# PAIR: PREDICTION WITH ADDITIVE INTERMEDIATE READOUTS FOR LONG-HORIZON PDE FORECASTING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Long-horizon accuracy in autoregressive PDE forecasting depends on both prediction errors and their propagation through the dynamics. Multiscale architectures construct coarse-grid features, but these representations typically contribute to the predicted physical field implicitly through the remaining decoder. In our periodic Navier–Stokes simulations, we observed that low-frequency initial perturbations produce larger deviations for long-horizon responses than equal-norm high-frequency perturbations. Motivated by this scale dependence, we investigate whether giving intermediate features an explicit coarse-scale prediction role can improve both coarse-scale learning and the full-resolution predictor. To this end, we introduce **Prediction with Additive Intermediate Readouts (PAIR)**, which maps intermediate features to a coarse physical-field contribution, upsamples it with a fixed operator, and adds it to the backbone's full-resolution prediction. A unified prediction loss on the combined output jointly trains both paths and their shared features, providing an additional route for the prediction loss to reach intermediate representations, allowing the coarse readout and the full-resolution path to adapt together without a separate coarse-scale supervision objective. Across periodic Navier–Stokes flow, reaction–diffusion, and heterogeneous acoustic scattering, PAIR reduces mean endpoint error by 72.8%, 87.5%, and 72.6%, respectively, relative to the corresponding U-Net baselines after 99–100 autoregressive steps. Further experiments show that joint training improves the full-resolution path's predictions in a high-frequency band outside the coarse readout's spectral support, suggesting our paired design decomposing the dynamics offer a better control of error propagation for long-term rollout.

## 1 INTRODUCTION

Repeatedly solving time-dependent PDEs across initial conditions and physical parameters can be computationally expensive. Neural surrogates such as DeepONet and Fourier neural operators learn PDE solution mappings from data to accelerate these computations (Lu et al., 2021; Li et al., 2021; Kovachki et al., 2023). In autoregressive forecasting, each prediction becomes the input to the next update, carrying its errors into subsequent evolution. Even a model with accurate one-step predictions can therefore deviate from the reference trajectory over a long rollout. Long-horizon accuracy depends on both the size of prediction errors and how they propagate through the dynamics (Brandstetter et al., 2022; McCabe et al., 2023).

In our periodic Navier–Stokes reference simulations, low-frequency initial perturbations produce larger average long-horizon responses than high-frequency perturbations of the same norm (Figure 1). In this flow, equally sized state deviations can therefore have different long-term effects depending on their spatial scale. This difference motivates examining how coarse-scale predictions are learned and how they affect subsequent evolution.

Multiscale architectures such as U-Net already extract coarse-grid features (Ronneberger et al., 2015; Gupta & Brandstetter, 2023; Rahman et al., 2023; Raonic et al., 2023). At these hidden layers, the features typically have no direct mapping to the physical field and no additional prediction constraint. The final prediction loss acts on them through the remaining decoder. Additive prediction designs in vision and PDE modeling suggest a way to establish such a mapping (Long et al., 2015; Xie & Tu, 2015; Karras et al., 2020; Qin et al., 2024; Guo & Li, 2024). A physi-



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.

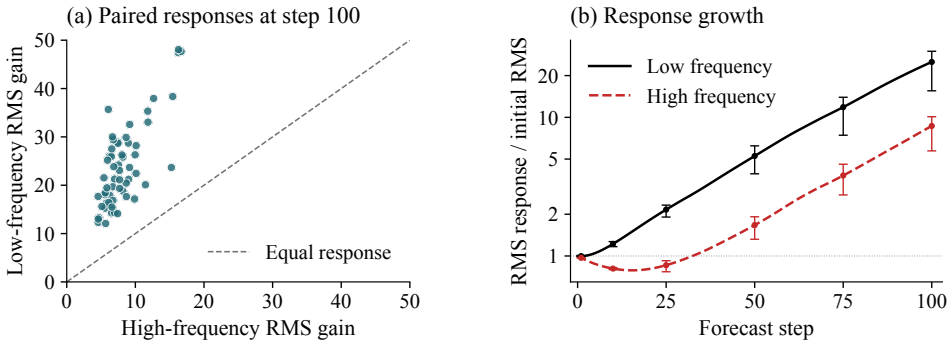



Figure 1: **Scale-dependent responses in periodic flow.** Initial perturbations have equal RMS in the two bands, set to 1% of each state's demeaned velocity RMS. Py2D responses are measured relative to clean rollouts. (a) Each point pairs high- and low-frequency RMS gains at step 100 for one of 60 IC2 states. The dashed line marks equal gains; all points lie above it. (b) Pooled RMS response gains over time, with bars showing the 16th–84th percentiles of state-wise gains at selected steps. Perturbation bands and averaging rules are given in Appendix F.

cal contribution read from intermediate features can be included in the final output, allowing the prediction loss to reach those features directly through this additional path. Since the decoder uses the same features, this learning can also affect the original full-resolution predictor.

