DRESO: AN ADAPTIVE COMPLEMENTARY SPECTRAL NEURAL OPERATOR FOR ROBUST AUTOREGRESSIVE PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Neural operators learn mappings between function spaces and provide fast surrogates for partial differential equations (PDEs), but accurate one-step prediction does not guarantee stable autoregressive rollouts. DRIFT-Net combines a spectral branch for global low-frequency structure with an image branch for local details and nonstationary patterns. The two branches are fused through lightweight band-wise weighting (Li & Salim, 2026). However, DRIFT-Net's hard spectral partition does not support direct gradient-based optimization of the cutoff, while its band-wise energy summary provides limited information about the spectral structure within each region. To address these limitations, we propose DRESO, a dual-region evidence-guided spectral operator, which consists of a learnable complementary filter (LCF) and an adaptive gain controller (AGC). First, LCF designs a smooth Gaussian low-pass filter with a learnable cutoff and its complementary high-pass filter. This design enables each spectral layer to adaptively learn its low–high frequency allocation during training. Then, based on the low- and high-frequency regions provided by LCF, AGC extracts ten spectral descriptors from each region, converts their normalized comparisons into sample–channel-specific gains, and uses these gains to recombine the independently transformed regional spectra before the inverse FFT. These descriptors provide richer regional spectral information for adaptive routing than raw features or low-dimensional spectral summaries. On the six CamLab-ETHZ tasks, DRESO-L achieves the lowest error on all six one-step problems. DRESO-B ranks second in each comparison and outperforms DRIFT-Net with 32.9% fewer parameters. DRESO-B also uses 44.0% fewer parameters than NESTOR-Small, the strongest competing baseline. Code is available at anonymous code repository.

1 INTRODUCTION

Partial differential equations (PDEs) model systems ranging from fluids and waves to climate and materials. Repeated high-resolution solutions remain expensive when conditions or physical parameters vary. Neural operators address this repeated-solve setting by learning mappings between function spaces rather than solutions at isolated coordinates (Kovachki et al., 2023). FNOs parameterize such mappings through a global spectral kernel (Li et al., 2021). Recent systems (Herde et al., 2024; Hao et al., 2024) extend operator learning to larger and more diverse PDE corpora. These advances support the use of neural operators as surrogates, but neural operators can achieve high one-step accuracy while performing poorly in autoregressive rollouts.

This gap between one-step accuracy and rollout performance arises in part from the recursive nature of autoregressive forecasting. At step t, the prediction $\widehat{u}_{t+1}$ becomes the input for the next update, so a small prediction error can alter the input spectrum at subsequent time steps. Vortices, fronts, and small-scale waves often leave signatures in non-dominant modes. A model that suppresses these modes may still attain low one-step error while losing useful information over successive prediction steps.

To address the limitation, existing methods explore different spectral processing strategies. FNO (Li et al., 2021) and related operators such as F-FNO (Tran et al., 2023) and G-FNO (Helwig et al.,

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.

2023) retain low modes through ideal truncation in their spectral branches. DRIFT-Net (Li & Salim, 2026) couples controlled spectral processing with a spatial branch, while FreqMoE (Chen et al., 2025), TF-SNO (Zhou et al., 2026), and RNO (Lu et al., 2025) explore increasingly fine-grained routing or alternative representations. However, these methods do not jointly provide a smooth learnable complementary partition, explicit region-specific channel transformations, and a tractable sample–channel controller. The structural requirement is to derive routing evidence from the partition being recombined while avoiding the cost of per-frequency control.

To meet the requirement, we introduce DRESO, a dual-region evidence-guided spectral operator. DRESO combines a learnable complementary filter (LCF) with an adaptive gain controller (AGC). **(i) LCF.** PDE-Refiner (Lippe et al., 2023) reports that retaining non-dominant components can improve long autoregressive prediction. Inspired by this finding, LCF forms low- and high-frequency regions with a radial Gaussian response and its complement. The two responses sum to one, preserving the full Fourier spectrum. The Gaussian cutoff is continuous, differentiable, and optimized separately in each spectral block, allowing blocks at different depths to learn their own frequency allocations. Independent frequency-shared complex mixers then specialize the channel transformations in the two regions. **(ii) AGC.** Radar spectral analysis uses energy and spectral shape to characterize underlying processes (Kollias et al., 2011). In multichannel signal analysis, relationships between channels provide information beyond individual power spectra (Carter, 1987). Motivated by these perspectives, AGC extracts ten descriptors from each LCF region to summarize energy, spectral shape, frequency location, and cross-channel relationships. AGC jointly standardizes the low–high descriptor pairs and converts the comparisons into regional scores through two evidence paths. The linear path is initialized with an empirical DSP prior, while a nonlinear residual path learns task- and state-dependent corrections. The combined scores produce two bounded gains for every sample and channel. A positive gain floor keeps both regions active, enabling selective regional recombination without discarding either route.

The coupling between LCF and AGC is central to the design. LCF supplies the regional spectra to the mixers and AGC; AGC extracts descriptors from these spectra and uses the resulting gains to recombine the transformed regions before the inverse FFT. During autoregressive rollout, every predicted state produces new regional descriptors and new gains, allowing the balance between the two regions to respond to the evolving trajectory. The coupled design jointly learns spectral coverage, regional transformation, and sample–channel routing while avoiding the control cost of per-frequency gates.

Our contributions are as follows:

- We introduce DRESO, a new dual-region evidence-guided spectral operator, consisting of a LCF and an AGC module, which achieves stable autoregressive rollouts.
- We design LCF with a smooth Gaussian response and its complement to form overlapping low- and high-frequency components while retaining all Fourier modes. The smooth response supports gradient-based cutoff learning in each spectral block, enabling adaptive low–high frequency allocation during training.
- We propose an AGC module that uses regional spectral descriptors to determine gains for each sample and channel. These gains can respond to state-dependent spectral changes while avoiding per-frequency control.
- We develop three variants, DRESO-L, DRESO-B, and DRESO-T, with different parameter budgets. In the main benchmark comparisons, DRESO-L ranks first on all six CamLab-ETHZ one-step tasks, five of six CamLab-ETHZ rollouts, and all four The Well one-step tasks. The 11.48M-parameter DRESO-B outperforms the 17.10M DRIFT-Net on all twelve CamLab-ETHZ one-step and rollout metrics. Controlled ablations and inference-time masking examine the integrated design and its routing evidence.

2 RELATED WORK

Fourier and spectral operators. Fourier Neural Operator (FNO) parameterizes a translation-invariant integral kernel with a truncated Fourier expansion (Li et al., 2021). AFNO replaces dense spectral mixing with token-wise Fourier filtering and sparsity-inducing thresholding (Guibas et al.,

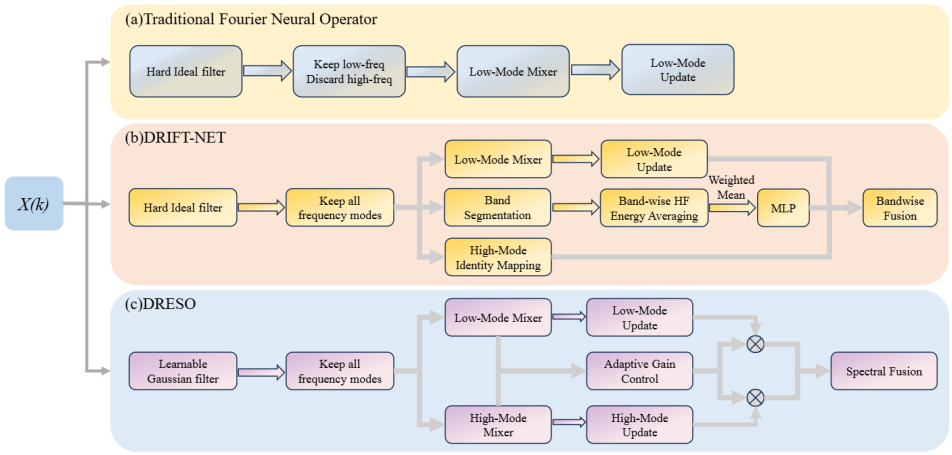


Figure 1: Comparison of frequency-domain processing. (a) Illustrates the spectral processing pipeline of classic FNO. It retains a hard low-mode subset and applies mode-specific channel mixing. (b) Outlines bandwise frequency processing in DRIFT-Net. It supplies its Bandwise operation with a high-frequency weighted mean for each band. (c) Details complementary spectral processing and adaptive gain control in DRESO. It preserves complementary Gaussian low/high regions, applies frequency-shared regional mixers, and conditions its dynamic gate on ten spectral descriptors. $X(k)$ denotes the Fourier-domain input.

2022), while DPOT combines Fourier attention with autoregressive denoising pretraining (Hao et al., 2024). In the FNO/AFNO low-mode filtering lineage, the retained spectrum is selected by a fixed budget or an idealized hard response; DPOT likewise uses Fourier-domain attention as a principal computational pathway. However, these methods rely on predefined spectral selection or filtering mechanisms rather than learning smooth, complementary frequency regions. Hard low-mode selection introduces a discontinuity at the cutoff and does not provide a differentiable mechanism for each layer to learn its regional allocation. *In contrast, our* DRESO *uses a smooth Gaussian response and its complement, preserves all Fourier modes, and learns the cutoff jointly with the operator objective.*

Large-scale PDE learning. POSEIDON uses a multiscale operator transformer with time-conditioned normalization and an all-to-all strategy motivated by the semigroup property (Herde et al., 2024). DPOT scales operator pretraining through autoregressive denoising and a Fourier-attention backbone (Hao et al., 2024). NESTOR combines an image-level mixture of experts for global dependencies with token-level experts for local structure (Sun et al., 2026). These systems demonstrate the value of scale, but they route information through different networks, tokens, or attention pathways rather than adaptively balancing complementary spectral regions within each channel.

Frequency-aware and adaptive operators. PDE-Refiner uses iterative denoising to improve all-frequency modeling and motivates retaining non-dominant components for autoregressive prediction (Lippe et al., 2023). FreqMoE follows a low-frequency pretraining/high-frequency fine-tuning schedule and activates spectral experts with a sample-wise gate (Chen et al., 2025). TF-SNO extracts four frequency-domain and four physical-space statistics and, in its original form, modulates each frequency component for every sample and channel (Zhou et al., 2026). RNO maps fields to a sinogram representation and combines angle-aware global processing with local spatial updates (Lu et al., 2025). DRIFT-Net is most closely related to our approach in its use of regional spectral processing: it partitions the spectrum into bands and aggregates the high-frequency energy fraction within each band before bandwise fusion (Li & Salim, 2026). However, its bandwise aggregation relies on a scalar energy summary rather than separately extracting multiple spectral descriptors from the complementary regions for adaptive gating. *These methods either apply an ideal low-pass response or use a hard cutoff registered as a learnable scalar. Differ from them, our method proposes a differentiable Gaussian filter. Its response has a nonzero smooth derivative, reduces Gibbs-type cutoff artifacts, and allows each layer to adapt its low/high allocation.* AGC extracts multiple regional

descriptors and produces two gains per sample and channel. Together, these components learn complementary regions and perform evidence-guided routing without per-frequency modulation.

3 Problem Setup

Let $D \subset \mathbb{R}^d$ be a spatial domain and consider a time-dependent PDE

$$\partial_t u(x,t) + \mathcal{L}\big(u, \nabla u, \nabla^2 u, \ldots; \lambda\big) = 0, \quad x \in D, \quad t \in (0,T], \tag{1}$$

with boundary operator $\mathcal{B}(u) = 0$ and initial condition $u(x,0) = a(x)$. The parameter λ may include coefficients, forcing, or boundary data. For a state space $\mathcal{X}$, the solution operator $\mathcal{S}_\tau : \mathcal{X} \to \mathcal{X}$ satisfies $u(t+\tau) = \mathcal{S}_\tau(u(t))$.

On a grid, $u_t \in \mathbb{R}^{C_{\text{in}} \times H \times W}$. We learn $\mathcal{G}_\theta$ such that $\widehat{u}_{t+\Delta t} = \mathcal{G}_\theta(u_t, \Delta t)$. The one-step task evaluates this map under ground-truth inputs. A K-step rollout recursively sets

$$\widehat{u}_{t+(k+1)\Delta t} = \mathcal{G}_\theta(\widehat{u}_{t+k\Delta t}, \Delta t), \qquad k = 0, \ldots, K-1. \tag{2}$$

We use relative L_1 error (RelL1) for both one-step and rollout evaluation. Its precise definition and the benchmark-specific aggregation protocol are provided in Appendix A.

4 Method

4.1 Architecture overview

Figure 2 shows the internal data flow of the two coupled components. For a hidden field $x \in \mathbb{R}^{B \times C \times H \times W}$, we compute a real two-dimensional FFT, $X = \mathcal{F}(x) \in \mathbb{C}^{B \times C \times H \times W_f}$. LCF applies a learnable Gaussian response and its complement to form overlapping low- and high-frequency components while retaining all modes. Independent frequency-shared complex mixers transform the two components. In parallel, AGC computes regional spectral descriptors and normalizes the low–high comparisons. It converts these comparisons into decision evidence and produces sample–channel gains. The gated regional updates are fused before the inverse FFT and residual update.

Outside this redesigned spectral module, we retain the established DRIFT-Net spatial paths and ConvNeXt (Liu et al., 2022) channel-processing blocks. Within the module, LCF determines both the mixer inputs and the spectral descriptors supplied to AGC, while AGC controls the recombination of the independently transformed regions.

4.2 Learnable complementary filter (LCF)

Hard partition and gradient obstruction. Let $\rho(k) \in [0,1]$ be the normalized radial frequency at Fourier coordinate k. An ideal low-pass filter uses a binary response,

$$I_L(\rho; c) = \mathbf{1}\{\rho \le c\}, \qquad I_H(\rho; c) = 1 - I_L(\rho; c). \tag{3}$$

On a discrete Fourier grid, its cutoff derivative is

$$\frac{\partial I_L(\rho; c)}{\partial c} = \begin{cases} 0, & \rho \neq c, \\ \text{undefined}, & \rho = c, \end{cases} \qquad \frac{\partial I_L(\rho_j; \sigma(\alpha))}{\partial \alpha} = 0 \quad \text{a.e.} \tag{4}$$

Declaring c a learnable parameter does not enable gradient-based optimization of the hard cutoff. Its derivative with respect to c is zero almost everywhere under ordinary backpropagation. Optimizing the cutoff therefore requires a search procedure or a surrogate gradient.

Smooth complementary response. We instead learn a smooth cutoff $c \in (0,1)$, parameterized as $c = \sigma(\alpha)$, and define the LCF responses

$$M_L(\rho; c) = \exp\left(-\frac{\rho^2}{2c^2}\right), \qquad M_H(\rho; c) = 1 - M_L(\rho; c). \tag{5}$$

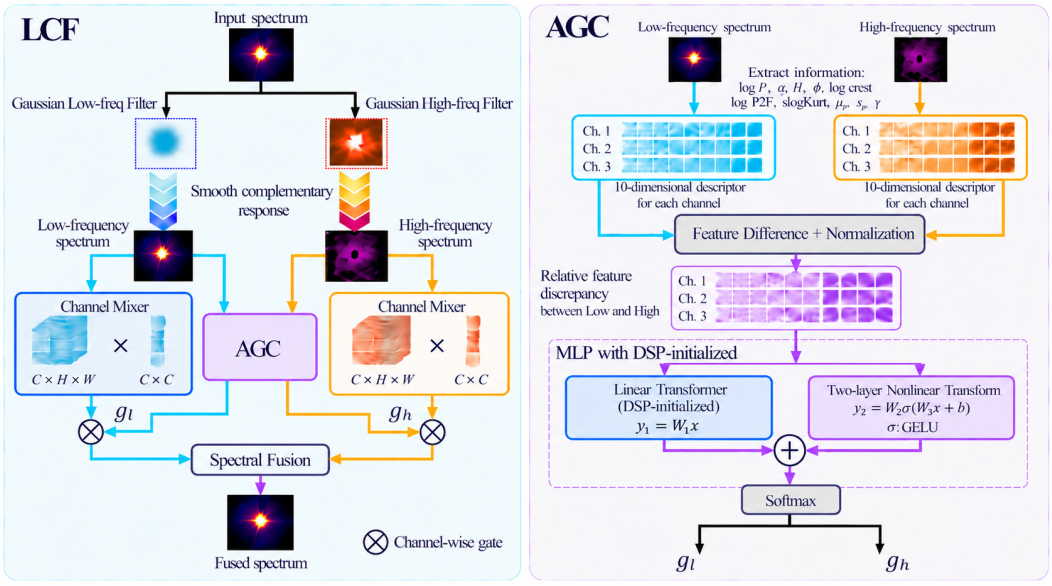


Figure 2: Detailed illustration of our DRESO. LCF forms complementary Gaussian low- and high-frequency views, applies independent frequency-shared channel mixers, and fuses their gated updates. AGC computes regional spectral descriptors, forms normalized low–high comparisons, and converts them into decision evidence before producing the low/high gains.

The regional spectra $X_L = M_L \odot X$ and $X_H = M_H \odot X$ satisfy $X_L + X_H = X$ exactly. Unlike an ideal cutoff, both regions remain differentiable and no mode is deleted. In particular, the derivative used by backpropagation is

$$\frac{\partial M_L(\rho; c)}{\partial c} = M_L(\rho; c)\frac{\rho^2}{c^3}, \qquad \frac{\partial M_L}{\partial \alpha} = M_L(\rho; c)\frac{\rho^2}{c^3}c(1-c), \tag{6}$$

which is finite and nonzero for $0 < \rho$ and $0 < c < 1$. The smooth response also avoids the abrupt frequency jump of Equation 3; the resulting spatial-localization and ringing comparison is proved in Appendix B. The experimental results further reveal that the learned cutoffs are broadly distributed across blocks and layers and exhibit structured variation; see Appendix C.4.

Regional channel mixing. The low- and high-frequency mixers use complex matrices $W_L, W_H \in \mathbb{C}^{C\times C}$:

$$\widetilde{X}_r(b, :, k) = W_r X_r(b, :, k), \qquad r \in \{L, H\}. \tag{7}$$

Separate matrices allow the two complementary spectral regions to learn different cross-channel couplings. Neither region alone determines the resulting error.

