \frac{5}{3}. \tag{58}$$

The data contain density, pressure, and velocity fields. The cooling time t_{cool} is varied across trajectories, and the fields are stored on a 128×384 grid.

MHD 64 (Ohana et al., 2024). This dataset solves the ideal magnetohydrodynamic equations

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0, \tag{59}$$

$$\frac{\partial (\rho \mathbf{v})}{\partial t} + \nabla \cdot \left[\rho \mathbf{v}\mathbf{v} + \left(p + \frac{\mathbf{B}^2}{8\pi} \right) \mathbf{I} - \frac{\mathbf{B}\mathbf{B}}{4\pi} \right] = \mathbf{f}, \qquad \frac{\partial \mathbf{B}}{\partial t} - \nabla \times (\mathbf{v} \times \mathbf{B}) = 0. \tag{60}$$

Where, ρ, $\mathbf{v}$, $\mathbf{B}$, and p denote density, velocity, magnetic field, and gas pressure. The simulations use periodic boundary conditions and the isothermal equation of state $p = c_s^2 \rho$. We use the 64^3 version containing density, velocity, and magnetic-field data.

Neutron Star Merger (Ohana et al., 2024). This dataset models the hot, dense accretion disk formed after the merger of two neutron stars. The simulations solve the general relativistic equations of ideal magnetohydrodynamics coupled to lepton conservation and neutrino transport. The data contain fluid density, internal energy, electron fraction, temperature, entropy, velocity, magnetic field, and the time-independent contravariant space–time metric. The fields are represented on a logarithmic quasi-spherical grid of size $192 \times 128 \times 66$.

Rayleigh–Taylor (Ohana et al., 2024). This dataset describes the instability between two initially stationary fluids with different densities. For miscible fluids with a common molecular diffusivity, the governing equations are

$$\partial_t \rho + \nabla \cdot (\rho \mathbf{u}) = 0, \qquad \partial_t (\rho \mathbf{u}) + \nabla \cdot (\rho \mathbf{u}\mathbf{u}) = -\nabla p + \nabla \cdot \boldsymbol{\tau} + \rho \mathbf{g}, \tag{61}$$

$$\nabla \cdot \mathbf{u} = -\kappa \nabla \cdot \left(\frac{\nabla \rho}{\rho} \right), \qquad \boldsymbol{\tau} = \rho \nu \left[\nabla \mathbf{u} + (\nabla \mathbf{u})^T - \frac{2}{3} (\nabla \cdot \mathbf{u}) \mathbf{I} \right]. \tag{62}$$

The data contain density and velocity fields on a 128^3 grid. Trajectories vary in the Atwood number and in the spectral initialization of the density interface.

Supernova Explosion (Ohana et al., 2024). This dataset models an explosion in a compressed monatomic ideal gas with specific-heat ratio $\gamma = 5/3$:

$$P = (\gamma - 1)\rho u, \qquad \frac{d\rho}{dt} = -\rho \nabla \cdot \mathbf{v}, \tag{63}$$

$$\frac{d^2\mathbf{r}}{dt^2} = -\frac{\nabla P}{\rho} + \mathbf{a}_{\text{visc}} - \nabla\Phi, \qquad \frac{du}{dt} = -\frac{P}{\rho}\nabla \cdot \mathbf{v} + \frac{\Gamma - \Lambda}{\rho}. \tag{64}$$

Where, P, ρ, and u denote pressure, smoothed density, and specific internal energy. The data contain pressure, density, temperature, and velocity fields. We use the 128^3 version of the dataset.

TGC (Ohana et al., 2024). The turbulence-gravity-cooling dataset uses the same compressible monatomic ideal-gas system as the Supernova Explosion dataset. Trajectories vary in their initial turbulent velocity field, initial temperature, hydrogen number density, and metallicity, which controls radiative cooling and heating. The data contain pressure, density, temperature, and velocity fields on a 64^3 grid.

TRL 3D (Ohana et al., 2024). The three-dimensional turbulent radiative layer dataset follows the same continuity, momentum, and energy equations as TRL 2D, including the cooling term $-E/t_{\text{cool}}$. It contains density, pressure, and velocity fields on a $128 \times 128 \times 256$ grid. The cooling time t_{cool} is varied across trajectories.