We introduce **Prediction with Additive Intermediate Readouts (PAIR)** to give existing coarse-grid features a direct role in physical-field prediction (Figure 2). PAIR learns a coarse contribution jointly with the backbone's original full-resolution path, which can predict both coarse and fine structure.

Our contributions are threefold:

- **Coarse-scale prediction through intermediate readouts.** PAIR maps intermediate features to a coarse-scale output contribution, applies unlearnable upsampling, and adds it to the full-resolution prediction. Both paths are trained through a single loss on their sum, with the readout providing an additional gradient route to the shared features.
- **Local analysis of intermediate-readout learning.** We analyze how the added readout modifies training dynamics under a local approximation with frozen tangent kernels. The analysis identifies conditions under which errors along readout-supported directions decay faster than under the original backbone alone.
- **Improved long-horizon prediction accuracy.** Across three PDE tasks, PAIR reduces mean endpoint error relative to U-Net by 72.6–87.5% after 99–100 autoregressive steps. Controlled periodic-flow experiments further show that, on the same true input states, joint training improves the full-resolution path's predictions in a high-frequency band outside the readout's support.

## 2 Related Work

**Long-horizon PDE forecasting.** Autoregressive predictions carry model errors into subsequent updates. Pushforward training uses model-generated inputs to expose the model to deviations encountered during rollout (Brandstetter et al., 2022). PDE-Refiner instead refines each update iteratively to reduce one-step errors (Lippe et al., 2023). Their subsequent propagation depends on the update operator: spectral-layer designs address amplification and aliasing (McCabe et al., 2023), while SGNO constrains spectral evolution for periodic linear and semilinear PDEs (Li et al., 2026).

**Multiscale features and intermediate predictions.** Multiresolution operators and architectures such as U-Net represent coarse and fine structure through features at different resolutions (Ronneberger et al., 2015; Gupta & Brandstetter, 2023; Rahman et al., 2023; Raonic et al., 2023). Information from these scales can be fused as features within the network (Lin et al., 2017; Liu et al., 2017) or mapped to predictions that are then combined, as in FCN and StyleGAN2 (Long et al.,

2015; Karras et al., 2020). Prediction fusion and intermediate supervision can be chosen separately. HED fuses side outputs into a final prediction and trains with both side-output and fusion losses, while also evaluating training with the fusion loss alone (Xie & Tu, 2015). Deep supervision assigns training targets directly to intermediate predictions (Lee et al., 2015). HiLNO supervises predictions at multiple scales and uses the finest-scale output at inference (Hu et al., 2026).

**Additive PDE prediction and joint learning.** Additive prediction combines a base prediction with a correction, with the learning process depending on how the components are trained. SpecB-FNO and MgFNO train residual operators in stages (Qin et al., 2024; Guo & Li, 2024), while PPNN and $P^2C^2$Net learn corrections to coarse PDE updates (Liu et al., 2024; Wang et al., 2024). Multiple prediction contributions can also be learned jointly: MscaleFNO sums parallel FNO branches (You et al., 2026), and an optional configuration of the official F-FNO implementation accumulates layerwise predictions within one backbone (Tran et al., 2023). PAIR adds a coarse readout to existing intermediate features and trains it jointly with the full-resolution path. We study how this joint training affects the full-resolution path, particularly whether a coarse readout can improve high-frequency predictions beyond its own output support.

## 3 Problem Setup

Let $\mathbf{u}_t \in \mathcal{Y} = \mathbb{R}^{H\times W\times d}$ denote a $d$-channel physical state on an $H \times W$ grid at time $t$. We learn a one-step predictor $F_{\boldsymbol{\omega}}$ from consecutive states, $F_{\boldsymbol{\omega}}(\mathbf{u}_t) \approx \mathbf{u}_{t+1}$. For a generic training pair, we write $\mathbf{u} = \mathbf{u}_t$ and $\mathbf{y} = \mathbf{u}_{t+1}$. Any problem-specific context supplied to the model is left implicit.