4.3 Adaptive Gain Controller (AGC)

Spectral descriptors and normalization. For each sample b, channel c, and region r, we summarize the current spectrum by

$$z_{bcr} = [\log P,\ q,\ \mathcal{H},\ \Phi,\ \log \text{crest},\ \log \text{P2F},\ \text{slogKurt},\ \mu_\rho,\ s_\rho,\ \gamma] \in \mathbb{R}^{10}. \tag{8}$$

Here P is regional power; q is its share of total power; $\mathcal{H}$ and Φ are normalized spectral entropy and flatness; crest and P2F are peak-to-mean and peak-to-floor ratios; slogKurt is signed log-kurtosis; μ_ρ and s_ρ are spectral centroid and spread; and γ measures cross-channel coherence. The descriptors are computed independently from X_L and X_H, while q already compares the two regional powers. For each sample b, channel c, and descriptor index j, we standardize the two regional candidates jointly:

$$\begin{aligned} \mu_{bc}^{(j)} &= \frac{1}{2}\sum_{r\in\{L,H\}} \phi_{bcr}^{(j)}, \qquad (\sigma_{bc}^{(j)})^2 = \frac{1}{2}\sum_{r\in\{L,H\}} \left(\phi_{bcr}^{(j)} - \mu_{bc}^{(j)}\right)^2, \\ z_{bcr}^{(j)} &= \frac{\phi_{bcr}^{(j)} - \mu_{bc}^{(j)}}{\sqrt{(\sigma_{bc}^{(j)})^2 + \epsilon_{\text{norm}}}}, \qquad \epsilon_{\text{norm}} = 10^{-5}. \end{aligned} \tag{9}$$

DSP-informed routing and gain allocation. After normalization, a router with DSP-informed initialization converts these comparisons into AGC decision evidence and two regional gains. The DSP coefficients specify an empirical initialization ratio:

$$\boldsymbol{w}_0 = [0,\ 0.35,\ -0.50,\ -0.50,\ 0.20,\ 0.25,\ 0.15,\ 0,\ 0,\ 0.40].$$

The entries follow the order $(\log P, q, \mathcal{H}, \Phi, \log \mathrm{crest}, \log \mathrm{P2F}, \mathrm{slogKurt}, \mu_\rho, s_\rho, \gamma)$, with their signs encoding the initial direction of the evidence comparison. The positive power-share weight favors the higher-energy region, while the negative entropy and flatness weights favor more concentrated spectra. Zero-initialized terms remain neutral at initialization.

For a batch of hidden fields, the normalized tensor has shape $Z \in \mathbb{R}^{B\times C\times 2\times 10}$. The complete regional logit is

$$\ell_{bcr} = \boldsymbol{w}^\top z_{bcr} + \boldsymbol{v}^\top \mathrm{GELU}(W z_{bcr} + \boldsymbol{b}) + c, \tag{10}$$

where $W \in \mathbb{R}^{64\times 10}$, $\boldsymbol{b}, \boldsymbol{v} \in \mathbb{R}^{64}$, and $c \in \mathbb{R}$. During training, it learns task- and layer-specific corrections. Appendix E.11 reports the learned linear-coefficient distributions across the spectral blocks of trained DRESO-L in Figure 11. The scalar ℓ_{bcr} is the regional score. We then normalize the two logits along the region axis and impose a lower bound of 0.25 on the resulting gains:

$$a_{bcr} = \mathrm{softmax}_r(\ell_{bcr}), \quad g_{bcr} = 0.25 + 1.5 a_{bcr}, \tag{11}$$

where $r \in \{L, H\}$. Therefore $a_{bcr} \in [0, 1]$, $g_{bcr} \in [0.25, 1.75]$, and $g_{bcL} + g_{bcH} = 2$. The minimum gain prevents either route from being switched off. Experiments confirm that, despite the lower bound of 0.25, neither spectral region is assigned the minimum gain during inference; see Appendix C.3. The gains are dynamic because the descriptors are recomputed from each current input state. The spectral update is

$$Y_{bc}(k) = g_{bcL}\widetilde{X}_{L,bc}(k) + g_{bcH}\widetilde{X}_{H,bc}(k), \qquad y = \mathcal{F}^{-1}(Y). \tag{12}$$

This creates only two gains per sample and channel. Section 5.4 contrasts this granularity with alternative controllers, and Appendix C.1 gives complete feature definitions.

4.4 Training objective

We follow DRIFT-Net (Li & Salim, 2026) and train DRESO with a relative L_1 reconstruction term plus a radial frequency-weighted auxiliary term. Let $\widehat{u}_{t+1} = \mathcal{G}_\theta(u_t)$, u_{t+1} be the target state, and define the prediction residual and its two-dimensional real FFT as

$$E_{b,c}(x) = \widehat{u}_{t+1,b,c}(x) - u_{t+1,b,c}(x), \qquad \widehat{E}_{b,c}(k) = \mathrm{rFFT}_2(E_{b,c})\,(k). \tag{13}$$

For a batch of B samples, with $\mathcal{K}$ denoting the retained rFFT grid, the spatial reconstruction term and the frequency-weighted term are

$$\mathcal{L}_{\mathrm{RelL1}} = \frac{1}{B}\sum_{b=1}^{B} \frac{\sum_{c,x} |E_{b,c}(x)|}{\sum_{c,x} |u_{t+1,b,c}(x)| + \varepsilon}. \tag{14}$$

$$\mathcal{L}_{\mathrm{HF}} = \frac{1}{B}\sum_{b=1}^{B}\sum_{c=1}^{C}\sum_{k\in\mathcal{K}} w(\rho(k))\ |\widehat{E}_{b,c}(k)|^2, \qquad w(\rho) = \rho^\alpha, \quad \alpha = 1.5. \tag{15}$$

The training objective is

$$\mathcal{L}_{\mathrm{train}} = \mathcal{L}_{\mathrm{RelL1}} + \lambda_{\mathrm{HF}}\mathcal{L}_{\mathrm{HF}}, \qquad \lambda_{\mathrm{HF}} = 0.3. \tag{16}$$

We set $\lambda_{\mathrm{HF}} = 0.3$ following DRIFT-Net (Li & Salim, 2026). We also report results without the high-frequency auxiliary loss in Appendix E.9.

5 Experiments

5.1 Evaluation protocol

We evaluate three DRESO scales (T/B/L) against convolutional, spectral, recurrent, and pretrained baselines under a common single-state protocol. To improve comparability in parameter count

where feasible, we adapt NESTOR and PDE-ReFiner into smaller variants, NESTOR-Small and PDE-ReFiner-Small, respectively; full-sized results are reported in Appendix E.10. CamLab-ETHZ contains Navier–Stokes (NS-Gauss, NS-Sines) and compressible Euler variants (CE-RP, CE-CRP, CE-Gauss, CE-KH) (Herde et al., 2024). The Well covers acoustic scattering, active matter, Gray–Scott reaction diffusion, and planetary weather (Ohana et al., 2024). Tables report relative L_1 error (RelL1; lower is better); protocols, sources, configurations, and metric details appear in Appendix A. Bold and dagger denote the best and runner-up values.

5.2 One-step prediction

Table 1: One-step prediction on CamLab-ETHZ tasks ($\Delta t = 1$); model labels report total parameters (M). DRESO-L is the best on all six tasks under each protocol.

Protocol	Model	NS-Gauss	CE-RP	CE-CRP	CE-Gauss	CE-KH	NS-Sines
Pretrain	NESTOR-Small 20.51M	1.948	7.243	24.361	4.806	9.561	5.021
	UNet 7.76M	6.643	14.970	28.621	11.074	10.606	9.011
	FNO 0.47M	2.744	11.871	30.940	6.684	9.391	8.625
	PDE-ReFiner-Small 36.64M	3.766	9.432	24.168	5.943	11.715	7.830
	Poseidon-T 20.77M	8.420	12.297	41.785	15.731	14.306	20.416
	DRIFT-Net 17.10M	7.140	12.011	35.051	8.229	15.007	20.789
	NIPs 0.055M	19.137	18.877	61.318	39.031	14.997	40.525
	RNO 2.84M	9.481	9.763	23.409	8.639	10.075	10.774
	DRESO-L 22.15M	**1.337**	**3.997**	**17.498**	**1.917**	**5.823**	**3.571**
	DRESO-B 11.48M	1.586†	4.370†	19.233†	2.181†	6.556†	4.151†
	DRESO-T 2.90M	2.785	7.403	22.179	3.745	7.462	9.404
Finetune-50	UNet 7.76M	4.733	9.707	24.389	6.936	9.165†	5.972
	NESTOR-Small 20.51M	1.705†	6.311†	23.495†	3.795†	9.184	4.516†
	DRIFT-Net 17.10M	5.242	9.611	32.816	7.731	13.404	15.546
	DRESO-L 22.15M	**0.770**	**3.145**	**16.251**	**1.413**	**5.704**	**2.315**
Finetune-100	UNet 7.76M	4.379	9.741	22.833†	6.128	8.836†	5.681
	NESTOR-Small 20.51M	1.613†	5.879†	22.938	3.470†	8.939	4.128†
	DRIFT-Net 17.10M	4.918	9.268	32.224	7.505	12.068	13.510
	DRESO-L 22.15M	**0.797**	**2.796**	**15.638**	**1.400**	**5.190**	**2.137**

Table 1 shows that DRESO-L ranks first on all six tasks under pretraining, Finetune-50, and Finetune-100. Under pretraining, DRESO-B is the runner-up in every case and, with 32.9% fewer parameters, outperforms DRIFT-Net on all six tasks. On The Well in Table 2, DRESO-L ranks first in all four comparisons. Relative to the strongest non-DRESO baseline, it reduces error by 56.8%, 16.6%, 11.6%, and 34.3%.

Table 2: One-step results on four The Well datasets after training from scratch ($\Delta t = 1$). DRESO-L achieves the lowest error on every system.

Model	Acoustic Scattering	Active Matter	Gray-Scott	Planetary Weather
NESTOR-Small	6.249	97.314	64.243	2.610
UNet	7.904	75.460	43.995	1.704
FNO	11.623	95.455	43.030	1.143
PDE-ReFiner-Small	10.829	52.698	40.769	2.149
Poseidon-T	6.192	36.219	36.112	2.022
DRIFT-Net	6.910	40.557	39.212	2.089
NIPs	40.142	50.707	57.052	4.085
DRESO-L	**2.672**	**30.211**	**31.927**	**0.751**
DRESO-B	2.861†	31.564†	35.711†	0.765†
DRESO-T	3.521	32.000	38.579	0.976

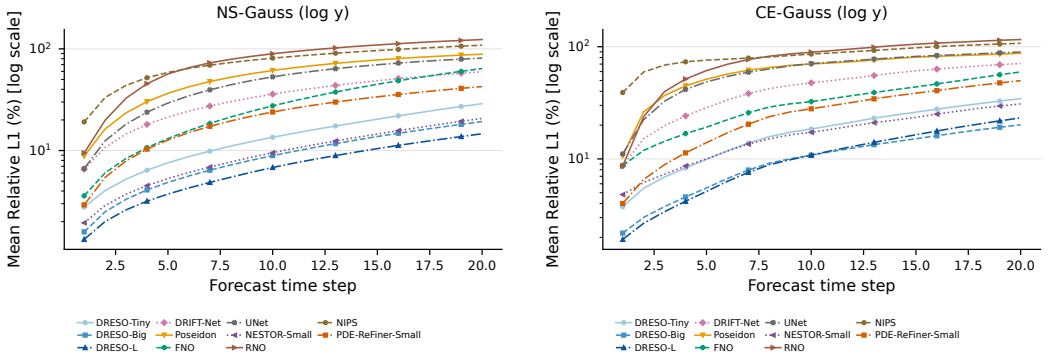
5.3 Autoregressive rollout and zero-shot transfer

Table 3 reports the pretraining results: DRESO-L ranks first on five of six datasets, while NESTOR-Small leads on CE-KH. Relative to DRIFT-Net, DRESO-L and DRESO-B extend their one-step advantage to all rollout tasks. The complete comparison under pretraining, finetuning for 50 epochs, and finetuning for 100 epochs is provided in Appendix E.2.

Figure 3 illustrates rollout dynamics on NS-Gauss and CE-Gauss, where DRESO maintains lower later-step errors; all six datasets appear in Appendix D. Additional 20-step and long-horizon The Well rollouts, including 100- and 199-step results, are reported in Appendix E.3.

Table 3: Autoregressive rollout on CamLab-ETHZ tasks under the pretraining protocol.

Protocol	Model	NS-Gauss	CE-RP	CE-CRP	CE-Gauss	CE-KH	NS-Sines
Pretrain	NESTOR-Small	10.512	27.041	52.840	17.803	**27.794**	29.968†
	UNet	50.682	69.374	81.735	64.582	37.652	73.991
	FNO	30.956	38.814	72.663	33.589	55.331	55.108
	PDE-ReFiner-Small	32.561	45.235	60.978	41.855	37.114	59.656
	Poseidon-T	50.951	58.364	96.351	65.495	54.176	81.771
	DRIFT-Net	31.753	43.611	60.038	38.161	49.587	75.499
	NIPs	77.152	104.961	120.785	85.352	70.035	97.710
	RNO	88.619	52.447	75.293	89.365	46.927	123.765
	DRESO-L	**7.485**	**15.714**	**40.372**	**11.059**	29.152†	**25.785**
	DRESO-B	9.811†	17.380†	41.722†	11.673†	31.899	30.960
	DRESO-T	14.791	26.442	45.825	19.097	34.935	44.428



(a) NS-Gauss (b) CE-Gauss

Figure 3: Error–time growth over the 20-step autoregressive rollout. The vertical axis is mean RelL1 (%) in the native normalized space; logarithmic scaling is used where indicated in the panel title.

We also evaluate transfer without target-task adaptation on five unseen systems. DRESO-L leads all five rollout comparisons and two of five one-step comparisons. The aggregate results of the zero-shot evaluation are reported in Appendix E.1.

5.4 GATE MECHANISM COMPARISON

Table 4 isolates routing with the operator fixed. FreqMoE maps raw Fourier chunks directly to sample-specific, channel-shared gates (Chen et al., 2025), whereas AGC derives sample–channel low/high gains from compact spectral descriptors. TF-SNO uses finer sample–channel–frequency modulation at $\mathcal{O}(BCHF)$ cost (Zhou et al., 2026); our adapted variant retains its eight statistics but outputs only sample–channel gates. In this setting, DRESO is best in 11 of 12 cells; TF-SNO leads only on CE-RP one-step prediction.

Table 4: Controlled comparison of gating mechanisms on CamLab-ETHZ one-step and rollout tasks. Our AGC is best in 11 of 12 cells; TF-SNO is best on CE-RP one-step prediction.

Gate mechanism	NS-Gauss		CE-RP		CE-CRP		CE-Gauss		CE-KH		NS-Sines	
	One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout
TF-SNO gate	1.553†	8.928†	**3.741**	16.663†	18.296†	41.515†	2.227†	12.923†	5.942†	31.386†	3.722†	26.920†
FreqMoE gate	2.013	13.769	4.168	23.513	20.778	48.002	3.167	20.386	6.514	32.241	5.005	37.351
Our AGC	**1.337**	**7.485**	3.997†	**15.714**	**17.498**	**40.372**	**1.917**	**11.059**	**5.823**	**29.152**	**3.571**	**25.785**

5.5 GATE INPUT ABLATION

Tables 6 and 7 assess each descriptor by removing it from an otherwise unchanged gate. Removing entropy, flatness, and coherence increases mean rollout RelL1 by 175.655%, 164.054%, and 147.917%, respectively. Entropy removal produces a 454.151% increase on CE-KH. Every removal increases one-step error on all six tasks, and nine of ten increase error across all rollout tasks. The exception is power share on NS-Gauss (-1.661%). These results support the use of all ten descriptors while showing task dependence for power share. Appendix C.2 provides corresponding rollout visualizations.

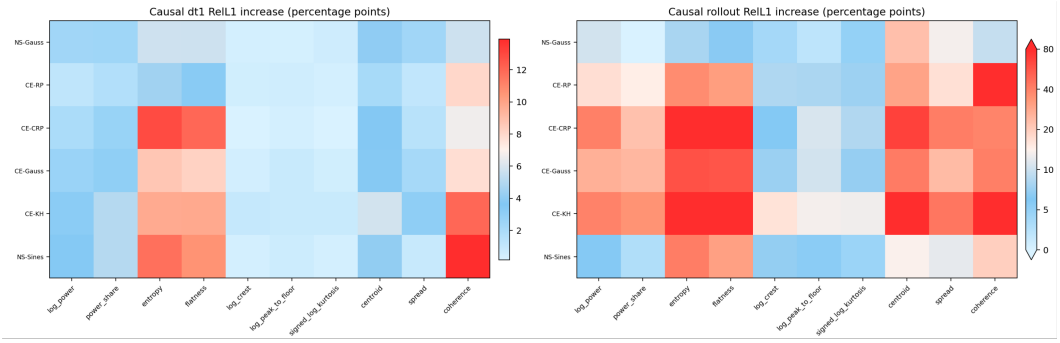


Figure 4: Causal feature ablation. The heatmaps show absolute RelL1 increases in percentage points after removing one gate input; warmer colors indicate a larger degradation. Exact relative increases, including values beyond the clipped color range, appear in Tables 6 and 7.

5.6 COMPONENT ABLATION

Table 5: Cumulative component ablation on CamLab-ETHZ. Components are added successively to the baseline; the full DRESO-L is best on 11 of 12 metrics.

Component		Model variant	NS-Gauss		CE-RP		CE-CRP		CE-Gauss		CE-KH		NS-Sines	
			One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout
		Baseline	3.118	18.881	6.256	22.689	20.026	44.923	3.435	20.928	9.461	32.982	6.609	48.833
LCF	Dual high-low region	✓	2.062	13.938	4.110	20.980	19.692	44.829	3.100	18.489	6.168	32.538	5.079	43.298
LCF	Learnable Gaussian filter	✓	1.690†	11.016†	**3.901**	17.740†	18.689†	42.830†	2.296†	13.337†	6.072†	31.822†	4.023†	31.391†
AGC	Gating	✓ **(Full Model)**	**1.337**	**7.485**	3.997†	**15.714**	**17.498**	**40.372**	**1.917**	**11.059**	**5.823**	**29.152**	**3.571**	**25.785**

Table 5 shows cumulative gains from each component: dual-region processing and the learnable Gaussian each improve all 12 metrics over their preceding configurations, while dynamic gating improves 11 of 12. This supports their complementary roles in the full model.