FNO-NS (Li et al., 2021). The FNO-NS benchmark considers the two-dimensional incompressible Navier–Stokes equation in vorticity form:

$$\partial_t w(\mathbf{x}, t) + \mathbf{u}(\mathbf{x}, t) \cdot \nabla w(\mathbf{x}, t) = \nu \Delta w(\mathbf{x}, t) + f(\mathbf{x}), \qquad \nabla \cdot \mathbf{u}(\mathbf{x}, t) = 0, \tag{65}$$

$$w(\mathbf{x}, 0) = w_0(\mathbf{x}), \qquad \mathbf{x} \in (0, 1)^2. \tag{66}$$

Where, $\mathbf{u}$ is the velocity, $w = \nabla \times \mathbf{u}$ is the vorticity, w_0 is the initial vorticity, $\nu = 10^{-5}$ is the viscosity, and f is the forcing function. Each trajectory contains 20 vorticity frames at a resolution of 64×64. 1000 trajectories are used for training and 200 for testing.

PDEArena-NS (Gupta & Brandstetter, 2023). This benchmark considers the two-dimensional incompressible Navier–Stokes equations in velocity–pressure form, coupled to a transported scalar concentration c:

$$\frac{\partial c}{\partial t} + \mathbf{u} \cdot \nabla c = 0, \qquad \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p - \nu \nabla^2 \mathbf{u} = \mathbf{f}, \qquad \nabla \cdot \mathbf{u} = 0. \tag{67}$$

Where, $\mathbf{u}$ is the velocity, $\nu = 0.01$ is the kinematic viscosity, and the external buoyancy force is $\mathbf{f} = (0, 0.5)^T$. The equations are solved on a closed 128×128 domain with Dirichlet boundary conditions for velocity and Neumann boundary conditions for the scalar field. Each trajectory contains 14 frames sampled every 1.5 seconds. 5,200 trajectories are used for training, 1,300 for validation, and 1,300 for testing.

D Experimental Details

D.1 Training and Evaluation

We train S^4-dyn with the relative L_2 loss using AdamW with a weight decay of 10^{-4}. The learning rate is scheduled by OneCycleLR with its default settings and a maximum learning rate of 10^{-3}. On The Well, models are trained for 100 epochs with a batch size of 4. On FNO-NS and PDEArena-NS, models are trained for 500 epochs with batch sizes of 32 and 8, respectively. Checkpoints are selected based on validation VRMSE on The Well and validation relative L_2 error on PDEArena-NS. Since FNO-NS does not provide a validation split, we use the checkpoint from the final training epoch for evaluation. All experiments are run in single precision on an NVIDIA A100 80GB GPU.

For a ground-truth physical field $\mathbf{u}$ and its prediction $\widehat{\mathbf{u}}$, the relative L_2 error is

$$L_{\text{rel}}(\mathbf{u}, \widehat{\mathbf{u}}) = \frac{\|\mathbf{u} - \widehat{\mathbf{u}}\|_2}{\|\mathbf{u}\|_2}. \tag{68}$$

On The Well, we follow the evaluation setting of Flower (Muser et al., 2026). Four observed frames are used to predict the next frame, and the prediction is recursively fed back during autoregressive

rollout. We report the next-step VRMSE and the mean VRMSE over rollout steps 1:20 and 21:60. Following The Well (Ohana et al., 2024), VRMSE is defined as

$$\mathrm{VRMSE}\left(\mathbf{u}, \widehat{\mathbf{u}}\right) = \left(\frac{\left\langle \left|\mathbf{u} - \widehat{\mathbf{u}}\right|^2 \right\rangle}{\left\langle \left|\mathbf{u} - \overline{\mathbf{u}}\right|^2 \right\rangle + \epsilon} \right)^{1/2}, \qquad \epsilon = 10^{-7}, \tag{69}$$

where $\langle \cdot \rangle$ denotes spatial averaging and $\overline{\mathbf{u}}$ is the spatial mean of the target field. The reported values are averaged over physical fields and test trajectories, and additionally over the corresponding rollout steps for each rollout interval. After the four input frames, Supernova Explosion and TGC contain 55 and 46 target frames, respectively. Their final rollout intervals therefore correspond to steps 21:55 and 21:46.

FNO-NS and PDEArena-NS use ten input frames and are evaluated by autoregressive rollout over 10 and 4 prediction steps, respectively. Their performance is measured by the relative L_2 error in Eq. 68.