Long-horizon forecasts are generated autoregressively:

$$\widehat{\mathbf{u}}_0 = \mathbf{u}_0, \qquad \widehat{\mathbf{u}}_{t+1} = F_{\boldsymbol{\omega}}(\widehat{\mathbf{u}}_t). \tag{1}$$

We also evaluate *true-input* predictions $F_{\boldsymbol{\omega}}(\mathbf{u}_t)$ on reference states. This separates one-step prediction error from the error accumulated by feeding previous forecasts back into the model. Dataset-specific state representations, normalization, and training losses are given in Appendix A.

## 4 Method

In this section, we introduce PAIR's additive intermediate readout for coarse-scale prediction and analyze its local training effects. The design is motivated by the larger average long-horizon responses to low-frequency initial perturbations observed in Figure 1.

### 4.1 Coarse Prediction from Intermediate Features

Multiscale backbones already provide coarse-grid features, but these hidden representations typically have no direct physical-field mapping or additional prediction constraint. They are trained through the prediction loss propagated by the remaining decoder. PAIR establishes this mapping at a selected intermediate layer and includes the resulting physical contribution in the final prediction.

Let $E_{\boldsymbol{\eta}}$ denote the backbone computation from the input to the selected intermediate layer, with parameters $\boldsymbol{\eta}$. The resulting feature map is

$$\mathbf{z} = E_{\boldsymbol{\eta}}(\mathbf{u}) \in \mathbb{R}^{H_{\text{int}}\times W_{\text{int}}\times d_z}.$$

denote the selected intermediate feature map. To keep the notation light, any additional context supplied to the backbone, including skip or pyramid features and problem-specific inputs, is left implicit. We write $D_{\boldsymbol{\theta}}(\mathbf{z})$ for the remaining backbone computation and its original output head. The unmodified predictor is therefore

$$F_{\text{base}}(\mathbf{u}) = D_{\boldsymbol{\theta}}(\mathbf{z}). \tag{2}$$

An affine convolutional readout $q_{\boldsymbol{\phi}}$ maps $\mathbf{z}$ from $d_z$ hidden channels to $d$ physical channels on the intermediate grid. An unlearnable linear upsampling operator $\mathbf{U}$ maps this field to the output grid; $\mathbf{U}$ is the identity when the two grids coincide. PAIR combines

$$F_{\text{coarse}}(\mathbf{u}) = \mathbf{U}q_{\boldsymbol{\phi}}(\mathbf{z}), \qquad F_{\text{full}}(\mathbf{u}) = D_{\boldsymbol{\theta}}(\mathbf{z}), \tag{3}$$

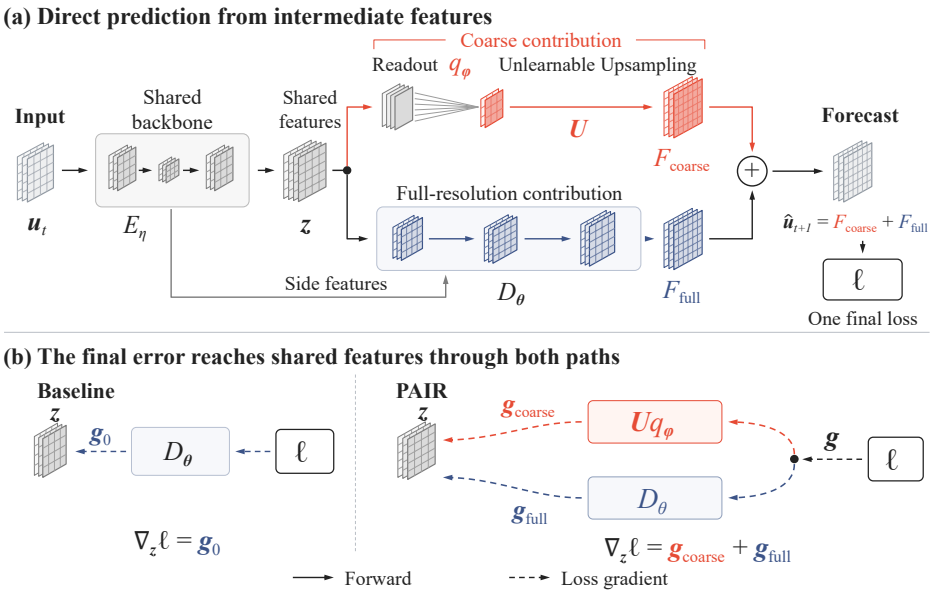



Figure 2: **Direct coarse prediction also trains shared features.** (a) PAIR maps intermediate features to a coarse physical contribution in the final prediction, while the full-resolution path can predict both coarse and fine structure. (b) A single loss on their sum updates shared features $\mathbf{z}$ through both paths. Learning the coarse contribution can therefore also change the representation used by the full-resolution predictor.