Appendix B provides the corresponding analysis of differentiability and cutoff artifacts. In Appendix C.5, Table 8 further isolates the filter response by freezing the cutoff, showing that the Gaussian response itself contributes beyond adaptive cutoff learning. Appendix E.6 provides trajectory-level visualizations of the full, low-, and high-frequency regions.

6 CONCLUSION

We presented DRESO, a dual-region evidence-guided spectral operator that couples learnable complementary filtering with adaptive regional recombination. LCF forms complementary spectral regions and supplies their spectral descriptors to AGC, which assigns sample–channel-specific gains to the independently transformed regions. DRESO-L achieves the lowest one-step error across six CamLab-ETHZ and four The Well tasks and the lowest rollout error on five of six CamLab-ETHZ tasks. Ablation studies further demonstrate the contributions of complementary filtering, regional processing, and adaptive gating. Future work will extend the current radial dual-region design to anisotropic, geometry-aware, and multi-region spectral decompositions.

AI USE STATEMENT

In this work, we used OpenAI Codex to assist with interpreting author-supplied experimental outputs; developing and revising mathematical exposition and proof presentation; drafting, translating, restructuring, and editing manuscript text; reviewing citation placement and literature summaries; and producing LaTeX formatting and layout changes. We also used Codex for software-engineering tasks, including creating Conda environment setup scripts, preparing basic training scripts and standardized model-evaluation commands, and implementing or adapting code for selected experimental interfaces. Generative AI was not used to fabricate, impute, or manually alter datasets, experimental measurements, or reported model outputs. We reviewed and tested all AI-assisted code in the controlled experimental environment and checked the resulting mathematical statements, citations, interpretations, and manuscript edits against the source papers, experiment records, and compiled document. The authors take responsibility for the final content of this work.

REFERENCES

G. Clifford Carter. Coherence and time delay estimation. *Proceedings of the IEEE*, 75(2):236–255, 1987. doi: 10.1109/PROC.1987.13723.

Tianyu Chen, Haoyi Zhou, Ying Li, Hao Wang, Zhenzhe Zhang, Tianchen Zhu, Shanghang Zhang, and Jianxin Li. FreqMoE: Dynamic frequency enhancement for neural PDE solvers. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pp. 7356–7364, 2025. doi: 10.24963/ijcai.2025/818.

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, 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, pp. 17616–17635, 2024.

Jacob Helwig, Xuan Zhang, Cong Fu, Jerry Kurtin, Stephan Wojtowytsch, and Shuiwang Ji. Group equivariant Fourier neural operators for partial differential equations. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pp. 12907–12930, 2023. URL https://proceedings.mlr.press/v202/helwig23a.html.

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*, volume 37, 2024.

Pavlos Kollias, Jasmine Rémillard, Edward Luke, and Wanda Szyrmer. Cloud radar doppler spectra in drizzling stratiform clouds: 1. forward modeling and remote sensing applications. *Journal of Geophysical Research: Atmospheres*, 116(D13):D13201, 2011. doi: 10.1029/2010JD015237.

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.

Jiayi Li and Flora D. Salim. DRIFT-Net: A spectral-coupled neural operator for PDEs learning. In *International Conference on Learning Representations*, 2026.

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.

Phillip Lippe, Bas Veeling, Paris Perdikaris, Richard E. Turner, and Johannes Brandstetter. PDE-Refiner: Achieving accurate long rollouts with neural PDE solvers. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Ning Liu and Yue Yu. Neural interpretable PDEs: Harmonizing fourier insights with attention for scalable and interpretable physics discovery. In *Proceedings of the 42nd International Conference on Machine Learning*, 2025. URL https://github.com/fishmoon1234/Nonlocal-Attention-Operator.

Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A ConvNet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11976–11986, 2022. doi: 10.1109/CVPR52688.2022.01167.

Wenbin Lu, Yihan Chen, Junnan Xu, Wei Li, Junwei Zhu, and Jianwei Zheng. Solving partial differential equations via radon neural operator. In *Advances in Neural Information Processing Systems*, 2025. URL https://github.com/wenbin-lu/Radon-Neural-Operator.

Ruben Ohana, Michael McCabe, Lucas Meyer, Rudy Morel, Fruzsina J. Agocs, et al. The well: A large-scale collection of diverse physics simulations for machine learning. In *Advances in Neural Information Processing Systems: Datasets and Benchmarks Track*, volume 37, 2024.

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. *arXiv preprint arXiv:2602.22059*, 2026.

Makoto Takamoto, Timothy Praditia, Raphael Leiteritz, Dan MacKinlay, Francesco Alesiani, Dirk Pflüger, and Mathias Niepert. PDEBench: An extensive benchmark for scientific machine learning. In *Advances in Neural Information Processing Systems: Datasets and Benchmarks Track*, volume 35, 2022. URL https://github.com/pdebench/PDEBench.

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

Yitian Zhou, Chaoning Zhang, Zhenzhen Huang, Haoxuan Yu, Jiaquan Zhang, Yiran Li, Fan Mo, Kuien Liu, Jie Zou, Caiyan Qin, and Yang Yang. TF-SNO: Time-frequency gated spectral neural operators for learning non-stationary partial differential equations. *arXiv preprint arXiv:2606.21189*, 2026.

APPENDIX

A EXPERIMENTAL SETUP AND HYPERPARAMETERS

A.1 OVERALL PROTOCOL

We follow the public POSEIDON downstream-task definitions and The Well splits (Herde et al., 2024; Ohana et al., 2024). POSEIDON inputs use channel-first tensors in $\mathbb{R}^{C\times 128\times 128}$, whereas every The Well experiment retains the native spatial resolution of its dataset and uses tensors in $\mathbb{R}^{C\times H_{\mathcal{D}}\times W_{\mathcal{D}}}$. Every temporal experiment learns the one-step map $\mathcal{F}_\theta : u_t \mapsto u_{t+1}$ with teacher forcing. At test time, one-step error is evaluated from a ground-truth input at $t = 0$, whereas rollout predictions are generated in closed loop by feeding $\widehat{u}_t$ back to the model, as in Equation 2.

POSEIDON incompressible Navier–Stokes files contain two physical velocity channels. For the joint four-channel interface, the adapters add constant density and pressure channels, but mask these two synthetic channels from both training loss and evaluation; only (u, v) are scored. Compressible-Euler files use the four physical channels (ρ, u, v, p). All models receive the same channel contract and no baseline receives additional history or privileged physical variables.

All reported metrics are percentage relative L_1 errors (RelL1):

$$\mathrm{RelL1}(\widehat{u}, u) = 100\,\frac{\|\widehat{u} - u\|_1}{\|u\|_1}. \tag{17}$$

For rollout tables, the same per-state RelL1 is aggregated over the evaluation trajectory according to the dataset's prescribed protocol.

A.2 DATASETS AND TASKS

The POSEIDON tasks are stored on a fixed 128×128 Cartesian grid with 21 snapshots unless stated otherwise. A time-dependent array is written below as trajectories $\times$ time $\times$ channels $\times H \times W$. We jointly pretrain on six datasets using 200 trajectories from each dataset and reserve the official validation/test portions for model selection and evaluation.

NS-Gauss. NS-Gauss is two-dimensional incompressible Navier–Stokes flow initialized by Gaussian random fields. Its velocity array has shape $20000 \times 21 \times 2 \times 128 \times 128$ and the official split is $19640/120/240$ trajectories for train/validation/test. The joint interface is $4 \times 128 \times 128$, but RelL1 is computed only on the two velocity channels.

NS-Sines. NS-Sines uses the same incompressible Navier–Stokes equations and tensor shape as NS-Gauss, but its initial fields are constructed from sinusoidal modes. The array is $20000 \times 21 \times 2 \times 128 \times 128$ with the same $19640/120/240$ split; only (u, v) are supervised and scored.

CE-RP. CE-RP contains compressible-Euler Riemann problems with piecewise-constant states separated by discontinuities. The data tensor has shape $10000 \times 21 \times 4 \times 128 \times 128$ for density, two velocity components, and pressure, with a $9640/120/240$ train/validation/test split.

CE-CRP. CE-CRP is the curved-interface Riemann-problem variant and tests whether a model can propagate non-axis-aligned shocks and contacts. It has the same $10000 \times 21 \times 4 \times 128 \times 128$ shape, four physical channels, and $9640/120/240$ split as CE-RP.

CE-Gauss. CE-Gauss evolves compressible-Euler states generated from Gaussian initial perturbations. Its tensor has shape $10000 \times 21 \times 4 \times 128 \times 128$, and we use the official $9640/120/240$ split.

CE-KH. CE-KH is the compressible Kelvin–Helmholtz instability, where shear-driven interface roll-up creates progressively finer structures. It also contains $10000 \times 21 \times 4 \times 128 \times 128$ values and uses the $9640/120/240$ split.

The following five POSEIDON tasks are excluded from joint pretraining and used only for strict zero-shot evaluation.

CE-RM. CE-RM is the compressible Richtmyer–Meshkov instability produced by the interaction of a shock with a perturbed material interface. The evaluated four-channel state has shape $1260 \times 21 \times 4 \times 128 \times 128$ and the official split is $1030/100/130$ trajectories.

NS-BB. NS-BB is incompressible Navier–Stokes flow initialized from Brownian-bridge fields. Its velocity tensor has shape $20000 \times 21 \times 2 \times 128 \times 128$ with a $19640/120/240$ split; evaluation again excludes the two synthetic padding channels.

NS-SL. NS-SL is the incompressible double-shear-layer benchmark. The velocity variable has shape $40000 \times 21 \times 2 \times 128 \times 128$ and uses $39640/120/240$ trajectories for train/validation/test.

CE-RPUI. CE-RPUI combines a compressible Riemann configuration with an unstable interface, testing shock propagation and instability growth in the same trajectory. The four-channel tensor has shape $10000 \times 21 \times 4 \times 128 \times 128$ with a $9640/120/240$ split.

Wave-Gauss. Wave-Gauss solves a scalar wave equation with a spatially varying wave-speed field. The dynamic solution has shape $10512 \times 15 \times 128 \times 128$, and the static speed map $c(x, y)$ has shape $10512 \times 128 \times 128$. We concatenate c to every input state, keep it fixed during rollout, and use the official $10212/60/240$ split.

The main The Well experiments use four two-dimensional datasets. Each subset is trained independently because its fields have different physical meanings, channel counts, and spatial resolutions. No common spatial resizing is applied: for dataset $\mathcal{D}$, a training pair retains the native shape $(C, H_{\mathcal{D}}, W_{\mathcal{D}}) \to (C, H_{\mathcal{D}}, W_{\mathcal{D}})$. The separate Shear Flow evaluation below provides an additional long-horizon stress test.

Acoustic scattering. The `acoustic_scattering_discontinuous` subset models first-order acoustic propagation through a medium with discontinuous material coefficients. Its native grid is 256×256; the three dynamic channels comprise pressure and the two components of particle velocity. The official train split yields 161,600 adjacent one-step windows under history $= 1$ and horizon $= 1$.

Active matter. The active-matter subset represents a coupled, nonequilibrium active-fluid system whose orientational order and flow interact across scales. It has 11 dynamic fields on a native 256×256 grid, 81 snapshots per trajectory, and 360 trajectories in the complete dataset. The official train split provides 14,000 adjacent one-step windows.

Gray–Scott. Gray–Scott is a two-species reaction–diffusion system; the two channels are the interacting concentrations. It contains 1,200 trajectories with 1,001 snapshots on a native 128×128 grid. The official train split supplies 960,000 adjacent windows.

Planetary weather. PlanetSWE solves the shallow-water equations on a rotating sphere and records fluid height together with two horizontal velocity components. It contains three dynamic fields on a native 256×512 angular grid, 1,008 snapshots per trajectory, and 120 trajectories. The official train split supplies 96,672 adjacent windows, all evaluated on the native 256×512 grid.

Shear Flow. The Shear Flow subset describes a two-dimensional shear-driven system with coupled velocity, pressure, and tracer fields. Each trajectory is represented as a tensor of shape $T \times 4 \times 256 \times 512$ on the native grid; Reynolds and Schmidt numbers are retained as dataset metadata. We train on adjacent one-step windows with history $= 1$ and horizon $= 1$, and evaluate the closed-loop trajectory at 100 and 199 autoregressive steps.

A.3 Dataset-specific and single-state protocols

Dataset-specific training protocol. The two benchmark families use different training organizations. The Well contains systems with different channel counts, spatial resolutions, and physical meanings. Pooling these systems would mix unrelated state variables. We therefore train each model from scratch independently on acoustic scattering, active matter, Gray–Scott reaction diffusion,

and planetary weather. Each model produces one predictor for one PDE subset and retains that subset's native spatial resolution. The CamLab-ETHZ suite uses a standardized channel dimensionality and aligned channel semantics across tasks. Following the dataset categorization defined by CamLab-ETHZ, we use the six datasets assigned to its *Pretrain* partition for joint large-scale pretraining.

Single-state protocol. Every model receives one state at the current time and predicts the next state. For the retained multi-frame baselines, we reduce the temporal input to one state. This follows the canonical Markovian neural-operator formulation $\mathcal{S}_{\Delta t} : u(t) \mapsto u(t + \Delta t)$, in which one complete state is the operator input. Multiple frames can be treated as an augmented state, but they provide additional observations that are unavailable to a single-state model. We therefore keep the temporal input budget fixed when comparing architectures. This change affects only the temporal interface. Each model retains its subsequent operator backbone. The protocol also reflects deployment settings in which acquiring, transmitting, or storing a long observation history is costly. A single-state operator can begin forecasting after one measurement, whereas a history-dependent model must first accumulate multiple frames. Appendix E.5 reports a supplementary one-step comparison under the default ten-frame input protocol for the retained history-dependent baselines.

A.4 Complete model configurations

This section records the public implementation, original architectural preset, evaluated adaptation, and parameter count of every baseline. Each paragraph gives the complete code address and then states the configuration used in our experiments. Counts in the main tables are the total trainable parameters of the evaluated single-state instances; where temporal-interface adaptation changes the count, we also report the original model size.

DRESO. DRESO is implemented in our anonymous code repository. Its three variants share a four-level encoder–decoder, patch size 4, window size 16, stage heads $(3, 6, 12, 24)$, and stage widths $d(1, 2, 4, 8)$. DRESO-T uses $d = 24$, depths $(2, 2, 2, 2)$, and skip depths $(1, 1, 1, 0)$; DRESO-B uses $d = 48$ with the same depths; DRESO-L uses $d = 48$, depths $(4, 4, 4, 4)$, and skip depths $(2, 2, 2, 0)$. The evaluated T, B, and L variants contain 2.90M, 11.48M, and 22.15M trainable parameters, respectively, matching the counts reported in the efficiency analysis. The shared spectral configuration operates on the full spectrum with the learnable complementary filter (LCF), initialized at $c = 0.25$, separate low/high mixers, a 64-unit hidden layer in the adaptive gain controller (AGC), ten normalized AGC statistics, softmax allocation with temperature 1 and floor 0.25, and identity-perturbed mixer initialization with scale 0.02.

PDEBench UNet. The PDEBench UNet baseline is instantiated from the original `UNet2d` implementation (code: `https://github.com/pdebench/PDEBench`) (Takamoto et al., 2022). It uses a four-level encoder–decoder with skip concatenations, base width `init_features`$= 32$, channel progression $(32, 64, 128, 256)$, and a 512-channel bottleneck. Each level contains two 3×3 convolutions followed by BatchNorm and Tanh, while four 2×2 max-pooling operations are mirrored by transposed-convolution upsampling. This configuration has 7.8M parameters.

PDEBench FNO. Our FNO baseline likewise uses the released PDEBench implementation rather than a reimplementation (code: `https://github.com/pdebench/PDEBench`) (Li et al., 2021; Takamoto et al., 2022). The evaluated `FNO2d` has width 20, four Fourier layers, 12×12 retained modes, a parallel 1×1 spatial convolution in each layer, two-cell nonperiodic padding, and a width-128 projection head, for a total of 0.5M parameters. For both PDEBench baselines, the original operator blocks remain unchanged; we adapt only data loading, physical-channel projections, and autoregressive wrappers.

Poseidon-T. Poseidon-T follows the official scOT implementation (code: `https://github.com/camlab-ethz/poseidon`) (Herde et al., 2024). The T preset uses patch size 4, window size 16, embedding width 48, and four encoder/decoder levels with depths $(4, 4, 4, 4)$, heads $(3, 6, 12, 24)$, and skip depths $(2, 2, 2, 0)$. It contains approximately 21M parameters. For the six-task experiment, this architecture is trained from scratch with one input frame.

DRIFT-Net. The DRIFT-Net baseline is obtained from its official codebase (code: `https://github.com/cruiseresearchgroup/DRIFT-Net`) (Li & Salim, 2026). We evaluate its T configuration with the same patch size, window size, width, depth, head, and skip schedule as Poseidon-T. Enabling `use_spectral=true` replaces the original attention blocks with DRIFT spectral–image coupled blocks, yielding approximately 17M parameters.

NIPs. For NIPs, we use the released Nonlocal Attention Operator implementation (code: `https://github.com/fishmoon1234/Nonlocal-Attention-Operator`) (Liu & Yu, 2025). Its Small adapter is configured with width 64, four layers, one attention head, and query dimension 40, resulting in 0.055M parameters.

RNO. The RNO baseline follows the official Radon Neural Operator implementation (code: `https://github.com/wenbin-lu/Radon-Neural-Operator`) (Lu et al., 2025). We set the hidden width to 128 and use eight RNO layers, eight attention heads, 32 slices, 16 projection angles, one Radon block, and angle chunks of four. This configuration contains approximately 2.8M parameters.

Fine-tuned UNet, DRIFT-Net, and DRESO rows preserve the corresponding architectures above. FT50 and FT100 denote target-task finetuning for 50 and 100 epochs, respectively, whereas the zero-shot rows use the jointly pretrained checkpoint without target-task updates.