D.2 Model Details

Table 4 lists the dataset-specific configurations of S^4-dyn. All State-Space Operators use the standard Mamba configuration (Gu & Dao, 2023) with state dimension $d_{\text{state}} = 16$, convolution width $d_{\text{conv}} = 4$, and expansion ratio 2. We set `dt_rank` to `auto`, `dt_min` $= 10^{-3}$, `dt_max` $= 0.1$, `dt_init` to `random`, `dt_scale` $= 1$, and `dt_init_floor` $= 10^{-4}$. Convolutional bias is enabled, whereas the linear projection bias is disabled. The fast Mamba implementation is used. The Local Convolutional Operator uses a kernel size of 3 in all experiments. All Fourier-domain operations are performed using `FFT` and `IFFT`, with BDRA applied to the retained Fourier region. Hermitian symmetry is enforced on the merged spectrum before the inverse transform to ensure a real-valued spatial output.

Table 4: Dataset-specific configurations of S^4-dyn. C_z and Latent Res. denote the channel dimension and spatial resolution of the latent state in the Spatial Feature Modulation Branch. Spec. Cutoff denotes R in 2D and the shared axis cutoff $R_x = R_y = R_z$ in 3D. Latent Cutoff denotes (R_x^z, R_y^z) in 2D and (R_x^z, R_y^z, R_z^z) in 3D.

Dataset	C	L	Spec. Cutoff	C_z	Latent Res.	Latent Cutoff
The Well: 2D systems						
Acoustic Maze	128	6	24	64	(32,32)	(8,8)
Active Matter	128	6	24	64	(32,32)	(8,8)
Gray–Scott	128	6	24	64	(32,32)	(6,6)
Planet SWE	128	6	24	64	(32,64)	(8,16)
Rayleigh–Bénard	128	6	24	64	(64,16)	(16,4)
Shear Flow	128	6	24	64	(32,64)	(8,16)
TRL 2D	128	6	24	64	(32,48)	(6,8)
The Well: 3D systems						
MHD 64	96	4	12	48	(16,16,16)	(4,4,4)
Neutron Star Merger	96	4	16	48	(16,16,11)	(4,4,3)
Rayleigh–Taylor	96	4	18	48	(16,16,16)	(4,4,4)
Supernova Explosion	96	4	18	48	(16,16,16)	(4,4,4)
TGC	96	4	16	48	(16,16,16)	(4,4,4)
TRL 3D	112	4	16	48	(16,16,32)	(4,4,8)
Additional Navier–Stokes systems						
FNO-NS	128	6	18	64	(32,32)	(8,8)
PDEArena-NS	128	6	20	64	(32,32)	(8,8)

The comparison methods use the configurations reported by Flower (Muser et al., 2026) on The Well, AMO (Song & Jiang, 2026) on FNO-NS, and CViT (Wang et al., 2025) on PDEArena-NS. Flower is trained by us on FNO-NS and PDEArena-NS under the corresponding dataset settings.

E RESULTS ON FNO-NS AND PDEARENA-NS DATASETS

Tables 5 and 6 report the full relative L_2 comparisons on FNO-NS and PDEArena-NS. S^4-dyn obtains errors of 0.0276 and 0.0085, compared with 0.0278 for AMO on FNO-NS and 0.0103 for Flower on PDEArena-NS.

Table 5: Results on FNO-NS.

Model	Relative L_2 error↓
FNO (Li et al., 2021)	0.0417
U-FNO (Wen et al., 2022)	0.2231
F-FNO (Tran et al., 2023)	0.2322
LNO (Wang & Wang, 2024)	0.0845
ONO (Xiao et al., 2024)	0.1195
MWT (Gupta et al., 2021)	0.1541
Galerkin (Cao, 2021)	0.1401
LSM (Wu et al., 2023)	0.1535
OFormer (Li et al., 2023)	0.1705
Transolver (Wu et al., 2024)	0.0879
Transolver++ (Luo et al., 2025)	0.1010
LaMO (Tiwari et al., 2025)	0.0460
AMO (Song & Jiang, 2026)	<u>0.0278</u>
Flower (Muser et al., 2026)	0.0447
S^4-dyn	**0.0276**

Table 6: Results on PDEArena-NS.