into the final prediction

$$F_{\text{PAIR}}(\mathbf{u}) = F_{\text{coarse}}(\mathbf{u}) + F_{\text{full}}(\mathbf{u}). \tag{4}$$

The unlearnable upsampling operator determines the output space available to the coarse contribution. With Fourier upsampling, this space is restricted to the coarse spectral support. The full-resolution path retains the backbone's original output head and can predict both coarse- and fine-scale structure. The two contributions can therefore overlap within the coarse output space. Their combination is learned through a single loss on the final prediction, as detailed next. Appendix A specifies the selected interfaces, readout heads, unlearnable upsampling rules, and initialization.

### 4.2 Joint Learning Through the Final Prediction

PAIR trains the two paths with a single loss on their final sum, with parameters $\boldsymbol{\omega} = (\boldsymbol{\eta}, \boldsymbol{\theta}, \boldsymbol{\phi})$:

$$\mathcal{L}(\boldsymbol{\omega}) = \mathbb{E}_{(\mathbf{u},\mathbf{y})}[\ell(F_{\text{coarse}}(\mathbf{u}) + F_{\text{full}}(\mathbf{u}), \mathbf{y})]\,. \tag{5}$$

The objective supervises the combined prediction without assigning an independent target to either contribution. Each path is updated using the error of the current combined prediction. Because the two paths also share intermediate features, their learning is coupled through the representation used by the decoder.

To make this coupling explicit, let $\mathbf{g} = \nabla_{\hat{\mathbf{y}}} \ell(\hat{\mathbf{y}}, \mathbf{y})$ for $\hat{\mathbf{y}} = F_{\text{PAIR}}(\mathbf{u})$, and define $\mathbf{J}_{\text{coarse}} = \partial F_{\text{coarse}}/\partial \mathbf{z}$ and $\mathbf{J}_{\text{full}} = \partial F_{\text{full}}/\partial \mathbf{z}$. Then

$$\nabla_{\mathbf{z}} \ell = \underbrace{\mathbf{J}_{\text{coarse}}^{\top}\mathbf{g}}_{\mathbf{g}_{\text{coarse}}} + \underbrace{\mathbf{J}_{\text{full}}^{\top}\mathbf{g}}_{\mathbf{g}_{\text{full}}}. \tag{6}$$

The first term is the direct gradient route created by the readout. Because both paths share $\mathbf{z}$, this route can reorganize the intermediate representation and thereby change the prediction learned by $F_{\text{full}}$. Sections 5.3–5.4 test these two effects through gradient controls and improvements outside the readout's output support.

### 4.3 Training Effects of the Coarse Readout

The added readout changes learning through both its own trainable parameters and its coupling to shared features. We first isolate the additional parameter channel by comparing PAIR and the backbone at an identical initial predictor. We then examine how changes in the trained full-resolution path can be distinguished from the readout's direct contribution. Throughout, we define the output space permitted by the unlearnable upsampling operator as $\mathcal{V} = \mathrm{Range}(\mathbf{U})$.

**Faster error reduction in supported directions.** Match the backbone parameters of PAIR and the baseline, and initialize all readout weights and biases to zero. The two models then begin from the same predictor and residual on a fixed training batch. Let $\mathbf{J}_{\mathrm{base}}$ be the output Jacobian with respect to the backbone parameters, and let $\mathbf{J}_{\boldsymbol{\phi}}$ be the output Jacobian with respect to the added readout. At this matched state,

$$\mathbf{K}_{\mathrm{PAIR}} = \underbrace{\mathbf{J}_{\mathrm{base}}\mathbf{J}_{\mathrm{base}}^{\top}}_{\mathbf{K}_{\mathrm{base}}} + \underbrace{\mathbf{J}_{\boldsymbol{\phi}}\mathbf{J}_{\boldsymbol{\phi}}^{\top}}_{\mathbf{K}_{\mathrm{coarse}}}, \qquad \mathbf{K}_{\mathrm{coarse}} \succeq 0. \tag{7}$$

Thus the added route contributes a positive-semidefinite learning operator. For squared loss $L = \frac{1}{2}\|\mathbf{e}\|^2$ under unit-rate gradient flow,

$$\dot{L}_{\mathrm{PAIR}} - \dot{L}_{\mathrm{base}} = -\mathbf{e}^{\top}\mathbf{K}_{\mathrm{coarse}}\mathbf{e} = -\|\mathbf{J}_{\boldsymbol{\phi}}^{\top}\mathbf{e}\|^2 \leq 0. \tag{8}$$

The readout therefore gives any residual direction visible to $\mathbf{J}_{\boldsymbol{\phi}}$ an additional descent channel. At zero initialization this channel first updates the readout parameters; as the readout becomes nonzero, Equation (6) also couples it to the shared intermediate state.