A.5 Training and optimization

POSEIDON six-task pretraining. DRESO, Poseidon-T, and DRIFT-Net use 200 trajectories from each of the six pretraining datasets and all 20 adjacent $\Delta t = 1$ transitions, for 24,000 mappings per epoch. DRESO uses AdamW with learning rate 10^{-3}, while Poseidon-T and DRIFT-Net use their adjusted learning rate 10^{-4}. All three retain weight decay 10^{-6}, cosine decay with a 3% warm-up, batch size 32, gradient clipping at 5, and 200 epochs. NESTOR and NESTOR-Small retain their original learning rate 10^{-3} and original optimizer configuration. Following DRIFT-Net, both DRIFT-Net and DRESO use the same composite objective,

$$\mathcal{L} = \mathcal{L}_{\mathrm{RelL1}} + \lambda_{\mathrm{HF}} \mathcal{L}_{\mathrm{HF}}, \qquad \lambda_{\mathrm{HF}} = 0.3, \tag{18}$$

where the high-frequency auxiliary term uses radial exponent 1.5. This shared objective makes their comparison an architectural comparison rather than a loss-function comparison. No early stopping is applied; checkpoints are selected using validation performance.

PDEBench FNO and UNet are also trained for 200 epochs on 200 trajectories per dataset and 20 one-step mappings per trajectory. Both use AdamW with learning rate 10^{-3}, weight decay 10^{-4}, five warm-up epochs followed by cosine decay, batch size 32, relative L_1 loss, gradient clipping at 1, and validation every five epochs. NIPs-Small and RNO use 200 epochs, learning rate 5×10^{-5}, and batch size 32.

The Well training. Each model–dataset pair is initialized and trained independently for 80 epochs on the complete official train split at that dataset's native spatial resolution: 256×256 for acoustic scattering and active matter, 128×128 for Gray–Scott, and 256×512 for planetary weather. With history and horizon equal to one, every adjacent window is visited once per epoch in shuffled order and `drop_last=false`. All participating models use batch size 32. FNO and UNet use relative L_1 loss for training. RNO is excluded from The Well experiments because its memory consumption exceeds the available GPU capacity for these datasets. Validation and test data come from the official `valid` and `test` directories. Evaluation reports relative L_1 error (RelL1) for $t_0 \to t_1$ prediction and closed-loop prediction over the available 20-state sequence, corresponding to 19 autoregressive transitions, on at most 200 distinct test trajectories.

Cross-resolution zero-shot protocol. For Appendix E.7, each reported model uses its jointly trained 128×128 checkpoint without fine-tuning, calibration, or target-resolution parameter updates. The same checkpoint is evaluated at 128×128, 96×96, and 64×64; only shape-dependent grids, coordinates, or interpolated positional tables are regenerated during the forward pass. Architectural compatibility and exclusions are detailed alongside Table 15.

Hardware and precision. All experiments are run on a single NVIDIA A40 or 4090 GPU in full single precision (FP32) without mixed-precision or low-precision training. Random seeds are fixed for each run so that our results are reproducible given the released code and configuration.

B GAUSSIAN VERSUS IDEAL SPECTRAL FILTERING

This appendix formalizes the claims referenced in Sections 4 and C.4: the Gaussian cutoff is trainable by ordinary backpropagation, and its spatial kernel is better localized than the kernel of an ideal low-pass response.

B.1 DIFFERENTIABILITY OF THE LEARNED CUTOFF

For $\rho > 0$ and $c > 0$, Equation 5 satisfies

$$\frac{\partial M_L(\rho; c)}{\partial c} = M_L(\rho; c)\frac{\rho^2}{c^3} > 0, \qquad \frac{\partial \log M_L(\rho; c)}{\partial c} = \frac{\rho^2}{c^3}. \tag{19}$$

Hence every nonzero Fourier radius supplies a finite, informative gradient for the cutoff. With $c = \sigma(\alpha)$, the chain rule gives $\partial M_L/\partial \alpha = (\partial M_L/\partial c)c(1-c)$, so an unconstrained parameter α can be optimized end to end.

By contrast, an ideal low-pass mask is

$$I_L(\rho; c) = \mathbf{1}\{\rho \le c\}. \tag{20}$$

For a fixed discrete Fourier grid with radii $\{\rho_j\}$, $I_L(\rho_j; c)$ is constant on every open interval between adjacent radii. Its ordinary derivative with respect to c is therefore zero almost everywhere and undefined when $c = \rho_j$. In the continuous distributional sense the derivative is $\delta(c - \rho)$, but standard automatic differentiation does not turn this impulse into a useful gradient on the discrete threshold. A hard ideal cutoff can be selected by search or approximated with a surrogate gradient, but it is not directly learnable through the task loss in the same sense as Equation 19.

B.2 SPATIAL LOCALIZATION AND RINGING

Ignoring Fourier-convention constants, the inverse transform of a Gaussian in d dimensions is another Gaussian:

$$\mathcal{F}^{-1}\left[e^{-\|\omega\|^2/(2c^2)}\right](x) \propto c^d e^{-c^2\|x\|^2/2}. \tag{21}$$

It is positive, smooth, non-oscillatory, and exponentially localized. In two dimensions, the inverse transform of a circular ideal mask instead has the form

$$\mathcal{F}^{-1}\left[\mathbf{1}\{\|\omega\| \le c\}\right](x) \propto \frac{cJ_1(c\|x\|)}{\|x\|}, \tag{22}$$

where J_1 is the first-order Bessel function. Because J_1 changes sign and decays only algebraically in its envelope, the kernel has long oscillatory sidelobes. The inverse Fourier transform of an ideal sharp spectral filter yields an oscillatory spatial convolution kernel, which can produce nonphysical over- and undershoots (ringing) near sharp fronts or discontinuities. The Gaussian complement avoids a hard boundary, and $M_L + M_H = 1$ preserves complete spectral coverage.

Discrete FFT form. The same phenomenon appears directly on the finite grid used by the operator. For an $N_1 \times N_2$ periodic grid, let

$$\widehat{x}_{p,q} = \sum_{m=0}^{N_1-1}\sum_{n=0}^{N_2-1} x_{m,n}\exp\left[-2\pi i\left(\frac{pm}{N_1} + \frac{qn}{N_2}\right)\right] \tag{23}$$

be the discrete Fourier transform. Applying an ideal mask and transforming back gives

$$y_{m,n} = \frac{1}{N_1N_2}\sum_{p=0}^{N_1-1}\sum_{q=0}^{N_2-1} I_L(\rho_{p,q}; c)\,\widehat{x}_{p,q}\exp\left[2\pi i\left(\frac{pm}{N_1} + \frac{qn}{N_2}\right)\right] = (h_I * x)_{m,n}, \tag{24}$$

where the circular convolution kernel is

$$h_I[a,b] = \frac{1}{N_1 N_2} \sum_{p,q} I_L(\rho_{p,q}; c) \exp\left[2\pi i \left(\frac{pa}{N_1} + \frac{qb}{N_2}\right)\right]. \tag{25}$$

The indicator changes abruptly along the discrete circular boundary $\rho_{p,q} = c$. Hence the finite sum in Equation 25 contains a sharp phase-coherent truncation rather than a rapidly decaying window. Along any grid direction that crosses this boundary, the sum reduces to a partial Fourier series; for a one-dimensional slice with K retained modes,

$$h_K[a] = \frac{1}{N} \sum_{k=-K}^{K} e^{2\pi i k a/N} = \frac{\sin\big((2K+1)\pi a/N\big)}{N \sin(\pi a/N)}. \tag{26}$$

This Dirichlet kernel has alternating sidelobes whose envelope decays slowly. When it convolves a jump or a steep PDE front, the sidelobes generate overshoot and undershoot near the interface; increasing resolution narrows the oscillatory region but does not remove its limiting amplitude (the classical Gibbs effect). The Gaussian response replaces the abrupt indicator by a smooth weight, so its discrete kernel is a weighted sum with no jump at the cutoff and substantially faster spatial decay, as formalized by Equation 21.

B.3 Cutoff sensitivity on the normalized radial grid

The learned cutoff acts on a normalized radial coordinate, so a small parameter displacement changes the response of an entire frequency annulus. From the Gaussian mask in Equation 5, the response ratio after a cutoff change $c \mapsto c + \Delta$ is

$$\frac{M_L(\rho; c+\Delta)}{M_L(\rho; c)} = \exp\left[\frac{\rho^2}{2}\left(\frac{1}{c^2} - \frac{1}{(c+\Delta)^2}\right)\right]. \tag{27}$$

For the observed displacement from $c = 0.25$ to $c = 0.29$, this ratio is 1.67 at $\rho = 0.5$ and 3.18 at $\rho = 0.75$. The equal-allocation boundary follows from $M_L = M_H = 0.5$:

$$\rho_{1/2} = c\sqrt{2\log 2}. \tag{28}$$

It moves from 0.294 to 0.341, which is a 16% radial shift. Away from boundary effects, the number of low-dominant modes on a two-dimensional grid scales with the enclosed area, $N_{\text{low}} \propto \rho_{1/2}^2$. The same displacement therefore changes the affected mode count by approximately 35%. These calculations explain why visually narrow cutoff variations can still alter spectral attenuation and regional assignment across many Fourier modes.

C Gate analysis

C.1 Gate feature definitions

Let $A_j = |X_r(k_j)|^2 + \epsilon$ be regional power at normalized radii ρ_j, $p_j = A_j / \sum_i A_i$, and N_r the number of regional frequency locations. The ten inputs in Equation 8 are:

$$P = \frac{1}{N_r} \sum_j A_j, \qquad q = \frac{\sum_j A_j}{\sum_{r' \in \{L,H\}} \sum_j A_j^{(r')}}, \tag{29}$$

$$\mathcal{H} = -\frac{\sum_j p_j \log p_j}{\log N_r}, \qquad \Phi = \frac{\exp(\frac{1}{N_r}\sum_j \log A_j)}{\frac{1}{N_r}\sum_j A_j}, \tag{30}$$

$$\text{crest} = \frac{\max_j A_j}{P}, \qquad \text{P2F} = \frac{\max_j A_j}{\min_j A_j}, \tag{31}$$

$$\mu_\rho = \sum_j p_j \rho_j, \qquad s_\rho = \left[\sum_j p_j (\rho_j - \mu_\rho)^2\right]^{1/2}. \tag{32}$$

Signed log-kurtosis is $\operatorname{sign}(\kappa)\log(1+|\kappa|)$ for the excess kurtosis κ of regional amplitudes. Coherence γ is the mean normalized cross-spectrum between a channel and the remaining channels. All ratios use $\epsilon > 0$ for numerical stability.

Each descriptor supplies a different kind of evidence to the shared gate:

- **Log power, $\log P$.** This is the logarithm of the mean energy in one spectral region. It records the absolute activity level while compressing the large dynamic range found in shocks, acoustic waves, and turbulent flows. An increase can therefore indicate that a regional correction has become dynamically significant.
- **Power share, q.** This is the fraction of total low/high energy assigned to the current region. Unlike $\log P$, it is amplitude-relative and exposes transfers between large and small scales, such as the redistribution of vorticity energy during a Navier–Stokes cascade or the generation of fine scales near an Euler shock.
- **Spectral entropy, $\mathcal{H}$.** Entropy measures how uniformly regional energy is distributed across frequency locations. Low entropy indicates concentration in a few modes, whereas high entropy indicates broadband structure. It helps distinguish coherent waves from fragmented interfaces or increasingly multiscale turbulence.
- **Spectral flatness, Φ.** The geometric-to-arithmetic mean ratio separates tone-like spectra from noise-like or broadband spectra. Smooth, coherent PDE states tend to concentrate energy, while mixing, instability, and interface breakup produce flatter regional distributions; the gate can adjust the regional update accordingly.
- **Log crest factor, $\log \mathrm{crest}$.** Crest compares the strongest frequency component with mean regional power. It detects whether one mode or harmonic dominates, which is useful for periodic waves and for localized structures whose Fourier response develops pronounced peaks.
- **Log peak-to-floor ratio, $\log \mathrm{P2F}$.** P2F measures regional spectral dynamic range from the strongest to the weakest component. Large values indicate sparse, high-contrast spectral structure, as can occur around sharp fronts; smaller values indicate a more uniformly populated region.
- **Signed log-kurtosis, $\mathrm{slogKurt}$.** This descriptor records the direction and strength of heavy-tailed regional amplitudes while controlling scale through a signed logarithm. It is sensitive to intermittent events, isolated vortices, and shock-driven outliers that may be understated by average power alone.
- **Spectral centroid, μ_ρ.** The centroid is the energy-weighted mean radial frequency. It tracks the characteristic spatial scale of the current state and reveals systematic movement toward coarser or finer structures during advection, diffusion, or instability growth.
- **Spectral spread, s_ρ.** Spread is the energy-weighted standard deviation around the centroid. It measures bandwidth, distinguishing nearly single-scale dynamics from simultaneous activity over many scales even when both have the same centroid.
- **Cross-channel coherence, γ.** Coherence measures normalized spectral coupling between one physical channel and the others. It exposes coordinated evolution, such as density–velocity–pressure interactions in compressible Euler flow or coupling between velocity components in Navier–Stokes dynamics, and lets the gate respond to relations that no channel-wise energy statistic can capture.

C.2 Inference-time AGC feature masking and spatial evidence

Starting from the same fully trained DRESO-L checkpoint, we mask one normalized AGC statistic at a time during inference without updating any model parameter. Masking is applied after feature standardization and before AGC-logit computation: the selected coordinate is set to zero for both the low- and high-frequency evidence vectors. Because zero is the training-distribution reference value and the neutral point of the low/high comparison, that coordinate contributes no feature-specific signal to either the linear evidence prior or the MLP. Tables 6 and 7 report the resulting relative increase in RelL1. These experiments measure the reliance of the learned routing policy on each statistic; they do not estimate the performance of a model retrained without that statistic. A dagger marks the second-largest increase within each dataset column.

Table 6: One-step inference-time AGC feature masking. From the same trained DRESO-L checkpoint, one normalized statistic is set to zero for both low/high evidence vectors before AGC-logit computation, without retraining. Entries are relative increases in prediction error. Bold and † mark the largest and second-largest increases.

Component	Ablated gate feature	NS-Gauss	CE-RP	CE-CRP	CE-Gauss	CE-KH	NS-Sines
Gate input	Log power	65.510	13.474	7.772	57.697	26.796	37.039
	Power share	66.666	18.221	10.631	66.122	41.020	53.776
	Entropy	**155.702**	48.759†	**50.479**	**188.322**	79.660	126.048†
	Flatness	153.690	36.233	46.978†	179.429†	79.961†	114.036
	Log crest	11.391	5.641	0.682	10.301	7.907	4.248
	Log peak-to-floor	10.413	6.090	1.655	17.813	7.061	7.467
	Signed log kurtosis	17.285	4.275	0.856	11.326	6.732	4.289
	Centroid	109.205	23.035	14.343	81.128	47.404	43.701
	Spread	65.141	12.253	5.842	47.340	25.121	9.364
	Coherence	154.862†	**87.400**	25.944	168.739	**97.949**	**149.149**

For one-step prediction, the learned routing policy is most sensitive on average to coherence, entropy, and flatness: masking them increases RelL1 by 114.007%, 108.162%, and 101.721%, respectively. Their largest individual increases are 168.739%, 188.322%, and 179.429%. Centroid masking raises error by 53.136% on average. Masking log power, power share, or spread also increases error on every task, showing that the trained policy uses their energy and bandwidth evidence consistently. Log crest, log peak-to-floor, and signed log kurtosis produce smaller but uniformly positive changes; the checkpoint therefore relies on their peak, dynamic-range, and tail-shape information even though its sensitivity is lower.

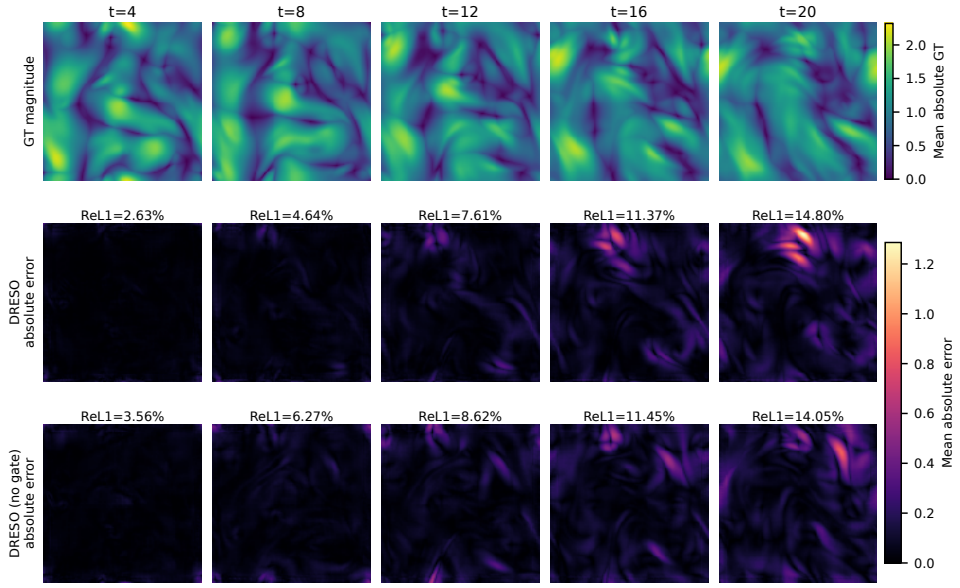
Table 7: Rollout inference-time AGC feature masking. From the same trained DRESO-L checkpoint, one normalized statistic is set to zero for both low/high evidence vectors before AGC-logit computation, without retraining. Entries are relative increases in prediction error. Bold and † mark the largest and second-largest increases.

Component	Ablated gate feature	NS-Gauss	CE-RP	CE-CRP	CE-Gauss	CE-KH	NS-Sines
Gate input	Log power	12.163	55.432	78.406	112.197	95.691	6.823
	Power share	-1.661	48.889	45.442	104.358	86.238	3.413
	Entropy	9.431	118.295†	173.707†	**250.202**	**454.151**	**48.142**
	Flatness	7.700	105.276	**176.887**	242.440†	414.943†	37.079†
	Log crest	4.291	26.916	11.679	30.097	40.776	8.260
	Log peak-to-floor	2.268	26.236	21.642	42.179	33.537	6.449
	Signed log kurtosis	5.614	24.290	16.755	28.981	32.335	5.082
	Centroid	**27.182**	100.370	136.116	171.356	270.426	16.682
	Spread	15.978†	54.893	81.580	100.539	107.590	14.364
	Coherence	11.023	**318.092**	75.806	165.177	294.808	22.593

During rollout, the trained policy relies most strongly on entropy: masking it increases mean RelL1 by 175.655% and reaches 454.151% on CE-KH. Flatness and coherence follow with mean increases of 164.054% and 147.917%, while centroid masking produces a 120.355% mean increase. Spread and log power remain positive on every task. Power-share masking increases error on five rollouts but decreases NS-Gauss error by 1.661%, so the checkpoint's use of this statistic is not uniformly beneficial. The remaining statistics produce smaller but consistently positive changes. These results establish sensitivity of this trained routing policy under the evaluated tasks.