Model	Relative L_2 error↓
FNO (Li et al., 2021)	0.0912
F-FNO (Tran et al., 2023)	0.0839
GK-T (Cao, 2021)	0.0952
GNOT (Hao et al., 2023)	0.1720
OFormer (Li et al., 2023)	0.1350
DPOT-L (Hao et al., 2024)	0.0278
CViT-L (Wang et al., 2025)	0.0235
Flower (Muser et al., 2026)	<u>0.0103</u>
S^4-dyn	**0.0085**

F INFERENCE TIME

Table 7 compares parameter count, next-timestep prediction time, and peak GPU memory. All models are evaluated with a batch size of 1 on a single NVIDIA A100 GPU. For each model, the time required to predict one future frame is measured over five runs and reported as mean ± standard deviation. Peak GPU memory is recorded under the same setting.

Table 7: Computational efficiency on Active Matter (2D, 256^2) and Rayleigh–Taylor (3D, 128^3). Next-timestep prediction time is measured with a batch size of 1 on a single NVIDIA A100 GPU and reported as mean ± standard deviation over five runs. Lower is better. An en-dash indicates that the model does not support 3D inputs.

	Active Matter (2D, 256^2)			Rayleigh–Taylor (3D, 128^3)		
Model	Params ↓	Per-Frame Time (s) ↓	Peak GPU Mem. (GB) ↓	Params ↓	Per-Frame Time (s) ↓	Peak GPU Mem. (GB) ↓
FNO	18.94M	0.00883 ± 0.00100	0.64	298.13M	0.17584 ± 0.00171	11.86
CNU-Net	18.59M	0.00874 ± 0.00084	0.24	22.29M	0.27119 ± 0.00062	4.43
scOT	17.73M	0.05874 ± 0.00091	0.15	–	–	–
Flower	17.34M	0.01312 ± 0.00276	0.48	23.92M	0.29255 ± 0.00192	12.27
S^4-dyn	6.59M	0.05069 ± 0.00112	0.92	6.98M	1.46451 ± 0.00449	13.59

G MORE VISUALIZATION RESULTS

Figures 5 and 6 provide the full qualitative comparisons on the two 2D systems. On Active Matter, S^4-dyn better preserves the locations and shapes of the high- and low-concentration regions, while its absolute-error map contains only weak residuals around local concentration variations. On TRL 2D, it follows the Kelvin–Helmholtz roll-up of the density interface between the cold dense and hot dilute phases, with residual error concentrated along the mixing layer.

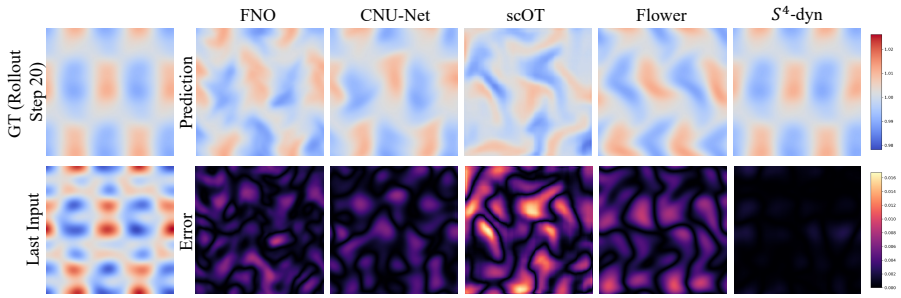

Figure 5: Active Matter concentration at rollout step 20.

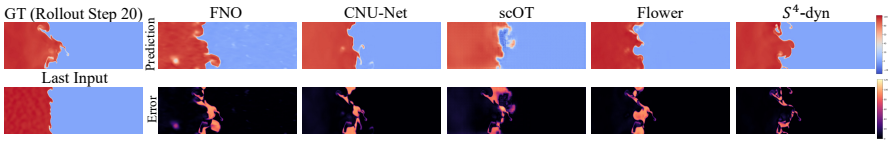

Figure 6: TRL 2D density at rollout step 20.

Figure 7 further compares Flower and S^4-dyn on Acoustic Maze from rollout steps 10 to 60. S^4-dyn more closely follows the pressure wave as it propagates through the maze paths, with smaller errors around the propagating wavefronts and the interfaces between the maze paths and walls. In contrast, Flower accumulates larger errors at these locations as the rollout progresses.