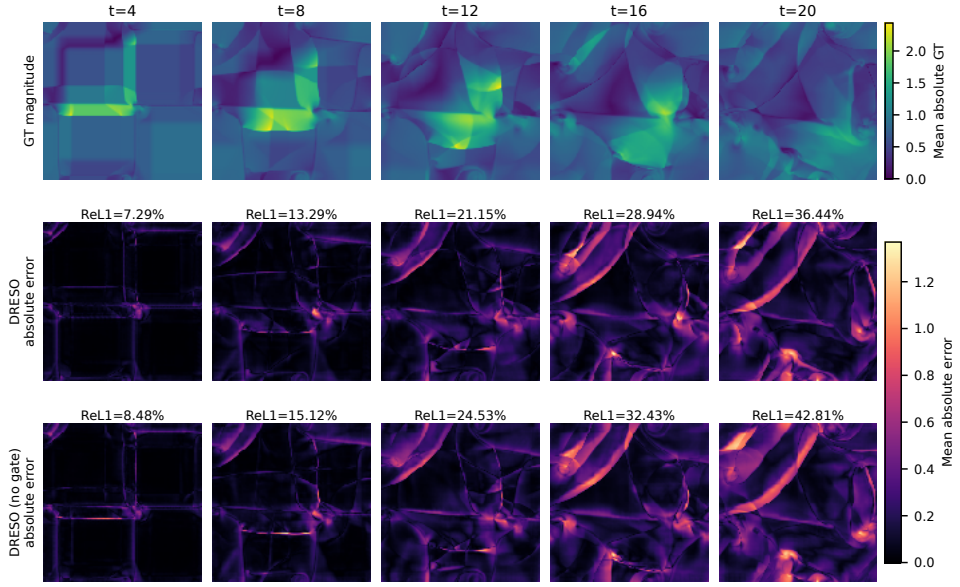
Full-field spatial evidence. We additionally compare the complete model with its no-AGC counterpart under the same 20-step autoregressive protocol. Figure 5 displays snapshots at $t \in \{4, 8, 12, 16, 20\}$ for all six CamLab-ETHZ systems. The first row is the ground-truth mean absolute field magnitude, and the second and third rows are absolute errors for DRESO and DRESO without AGC, respectively. Errors are computed in the native normalized space; the Navier–Stokes systems use the velocity channels, whereas the compressible-Euler systems use all four physical

channels. Across the six datasets, the visualizations expose how AGC affects the spatial accumulation of error through the rollout horizon, complementing the controlled AGC ablation in Table 5.



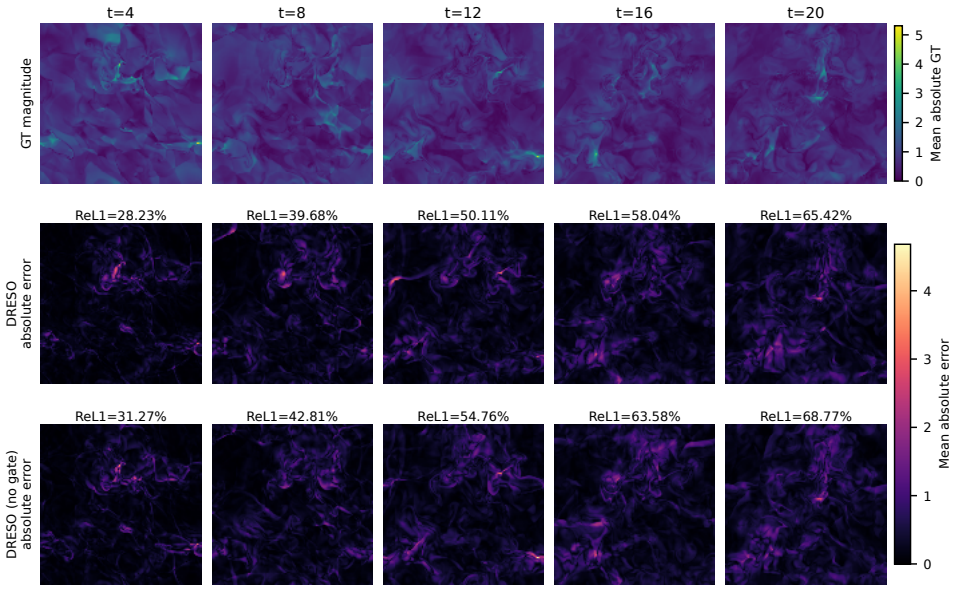
(a) NS-Gauss

Figure 5: Full-field gate ablation on NS-Gauss. Columns show rollout steps $t = 4, 8, 12, 16, 20$; rows show ground-truth mean absolute magnitude, DRESO absolute error, and no-gate absolute error.



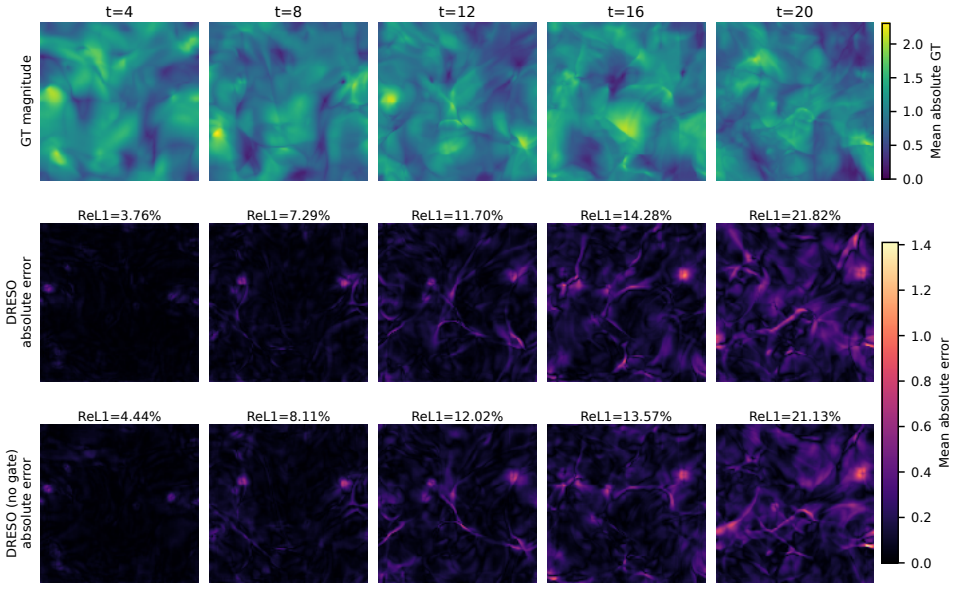
(b) CE-RP

Figure 5: Full-field gate ablation on CE-RP (continued). Columns and rows follow the organization in panel (a).



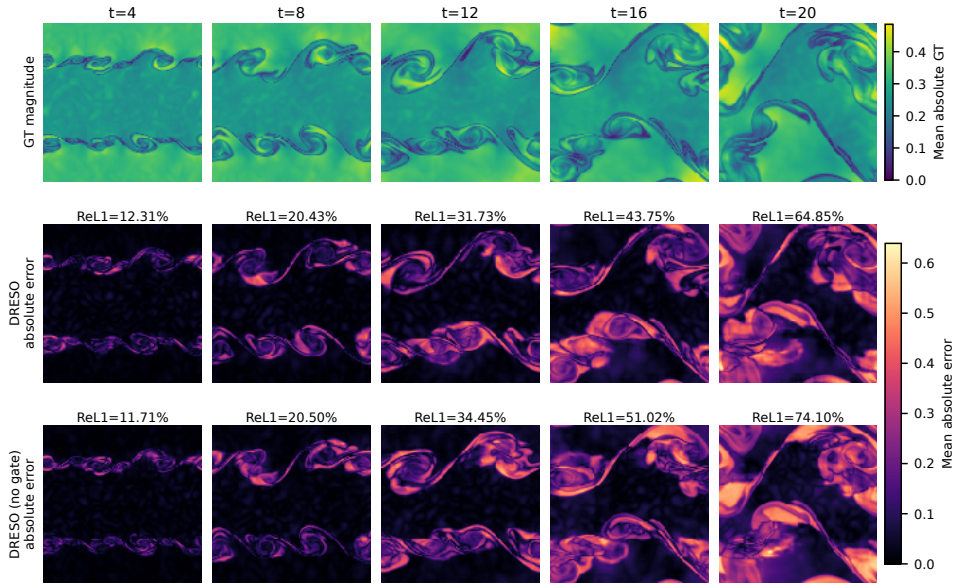
(c) CE-CRP

Figure 5: Full-field gate ablation on CE-CRP (continued). Columns and rows follow the organization in panel (a).



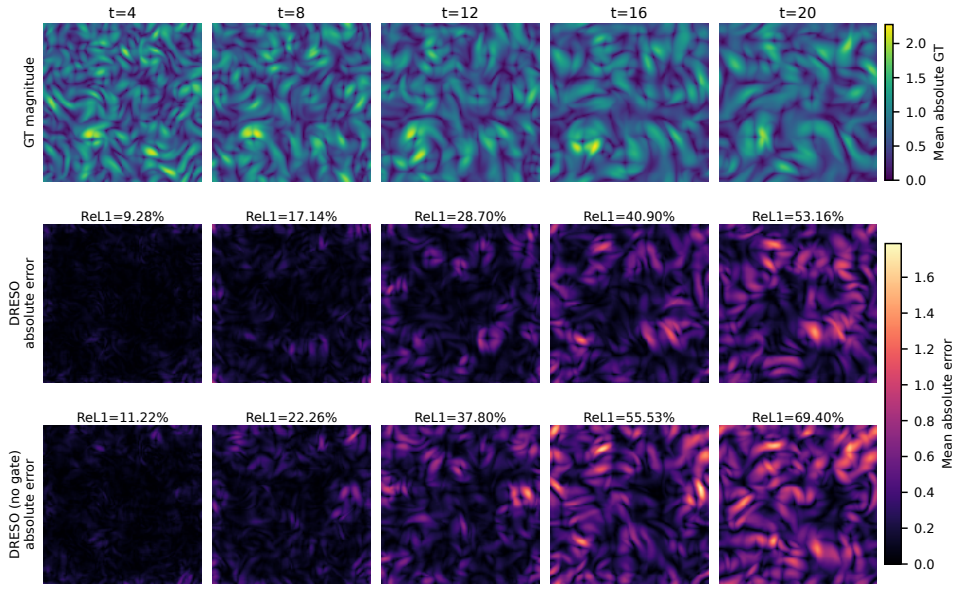
(d) CE-Gauss

Figure 5: Full-field gate ablation on CE-Gauss (continued). Columns and rows follow the organization in panel (a).



(e) CE-KH

Figure 5: Full-field gate ablation on CE-KH (continued). Columns and rows follow the organization in panel (a).



(f) NS-Sines

Figure 5: Full-field gate ablation on NS-Sines (continued). Columns and rows follow the organization in panel (a).

C.3 Gate dynamics during rollout

We record all per-sample, per-channel low- and high-frequency gains from every spectral block over 200 test trajectories and all 20 autoregressive steps. The aggregate view in Figure 6 and the dataset-resolved views in Figure 7 show piecewise, stage-like changes on most datasets rather than a single time-invariant allocation. These changes remain within the implemented range $[0.25, 1.75]$. Although the median low-frequency gain is often above one, the complementary high-frequency gain remains materially nonzero; it is therefore inaccurate to interpret the gate as assigning overwhelming

weight to the low-frequency route while deactivating the high-frequency route. Moreover, the pooled blockwise populations form several separated bands and, on some datasets, occupy a broad portion of the admissible interval. Thus the gates from different blocks do not all collapse to one narrow coefficient range.

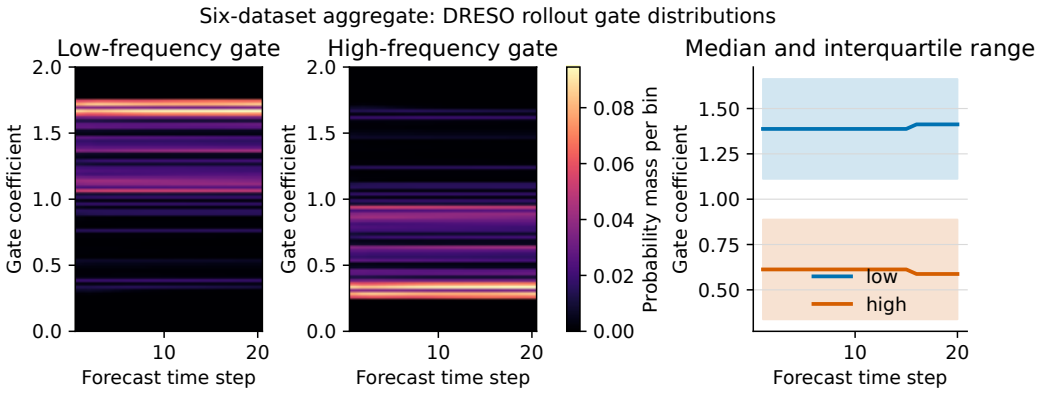
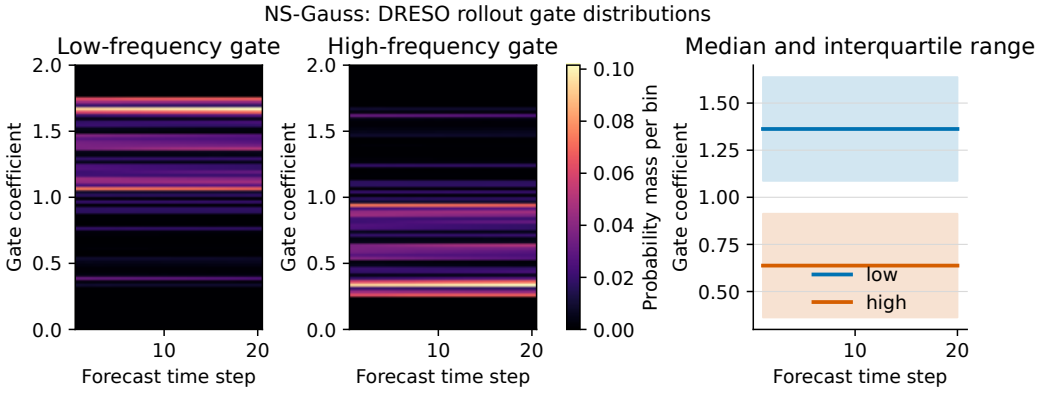
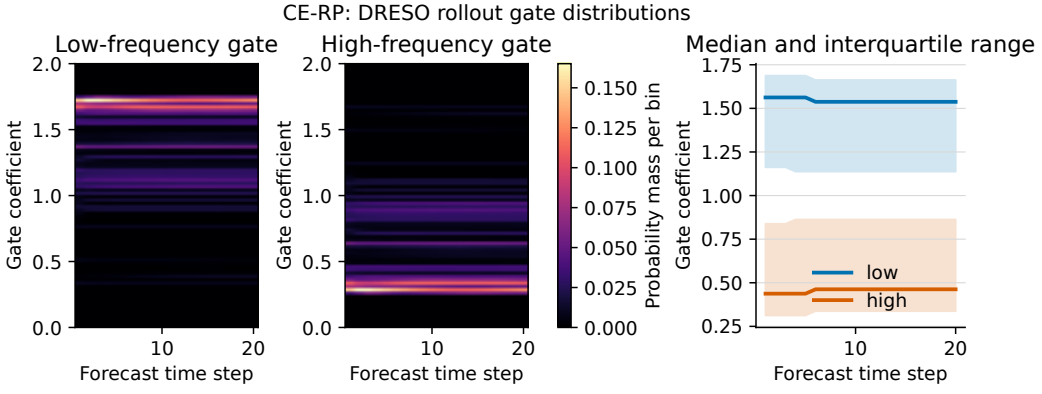


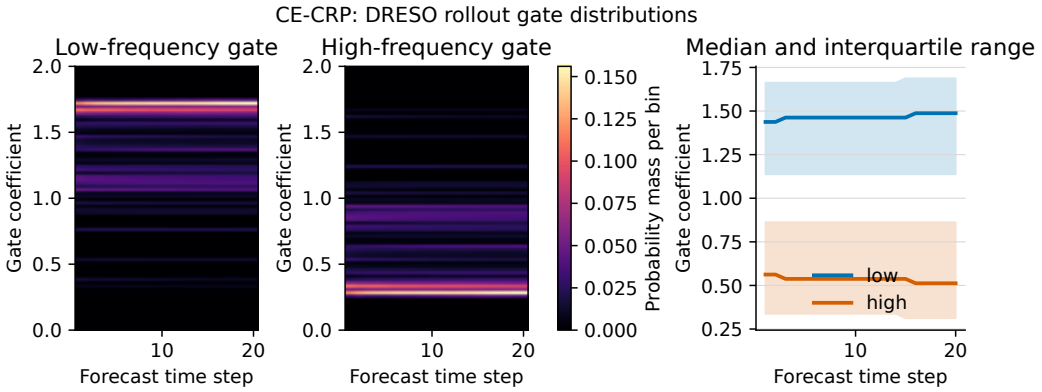
Figure 6: Aggregate gate dynamics over all six rollout datasets. Left and center panels show the distributions of low- and high-frequency gains over forecast time; the right panel shows their medians and interquartile ranges. The distributions pool all samples, channels, and spectral blocks.



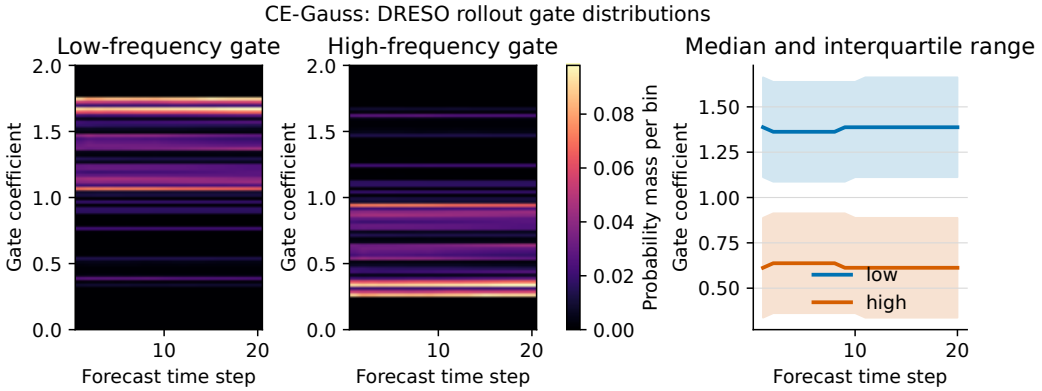
(a) NS-Gauss



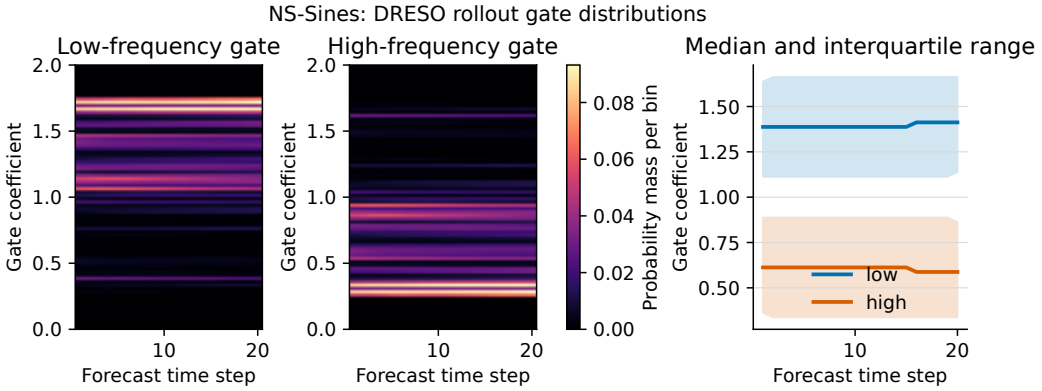
(b) CE-RP



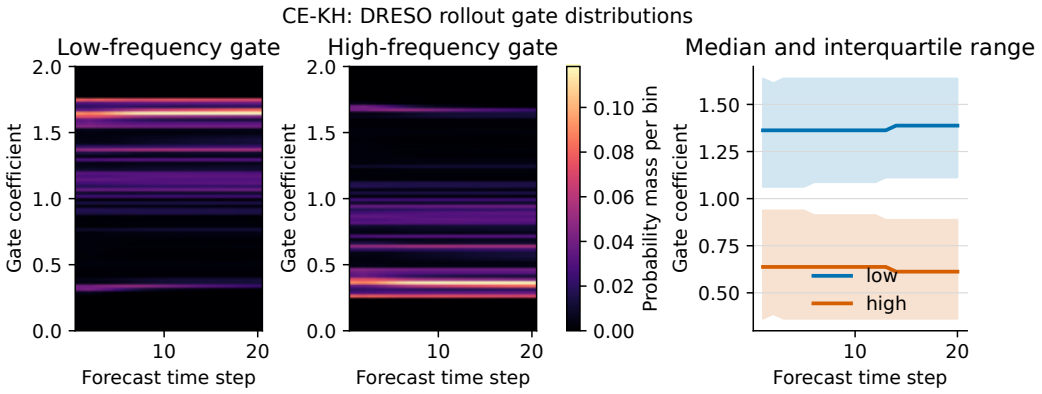
(c) CE-CRP



(d) CE-Gauss



(e) NS-Sines



(f) CE-KH

Figure 7: Gate-coefficient distributions and temporal summaries across the six datasets.

C.4 Learnable Gaussian cutoff analysis

Figure 8 shows that the learned parameters deviate from the 0.25 initialization. Encoder cutoffs remain near 0.242–0.259, while the final decoder blocks increase to about 0.29. This suggests a relatively balanced split in early representations, followed by greater low-frequency allocation during reconstruction without suppressing the complementary high-frequency path. Although numerically modest, these shifts affect entire radial frequency ranges rather than individual Fourier modes. Appendix B.3 derives this sensitivity.

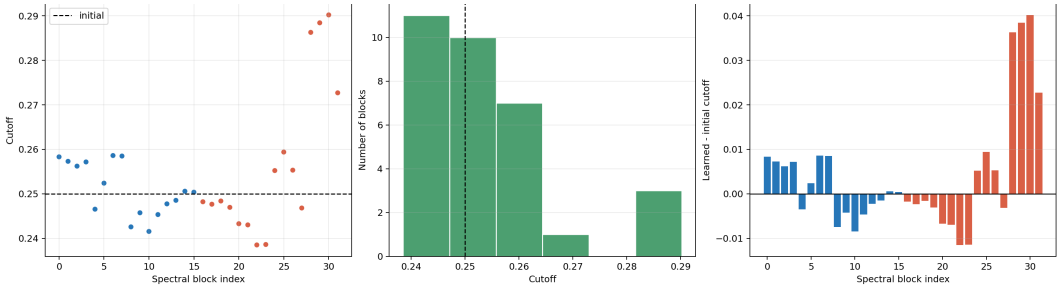


Figure 8: Learned normalized cutoffs by spectral block. Encoder blocks (blue) make moderate bidirectional adjustments, while late decoder blocks (red) move toward larger cutoffs. The apparently narrow range is significant because each displacement changes the response of a two-dimensional frequency annulus; see Appendix B.3.

C.5 Filter analysis

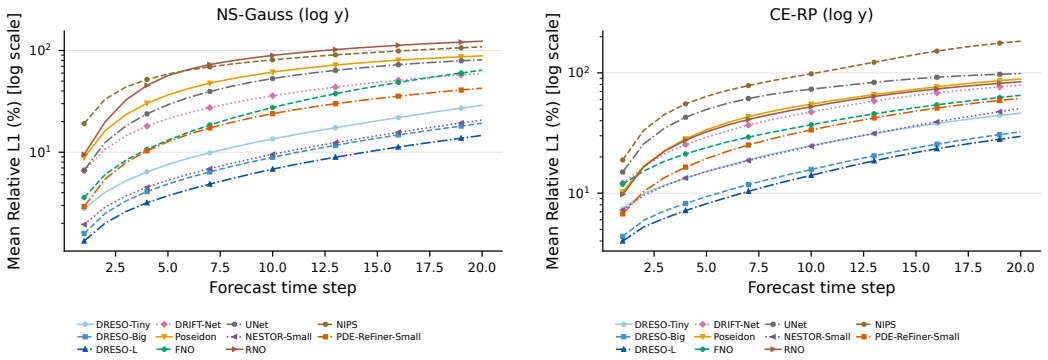
Table 8: Controlled comparison of fixed ideal and Gaussian filters.

Filter	NS-Gauss		CE-RP		CE-CRP		CE-Gauss		CE-KH		NS-Sines	
	One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout
Ideal filter	1.735	11.125	**4.317**	19.788	18.920	44.075	2.594	16.320	6.294	31.027	4.437	37.028
Fixed Gaussian filter	**1.555**	**9.081**	4.432	**17.649**	**17.786**	**40.540**	**2.208**	**13.347**	**6.143**	**30.195**	**4.008**	**27.882**

With both cutoffs fixed, the Gaussian response improves 11 of the 12 metrics over the ideal filter. The only exception is CE-RP one-step prediction, where the ideal filter is lower (4.317 versus 4.432). This broad advantage isolates the contribution of the smooth response from cutoff learnability, while the learned cutoff supplies an additional degree of freedom.

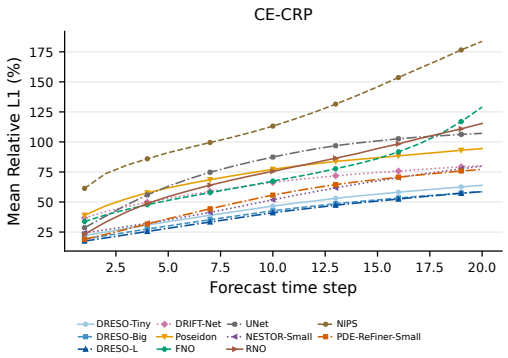
D ROLLOUT ERROR GROWTH

Figure 9 reports the complete dataset-specific error–time curves for the 20-step rollout. Each curve is the mean RelL1 (%) in the native normalized space, and panel titles identify logarithmic vertical axes. Across the six systems, DRESO generally shows lower error growth over the rollout horizon and exhibits the lowest or near-lowest growth trajectory; the contrast is especially pronounced on NS-Gauss and CE-Gauss, complementing the aggregate rollout results in Table 3.

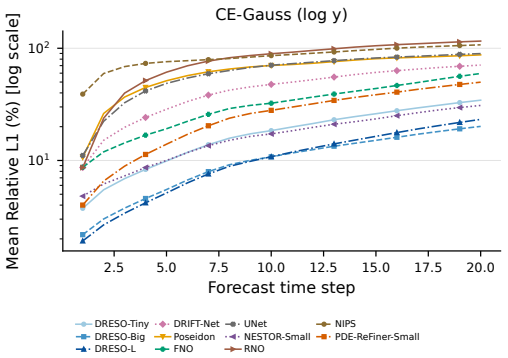


(a) NS-Gauss

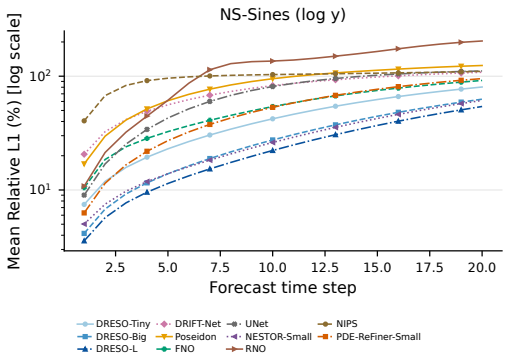
(b) CE-RP



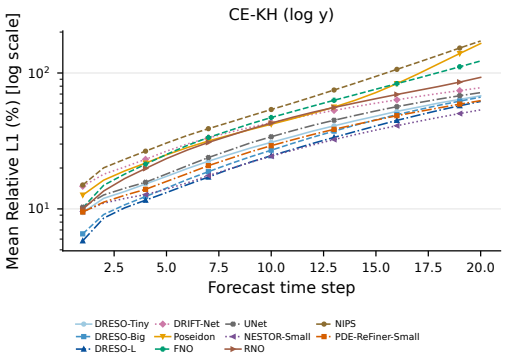
(c) CE-CRP



(d) CE-Gauss



(e) NS-Sines



(f) CE-KH

Figure 9: Rollout error growth across the six datasets over the 20-step autoregressive rollout.

E ADDITIONAL EXPERIMENTS

E.1 ZERO-SHOT TRANSFER ON UNSEEN SYSTEMS

Table 9: Zero-shot transfer to five unseen systems, evaluated using both one-step and rollout errors. DRESO-L is best on seven of ten metrics. Bold marks the best result; † marks the runner-up.

Model	CE-RM		NS-BB		NS-SL		CE-RPUI		Wave-Gauss	
	One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout
UNet	**99.974**	146.229	15.333	114.440	6.730	60.609	21.796	112.584	41.119†	190.409
FNO	146.971	279.153	**4.150**	95.687	6.897	66.195	24.116	118.989	64.215	137.105†
Small-NESTOR	117.438	127.682†	32.036	83.288†	**4.778**	32.707†	48.304	100.531	49.455	147.535
DRIFT-Net	114.065	142.321	29.476	118.770	9.588	37.866	18.948†	81.921†	52.422	194.646
DRESO-L	105.199†	**110.820**	10.668†	**81.958**	6.711†	**22.381**	**18.088**	**73.772**	**38.431**	**122.573**

Table 9 evaluates transfer without target-task adaptation. DRESO-L leads all five rollout comparisons and two of five one-step comparisons. UNet, FNO, and NESTOR-Small lead the one-step CE-RM, NS-BB, and NS-SL cases, respectively, but none retains that advantage over the trajectory. This contrast separates immediate fitting from closed-loop robustness under distribution shift.

E.2 COMPLETE CAMLAB-ETHZ AUTOREGRESSIVE ROLLOUT RESULTS

Table 10: Autoregressive rollout on CamLab-ETHZ tasks under pretraining, finetuning for 50 epochs, and finetuning for 100 epochs. Bold marks the best result; † marks the runner-up.

Protocol	Model	NS-Gauss	CE-RP	CE-CRP	CE-Gauss	CE-KH	NS-Sines
Pretrain	NESTOR-Small	10.512	27.041	52.840	17.803	**27.794**	29.968†
	UNet	50.682	69.374	81.735	64.582	37.652	73.991
	FNO	30.956	38.814	72.663	33.589	55.331	55.108
	PDE-ReFiner-Small	32.561	45.235	60.978	41.855	37.114	59.656
	Poseidon-T	50.951	58.364	96.351	65.495	54.176	81.771
	DRIFT-Net	31.753	43.611	60.038	38.161	49.587	75.499
	NIPs	77.152	104.961	120.785	85.352	70.035	97.710
	RNO	88.619	52.447	75.293	89.365	46.927	123.765
	DRESO-L	**7.485**	**15.714**	**40.372**	**11.059**	29.152†	**25.785**
	DRESO-B	9.811†	17.380†	41.722†	11.673†	31.899	30.960
	DRESO-T	14.791	26.442	45.825	19.097	34.935	44.428
Finetune-50	UNet	41.621	54.701	73.361	63.609	32.166	56.167
	NESTOR-Small	8.983†	16.581†	44.871†	14.096†	**23.965**	26.014†
	DRIFT-Net	29.081	38.094	58.741	37.948	43.591	70.425
	DRESO-L	**4.849**	**12.811**	**40.815**	**8.106**	27.915†	**20.012**
Finetune-100	UNet	39.495	50.301	69.041	60.544	96.473	54.947
	NESTOR-Small	8.041†	15.018†	43.530†	12.936†	**23.325**	24.288†
	DRIFT-Net	28.311	37.471	59.053	37.070	43.112	67.642
	DRESO-L	**5.224**	**12.164**	**38.921**	**8.083**	26.444†	**20.041**

Table 10 extends the pretraining-only presentation in Table 3 to all three training protocols. DRESO-L ranks first on five of six datasets in every protocol, while NESTOR-Small achieves the lowest CE-KH rollout error.

E.3 THE WELL AUTOREGRESSIVE ROLLOUT

Table 11: Autoregressive results for models trained from scratch on The Well's four native-resolution datasets and evaluated over 20-state sequences. All entries are RelL1 errors, for which lower is better. Bold marks the best result; † marks the runner-up.

Model	Acoustic Scattering	Active Matter	Gray-Scott	Planetary Weather
NESTOR-Small	25.501	**100.363**	104.966	13.040
UNet	28.459	134.342	103.883	14.798
FNO	26.989	136.738	102.886	8.314
PDE-ReFiner-Small	25.991	138.962	95.967	13.041
Poseidon-T	25.363	128.493	**81.594**	13.033
DRIFT-Net	25.154	140.893	88.437†	12.261
NIPs	161.797	157.511	131.182	90.001
DRESO-L	**6.102**	123.636†	92.702	**4.413**
DRESO-B	6.414†	126.662	94.106	4.705†
DRESO-T	9.533	128.717	94.939	5.549

Table 11 shows that the one-step ranking in Table 2 does not consistently extend to closed-loop prediction. The one-step task measures prediction from a ground-truth state. During rollout, predicted states become subsequent inputs, so small phase, amplitude, or structural errors can alter the input distribution and compound over time. DRESO-L ranks first on acoustic scattering and planetary weather. NESTOR-Small ranks first on active matter, and Poseidon-T ranks first on Gray–Scott. These results distinguish one-step accuracy from autoregressive rollout performance.

Table 12: One-step and long-horizon results on The Well Shear Flow. Rollouts are evaluated at 100 and 199 autoregressive steps. All entries are RelL1 errors; lower is better. Bold marks the best result; † marks the runner-up.

Dataset	Rollout steps	NESTOR-Small	DRIFT-Net	DRESO-L	DRESO-Big	DRESO-Tiny
Shear Flow	100	69.773†	187.862	165.548	101.049	**63.141**
	199	**94.525**	194.283	204.316	150.638	110.150†
	One step	13.418	10.865	**8.719**	9.090†	9.714

Table 12 extends the comparison from one-step prediction to long-horizon autoregressive evaluation. All three DRESO variants outperform DRIFT-Net at 100 steps, while DRESO-Big and DRESO-Tiny retain the advantage at 199 steps. Across all methods, DRESO-Tiny ranks first at 100 steps and second at 199 steps; NESTOR-Small ranks first at 199 steps. Relative to DRIFT-Net, DRESO-Tiny reduces RelL1 from 187.862 to 63.141 at 100 steps and from 194.283 to 110.150 at 199 steps. One plausible explanation is greater closed-loop stability in the smaller variants. They may be less sensitive to prediction errors at each step, leading to slower error amplification over long rollouts. Additional capacity does not necessarily improve this stability.

E.4 INFERENCE EFFICIENCY

Table 13: Unified inference-efficiency benchmark using a simulated 128×128 input. Every model is evaluated in FP32 with batch size 1, 20 warmup iterations, and 100 timed iterations. Latency, throughput, and peak inference memory use the same measurement protocol.

Setting	Model	Total Params (M)	Latency (ms)	Throughput (samples/s)	Peak VRAM (MiB)
FP32, batch 1	NESTOR	80.760	12.960	77.140	449.140
	NESTOR-Small	20.510	14.666	67.886	218.959
	FNO	0.470	1.630	613.590	30.870
	UNet	7.760	1.730	580.090	165.780
	PDE-ReFiner	146.510	49.102	20.306	724.239
	PDE-ReFiner-Small	36.640	30.220	33.009	291.204
	DRIFT-Net	17.100	38.660	25.870	87.440
	Poseidon-T	20.770	41.460	24.120	108.480
	RNO	2.840	18.020	55.610	289.390
	NIPs	0.055	1.376	726.360	37.090
	DRESO-Large	22.150	158.150	6.320	100.690
	DRESO-Big	11.480	82.770	12.080	58.240
	DRESO-Tiny	2.900	79.990	12.450	22.430

Table 13 shows that DRESO does not lead in latency or throughput: DRESO-Large requires 158.15 ms per sample, whereas several compared baselines are faster. DRESO-L ranks first on five of six Poseidon rollout tasks in Table 3 and all four The Well one-step tasks in Table 2. On The Well rollouts, it remains competitive and ranks first on acoustic scattering and planetary weather in Table 11. DRESO-Tiny uses 2.90M parameters and 22.43 MiB peak VRAM, the lowest memory footprint in the efficiency table, while retaining competitive accuracy. It therefore provides an accuracy–memory trade-off for edge settings where parameter storage and inference memory are more constrained than latency.