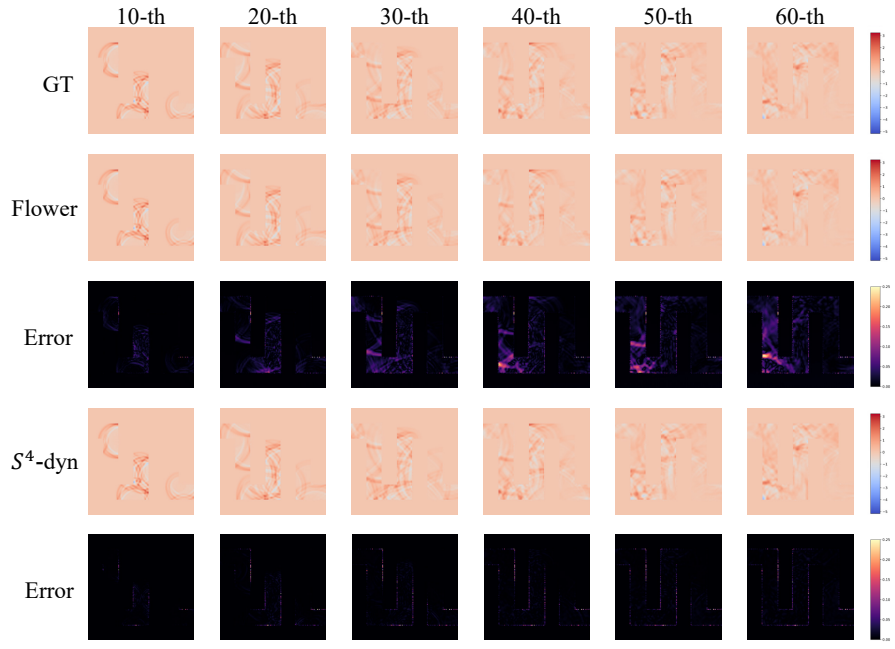

Figure 7: Pressure predictions and absolute-error maps of Flower and S^4-dyn on Acoustic Maze at rollout steps 10–60.

Figure 8 further shows the learned spatial modulation fields on Active Matter across different network layers and rollout steps. Since $\gamma^{(\ell)}$ and $\beta^{(\ell)}$ contain multiple feature channels, we compute the

root-mean-square (RMS) across the channel dimension for visualization:

$$\gamma^{sp,(\ell,t)}(x,y) = \sqrt{\frac{1}{C}\sum_{c=1}^{C}\left(\gamma_c^{(\ell,t)}(x,y)\right)^2}, \qquad \beta^{sp,(\ell,t)}(x,y) = \sqrt{\frac{1}{C}\sum_{c=1}^{C}\left(\beta_c^{(\ell,t)}(x,y)\right)^2}. \tag{70}$$

Where, C denotes the hidden feature dimension, ℓ denotes the network layer, and t denotes the rollout step. The resulting maps show that the spatial modulation varies across network layers and rollout steps.

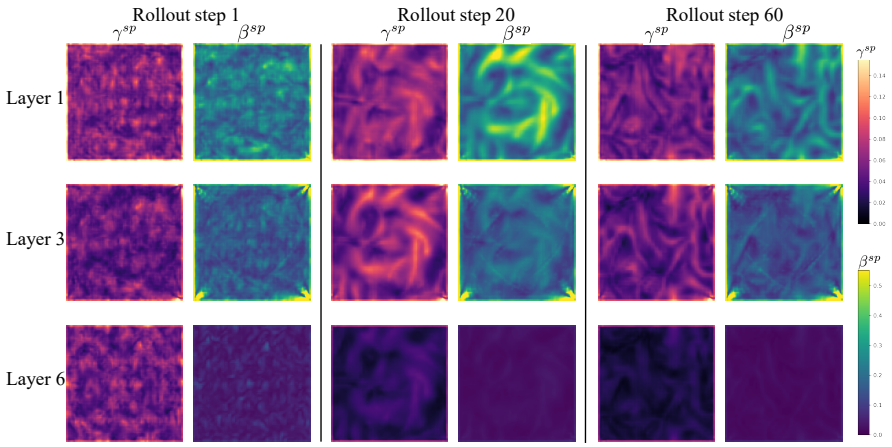

Figure 8: Spatial modulation fields on the Active Matter dataset.