E.5 DEFAULT TEN-FRAME INPUT COMPARISON

Table 14: One-step performance after restoring the default ten-frame inputs of the multi-frame baselines. DRESO rows retain our proposed single-state interface as a reference. All entries are RelL1 errors; lower is better. Bold marks the best result; † marks the runner-up.

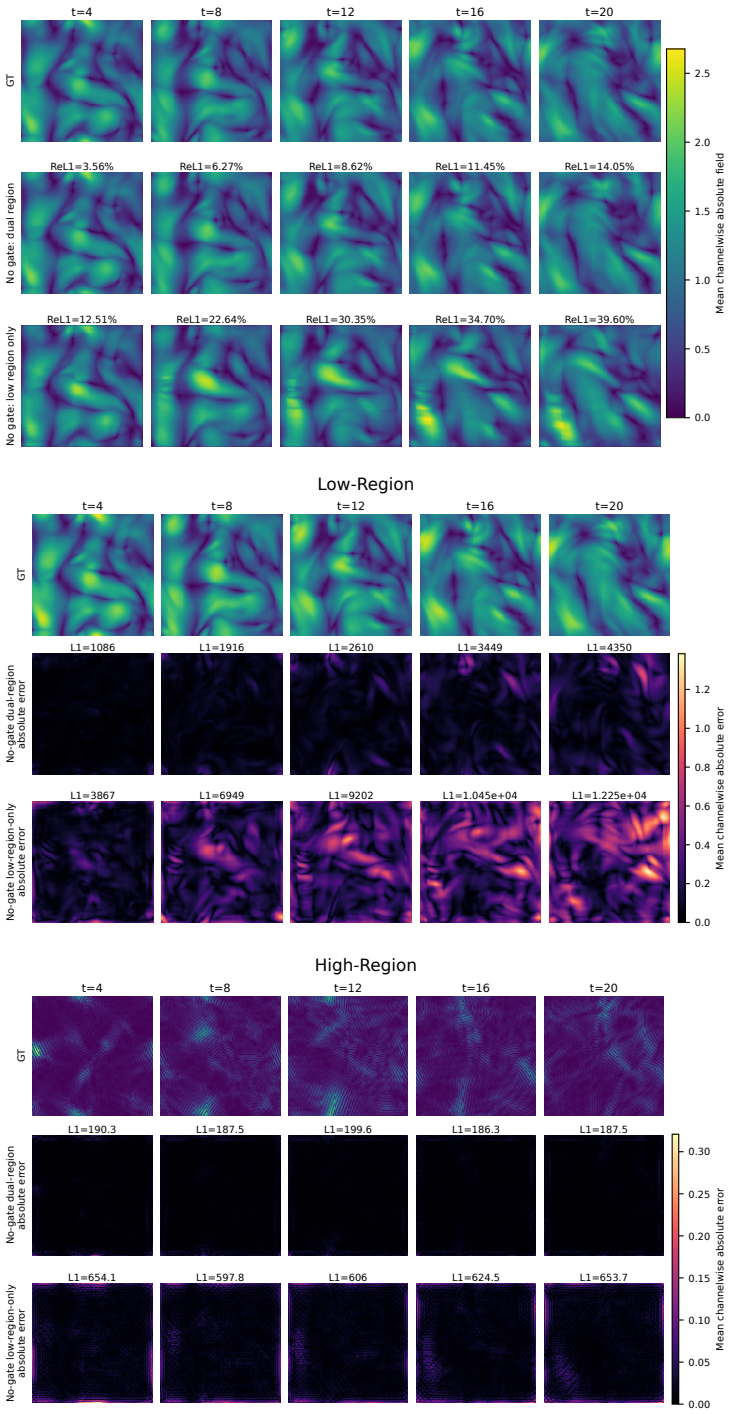
Model	NS-Gauss	CE-RP	CE-CRP	CE-Gauss	CE-KH	NS-Sines
NESTOR-Small	1.444†	5.245	**16.862**	4.441	8.834	**3.369**
FNO	2.354	8.849	26.083	7.919	15.145	6.019
UNet	2.391	5.317	16.993†	7.393	10.694	3.528†
DRESO-L	**1.337**	**3.997**	17.498	**1.917**	**5.823**	3.571
DRESO-B	1.586	4.370†	19.233	2.181†	6.556†	4.151
DRESO-T	2.785	7.403	22.179	3.745	7.462	9.404

Table 14 complements our controlled single-state comparison in Table 1 by restoring ten consecutive observations for the retained history-dependent baselines, including NESTOR, FNO, and UNet. Under this protocol, NESTOR-Small leads CE-CRP and NS-Sines, while DRESO-L, which retains the proposed single-state interface as a reference, leads NS-Gauss, CE-RP, CE-Gauss, and CE-KH. The table therefore separates the default-history setting of the retained baselines from the equal single-state observation budget used in the main comparison.

E.6 NO-GATE REGIONAL ROLLOUT VISUALIZATIONS

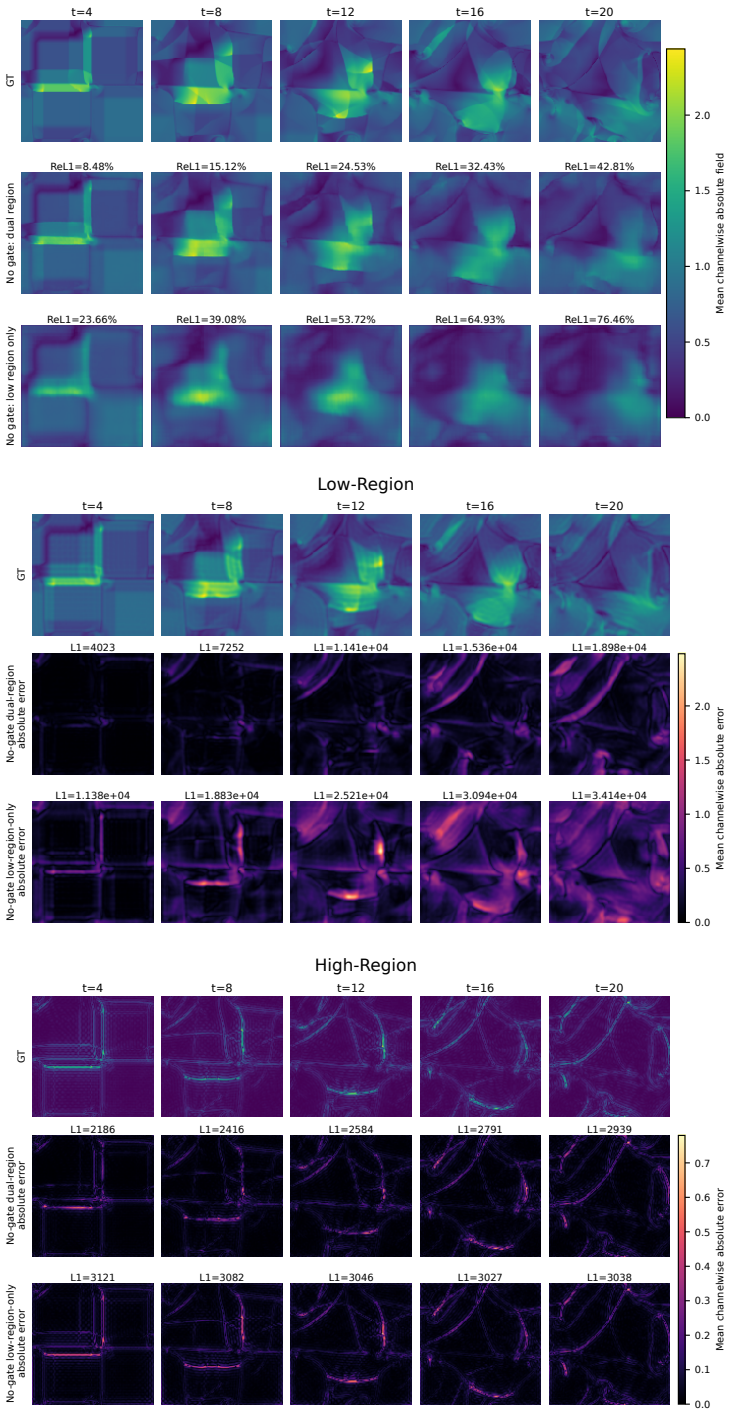
The following trajectory visualizations compare the same no-gate DRESO-L architecture trained with either both complementary regions or the low-frequency region alone. Each dataset panel presents the full field, low-frequency region, and high-frequency region in that order. Full-field annotations report RelL1, consistent with the main tables. The regional panels report absolute L_1 error on the same trajectory. The filtered reference can have a small magnitude, especially in the high-frequency region, making the denominator of a relative error numerically unstable.

Across all six visualized systems, the dual-region variant reduces the full-field error at every displayed rollout step. Its low- and high-frequency regional errors are also generally lower, although the advantage weakens or reverses for a small number of regional comparisons at later steps. This task- and time-dependent behavior motivates dynamic gating. The gate adjusts regional contributions using the current sample and channel. Its controlled benefit is measured separately in Table 5.



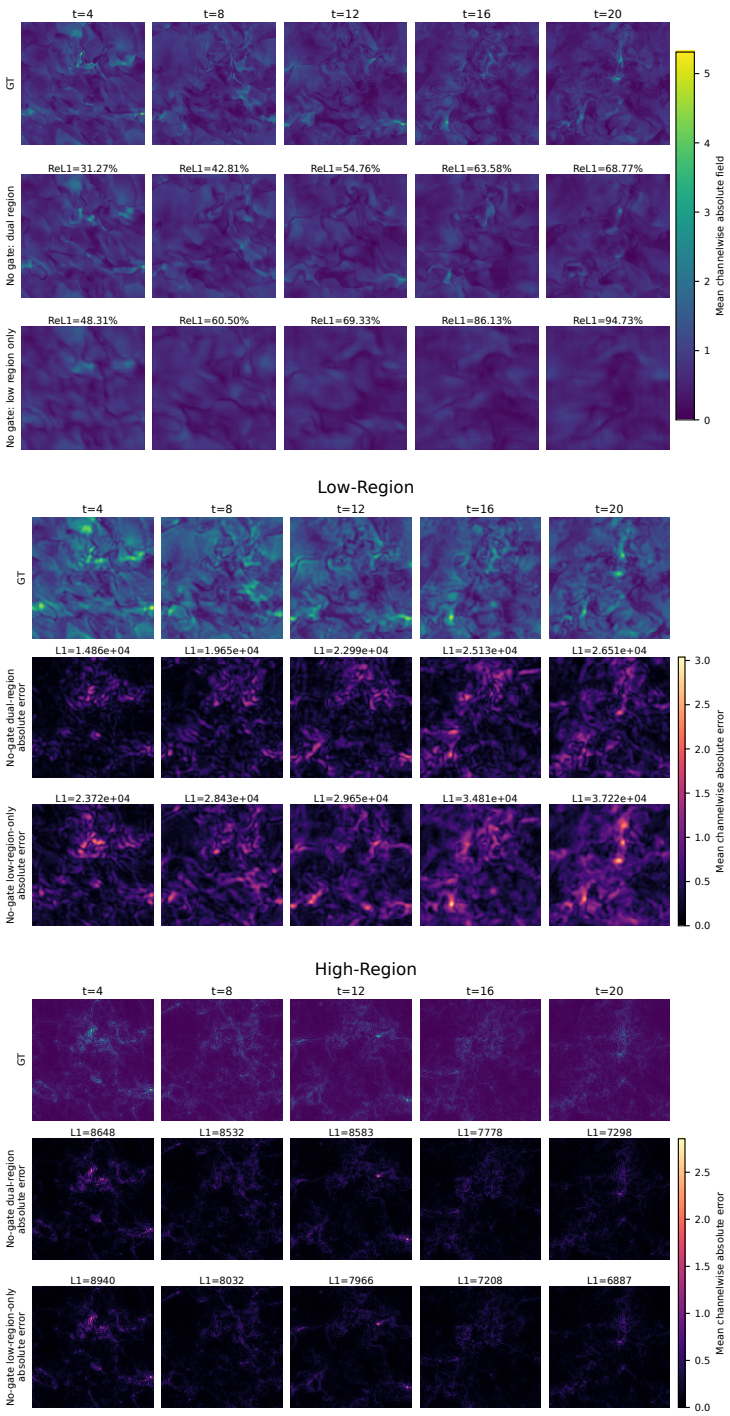
(a) NS-Gauss

Figure 10: No-gate regional rollout visualizations. Within each dataset panel, rows are full field, low-frequency region, and high-frequency region.



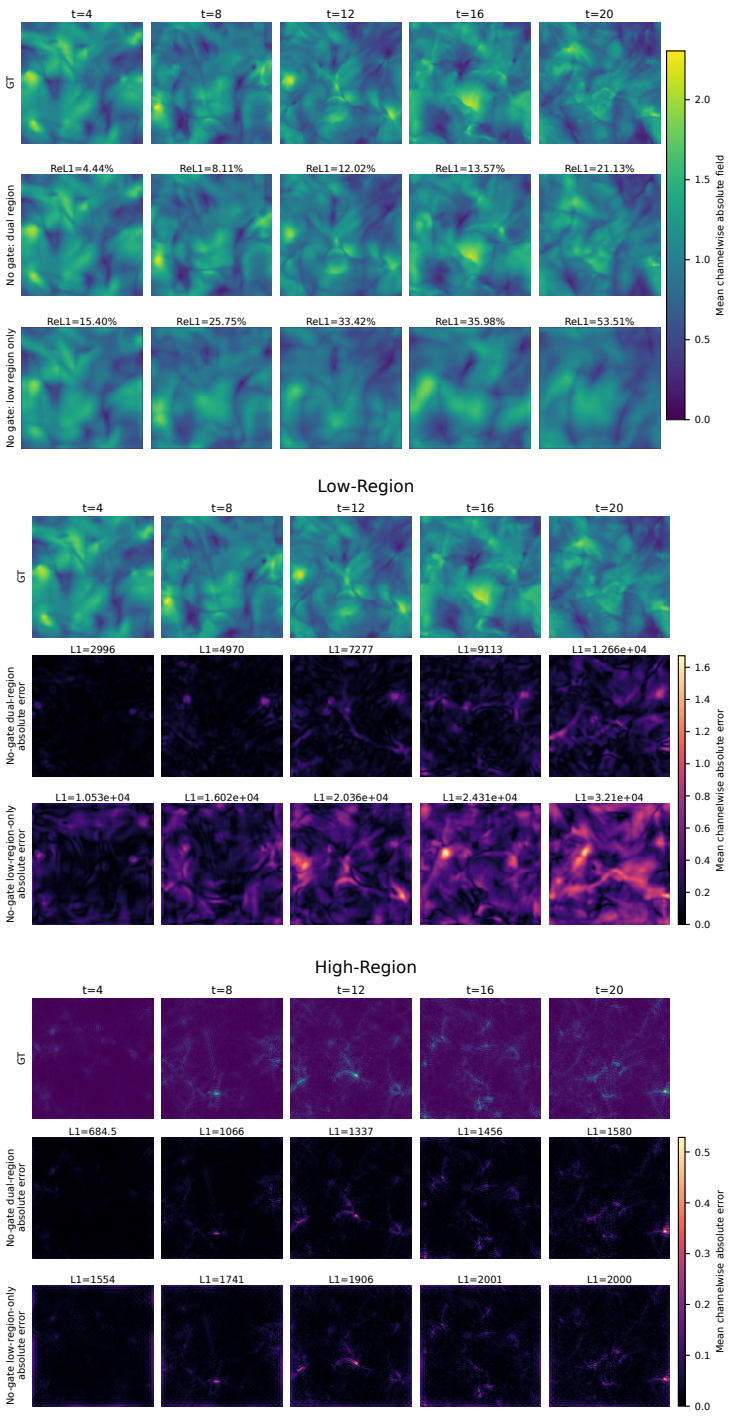
(b) CE-RP

Figure 10: No-gate regional rollout visualizations (continued).



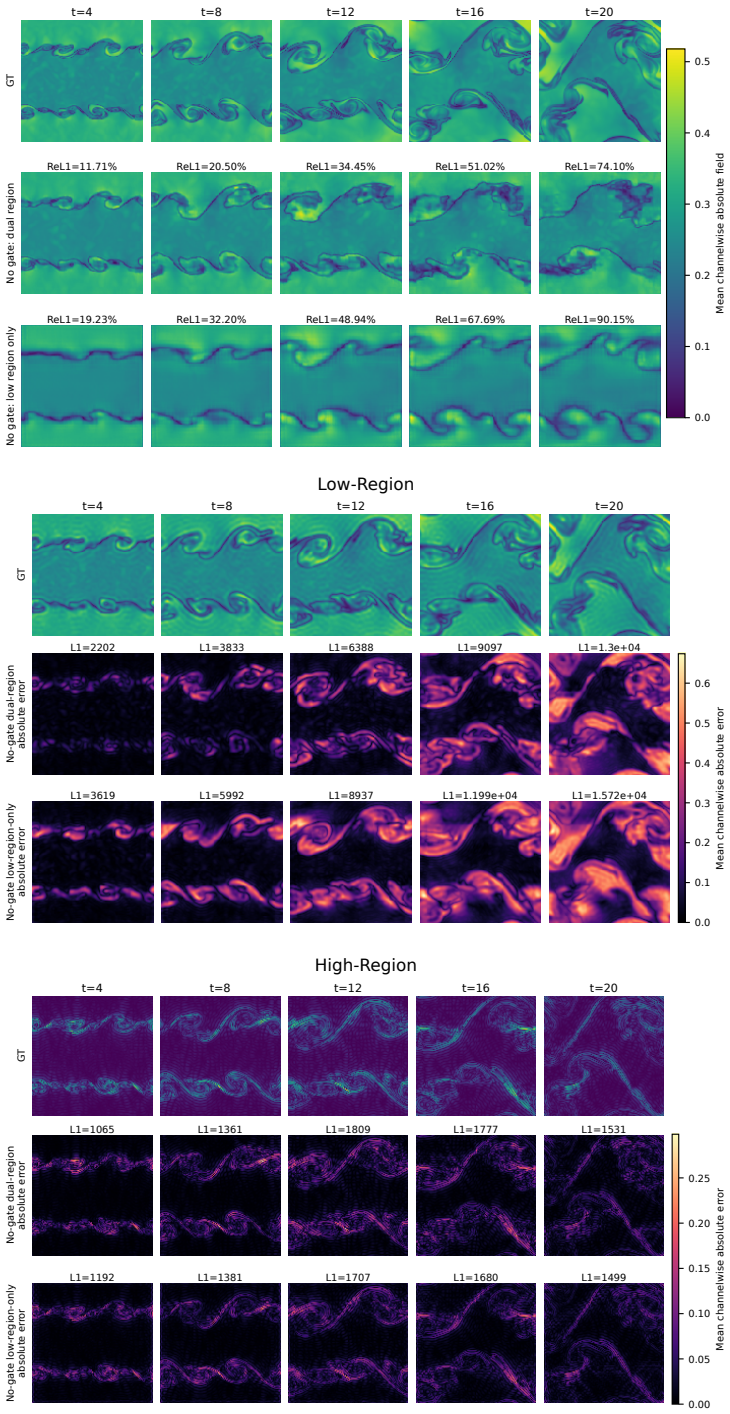
(c) CE-CRP

Figure 10: No-gate regional rollout visualizations (continued).



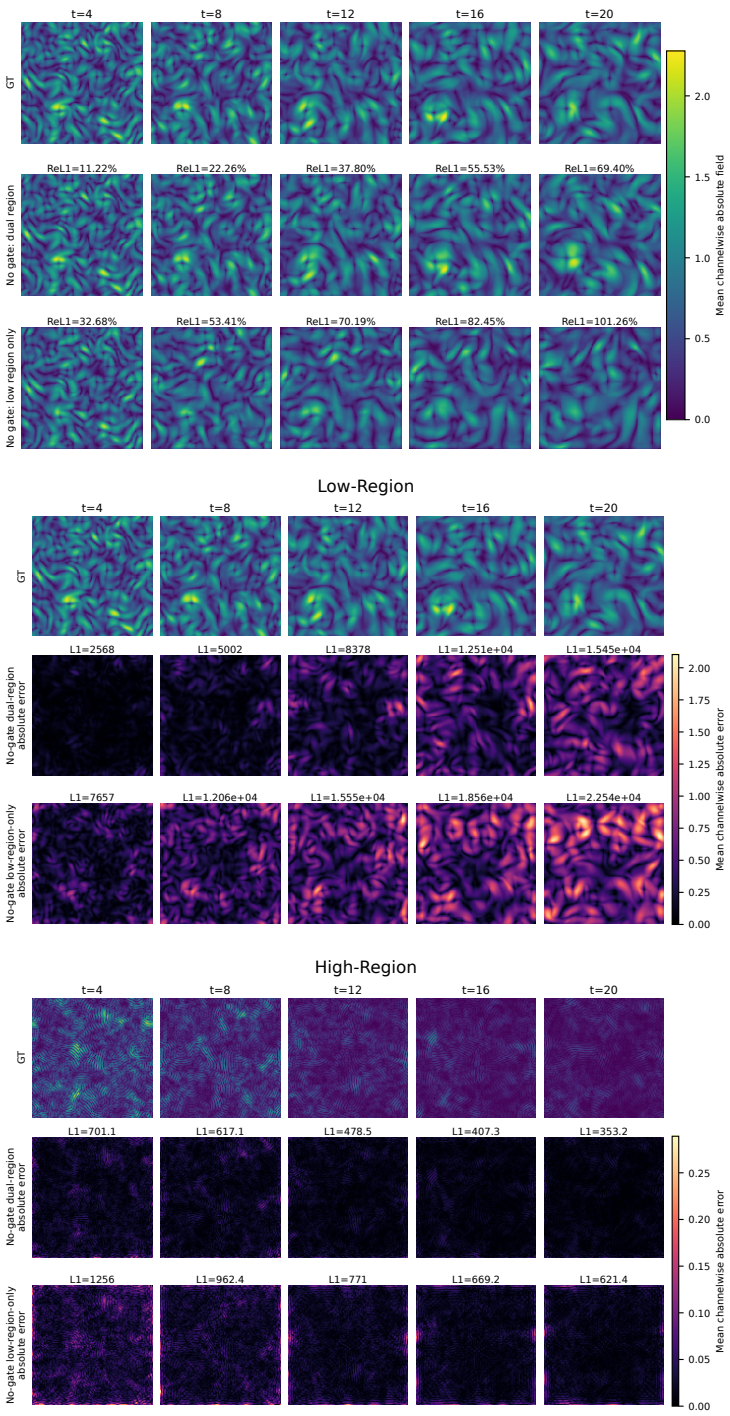
(d) CE-Gauss

Figure 10: No-gate regional rollout visualizations (continued).



(e) CE-KH

Figure 10: No-gate regional rollout visualizations (continued).



(f) NS-Sines

Figure 10: No-gate regional rollout visualizations (continued).

E.7 CROSS-RESOLUTION ZERO-SHOT EVALUATION

Table 15: Cross-resolution transfer across the six CamLab-ETHZ systems. Models are trained at 128×128 and evaluated without parameter updates at 128×128, 96×96, and 64×64. Entries are RelL1 errors; lower is better. Bold marks the best result; † marks the runner-up.

Resolution	Model	NS-Gauss		CE-RP		CE-CRP		CE-Gauss		CE-KH		NS-Sines	
		One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout
128x128	FNO	2.744	30.956	11.871	38.814	30.940	72.663	6.684	33.589	9.391	55.331	8.625	55.108
	UNet	6.643	50.682	14.970	69.374	28.621	81.735	11.074	64.582	10.606	37.652	9.011	73.991
	PDE-ReFiner-Small	3.766	32.561	9.432	45.235	24.168	60.978	5.943	41.855	11.715	37.114	7.830	59.656
	DRIFT-Net	7.140	31.753	12.011	43.611	35.051	60.038	8.229	38.161	15.007	49.587	20.789	75.499
	DRESO-L	**1.337**	**7.485**	**3.997**	**15.714**	**17.498**	**40.372**	**1.917**	**11.059**	**5.823**	**29.152**	**3.571**	**25.785**
	DRESO-B	1.586†	9.811†	4.370†	17.380†	19.233†	41.722†	2.181†	11.673†	6.556†	31.899†	4.151†	30.960†
	DRESO-T	2.785	14.791	7.403	26.442	22.179	45.825	3.745	19.097	7.462	34.935	9.404	44.428
96x96	FNO	**3.027**	41.894	9.729	**40.457**	**27.115**	**68.548**	**6.606**	**36.310**	**9.389**	57.176	**10.758**	**49.366**
	UNet	8.794	68.482	17.120	74.211	38.382†	113.234	17.909	76.026	16.847	94.891	16.583	86.702
	PDE-ReFiner-Small	7.995	54.856	12.451	61.532	42.615	83.161	18.759	64.480	11.858	55.315	17.812	80.221
	DRIFT-Net	11.386	57.333	15.507	51.926	44.956	78.437	21.772	59.735	19.519	61.711	41.293	90.526
	DRESO-L	6.559†	34.641	**7.833**	43.348	40.677	76.042†	17.408	56.913†	11.588†	50.161†	16.744	63.266†
	DRESO-B	6.599	**33.748**	7.849†	41.912†	41.018	82.182	17.191†	59.229	14.750	**44.073**	16.015†	64.359
	DRESO-T	7.425	33.884†	9.239	44.147	40.375	77.599	17.730	58.016	16.071	51.572	19.878	69.213
64x64	FNO	**5.145**	**55.800**	**10.205**	**60.825**	**26.176**	**68.071**	**10.467**	**47.563**	**13.908**	59.341	**10.023**	**63.760**
	UNet	15.623†	79.133	19.629	84.473	65.844†	120.926	35.095†	85.756	22.916	223.741	44.945	101.861
	PDE-ReFiner-Small	16.912	74.760	20.857	79.447	73.399	92.405	44.190	79.116	17.289	80.569	44.639†	102.723
	DRIFT-Net	19.211	63.928	19.069	67.228†	68.160	84.845†	46.586	81.822	22.938	87.292	62.515	94.056
	DRESO-L	17.289	61.352	17.463	76.565	70.674	85.293	43.608	72.248	14.133†	57.585†	50.736	86.676†
	DRESO-B	17.341	57.966	17.887	68.984	74.400	88.721	43.256	73.267	14.602	**47.227**	50.839	89.340
	DRESO-T	17.830	57.178†	17.205†	71.175	72.303	86.874	44.544	72.153†	20.180	64.201	51.792	92.869

Table 15 evaluates direct checkpoint transfer across spatial grids. DRESO and DRIFT-Net dynamically rebuild their patch grids and Fourier operations and interpolate absolute positional embeddings when enabled. PDEBench FNO evaluates the FFT on the target grid, regenerates normalized coordinates, and retains a fixed mode count smaller than the target grid's available frequencies. PDEBench UNet is fully convolutional and accepts these resolutions because each side length is divisible by its total downsampling factor of 16. These are direct inference changes induced by the input shape; no target-resolution optimization is performed.

Three retained baselines are not included because their released checkpoints do not support native cross-resolution inference under this protocol. NESTOR binds the patch embedding and learned positional table to the training token grid; rebuilding the target-size model and interpolating or migrating its positional parameters would constitute an additional adaptation procedure. NIPs associates independent learned complex weights with discrete Fourier coordinates, so changing the grid changes both the frequency lattice and parameter shape. RNO fixes the spatial size, coordinate buffers, and discrete Radon structure at construction and explicitly rejects a different input size. We therefore do not report these models as zero-shot failures; they are outside the native-inference comparison.

Relative to the closest spectral-coupled baseline, DRESO-L has lower error than DRIFT-Net in all 12 one-step and rollout cells at both 128×128 and 96×96. Under the larger $128\to64$ shift, it retains lower error on five of six one-step tasks and four of six rollout tasks, or nine of 12 cells overall. The updated results change the overall cross-resolution ranking: FNO attains the lowest error in nine of 12 cells at 96×96 and 11 of 12 cells at 64×64. The remaining best results are obtained by DRESO-L and DRESO-B at 96×96, and by DRESO-B on CE-KH rollout at 64×64. Thus, DRESO retains a consistent advantage over DRIFT-Net, whereas FNO provides the strongest direct transfer under the two down-resolution shifts.

Within the DRESO family, the Large variant has the best absolute performance in most cells, while its margin over Tiny narrows or reverses on several 64×64 entries. The table establishes this trend but not its mechanism. Greater specialization to the 128×128 training statistics is one possible explanation for future study; without targeted diagnostics, it should not be interpreted as a demonstrated capacity effect.

E.8 SHARED-REGION MIXER ABLATION

Table 16: Shared-region mixer ablation on one-step prediction and 1:20 rollout. The original DRESO-L design uses separate channel mixers for the low- and high-frequency regions, whereas the shared-mixer variants tie the two regions to one common mixer. All entries are RelL1 errors; lower is better. Bold marks the best result; † marks the runner-up.

Model variant	NS-Gauss		CE-RP		CE-CRP		CE-Gauss		CE-KH		NS-Sines	
	One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout
Shared mixer w/o dynamic gate	3.046	26.299	4.472	31.087	21.091	57.701	3.780	34.438	6.654	33.162	6.168	53.160
Shared mixer + dynamic gate	1.748†	11.382†	**3.991**	18.483†	19.026†	42.512†	2.309†	12.945†	6.421†	29.465†	4.412†	34.312†
Independent mixers + dynamic gate	**1.337**	**7.485**	3.997†	**15.714**	**17.498**	**40.372**	**1.917**	**11.059**	**5.823**	**29.152**	**3.571**	**25.785**

Table 16 evaluates the contribution of independent regional channel transformations in addition to separate gates. Compared with the gated shared-mixer variant, independent regional mixers reduce error in 11 of the 12 evaluated one-step and rollout settings. The exception is CE-RP one-step prediction, where the shared mixer is marginally better (3.991 versus 3.997). These results indicate that gate control alone is generally insufficient to differentiate the two branches. The gate adaptively weights their contributions, while independent mixers retain region-specific channel interactions. Removing the gate from the shared-mixer variant further increases error in all 12 settings. Thus, independent mixers specialize the regional transformations, and the gate controls how strongly the transformed signals enter each prediction.

E.9 EFFECT OF THE HIGH-FREQUENCY AUXILIARY LOSS

Table 17: One-step prediction and 1:20 rollout after removing the high-frequency-weighted auxiliary loss from each DRESO variant. DRIFT-Net is included as a reference. All entries are RelL1 errors; lower is better. No best or second-best markers are applied.

Model variant	NS-Gauss		CE-RP		CE-CRP		CE-Gauss		CE-KH		NS-Sines	
	One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout
DRIFT-Net (reference)	7.140	31.753	12.011	43.611	35.051	60.038	8.229	38.161	15.007	49.587	20.789	75.499
DRESO-L w/o HF loss	1.581	9.211	4.517	18.052	17.767	41.390	2.146	12.638	6.131	30.006	4.017	27.683
DRESO-B w/o HF loss	2.096	12.883	5.459	20.546	19.743	42.448	2.750	14.692	8.003	42.971	5.205	36.032
DRESO-T w/o HF loss	3.137	17.494	7.953	28.288	24.302	49.671	4.080	21.075	10.945	66.095	8.670	49.656

Table 17 isolates the contribution of the high-frequency-weighted auxiliary term in Equation 16. Relative to the matched DRESO rows trained with the complete objective in Tables 1 and 3, removing this term increases error in 35 of the 36 variant–metric comparisons. The remaining comparison improves: DRESO-T on NS-Sines one-step. Thus, the auxiliary signal is not universally beneficial. Despite its removal, DRESO-L and DRESO-B remain better than DRIFT-Net in all 12 comparisons, while DRESO-T remains better in 11 of 12. For DRESO-T, the only reversal is the CE-KH rollout. The auxiliary term therefore improves the aggregate accuracy profile, but the architectural advantage over DRIFT-Net largely remains without it.

E.10 ORIGINAL PRETRAINED BASELINE CONFIGURATIONS

Table 18: One-step prediction and 1:20 rollout for the original full-sized NESTOR and PDE-ReFiner configurations on the six CamLab-ETHZ tasks. All entries are RelL1 errors; lower is better.

Model	NS-Gauss		CE-RP		CE-CRP		CE-Gauss		CE-KH		NS-Sines	
	One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout	One-step	Rollout
NESTOR	0.869	4.254	3.664	10.361	18.295	37.430	2.036	7.619	4.774	19.714	2.225	13.408
PDE-ReFiner	2.354	19.530	4.808	27.282	14.747	48.115	3.060	23.865	4.455	43.004	8.242	29.670

E.11 LEARNED LINEAR EVIDENCE COEFFICIENTS

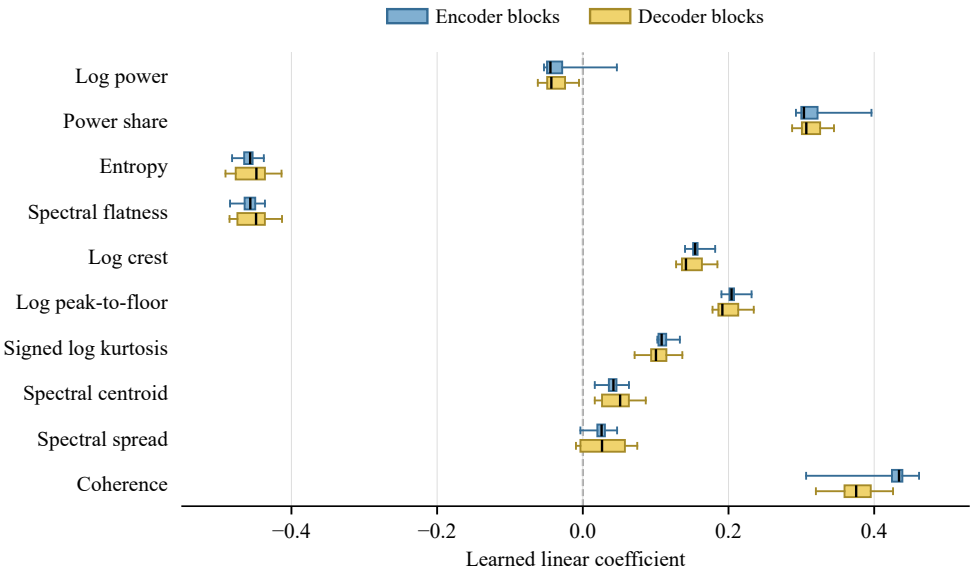


Figure 11: Learned linear evidence coefficients in trained DRESO-L. Blue boxes summarize the 16 encoder blocks, and yellow boxes summarize the 16 decoder blocks separately for each feature. Boxes show the interquartile range, black lines show the median, and whiskers span the minimum and maximum. The dashed line marks zero. These distributions describe variation within one trained model, not uncertainty across training runs.

Figure 11 characterizes the linear term $\boldsymbol{w}^\top z_{bcr}$ in Equation 10. The exported prior multipliers are all one, so the effective coefficients equal the learned weights. A positive coefficient increases the linear regional logit when the corresponding normalized descriptor increases; a negative coefficient decreases it. The same rule applies to both frequency regions. Its sign therefore does not specify a fixed preference for the low- or high-frequency branch.

Entropy and spectral flatness remain negative in all 32 blocks, with mean coefficients -0.456 and -0.455. Power share and coherence remain positive, with means 0.315 and 0.401. In the linear branch, these signs favor regions with greater energy share, more concentrated spectra, and stronger cross-channel coherence. Log crest, log peak-to-floor, and signed log kurtosis are also positive in every block. Log power, centroid, and spread acquire nonzero weights despite their neutral initialization. Log power and spread additionally vary in sign across blocks. This pattern is consistent with learned block-specific corrections to the DSP-informed starting direction.

These coefficients are a parameter diagnostic rather than a causal feature-importance measure. The realized contribution also depends on the normalized input descriptors. The nonlinear branch adds descriptor interactions, and the regional softmax converts the combined logits into gains. Coefficient

magnitude alone therefore does not establish a feature's effect on prediction accuracy or rollout stability.