The next result extends this instantaneous comparison over a local training interval.

**Theorem 1** (Finite-time error reduction in readout-supported directions)**.** *Assume that, over a local interval $[0, \tau]$, $\mathbf{K}_{\mathrm{base}}$ and $\mathbf{K}_{\mathrm{coarse}}$ are fixed positive-semidefinite matrices that commute. Let $\mathbf{v}_j$ be a common eigenvector with eigenvalues $\mu_j^{\mathrm{base}}$ and $\mu_j^{\mathrm{coarse}}$. Starting from the same residual, squared-loss gradient flow satisfies*

$$|e_{\mathrm{PAIR},j}(\tau)|^2 = e^{-2\mu_j^{\mathrm{coarse}}\tau}|e_{\mathrm{base},j}(\tau)|^2 \leq |e_{\mathrm{base},j}(\tau)|^2. \tag{9}$$

*The inequality is strict for $\tau > 0$ whenever $\mu_j^{\mathrm{coarse}} > 0$ and the initial residual has a nonzero component along $\mathbf{v}_j$.*

The theorem quantifies the direct optimization benefit of PAIR: supported directions with nonzero residual receive additional learning strength and decay faster within the local regime. These directions lie in $\mathcal{V}$ on each training example. With Fourier upsampling, they remain within the coarse spectral support, and an individual tangent direction may combine several Fourier modes. Appendices B.2–B.3 give the derivation and the corresponding aggregate error reduction.

**Adaptation of the full-resolution path.** The result above describes direct learning through the readout. Joint training simultaneously updates the shared representation and $F_{\mathrm{full}}$. Let $F_{\mathrm{full}}^{\star}$ denote the full-resolution contribution learned within PAIR and $F_{\mathrm{base}}^{\star}$ an independently trained baseline. If $\mathbf{P}$ is the orthogonal projector onto $\mathcal{V}$, then $F_{\mathrm{coarse}}^{\star} \in \mathcal{V}$ implies

$$(\mathbf{I} - \mathbf{P})\,(F_{\mathrm{PAIR}}^{\star} - F_{\mathrm{base}}^{\star}) = (\mathbf{I} - \mathbf{P})\,(F_{\mathrm{full}}^{\star} - F_{\mathrm{base}}^{\star})\,. \tag{10}$$

Therefore, any improvement outside the readout support must be produced by the jointly adapted full-resolution path. Section 5.4 tests this implication directly. Appendix B.1 gives the complete output-space decomposition.

**Universal operator approximation.** PAIR preserves the universal approximation capacity of its backbone and augments it with an intermediate prediction path. Let $\mathcal{F}_{\mathrm{base}}$ and $\mathcal{F}_{\mathrm{PAIR}}$ denote the corresponding operator families. Since the affine readout contains the zero map, every backbone operator is represented by PAIR through $q_{\boldsymbol{\phi}_0} \equiv 0$, and hence

$$\mathcal{F}_{\mathrm{base}} \subseteq \mathcal{F}_{\mathrm{PAIR}}. \tag{11}$$

Let $\mathcal{K} \subset \mathcal{X}$ be compact, and let $C(\mathcal{K}; \mathcal{Y})$ denote the space of continuous functions from $\mathcal{K}$ to $\mathcal{Y}$. If the backbone family is dense in this space under the uniform norm, then

$$\overline{\mathcal{F}_{\mathrm{PAIR}}}^{\,\|\cdot\|_\infty} = C(\mathcal{K}; \mathcal{Y}). \tag{12}$$

PAIR therefore inherits the backbone's universal approximation property, when it holds, while adding a direct learning path for coarse-scale prediction. Appendix B.4 states and proves the result.

Table 1: **Autoregressive prediction error.** Mean ± sample SD across three training seeds, after averaging test-case errors within each seed. The endpoint $h$ is 99 for RD and 100 otherwise. Bold marks the lowest mean per task and forecast step.

| Model | Step 1 | Step 10 | Step 50 | Step $h$ |
|---|---|---|---|---|
| **Periodic flow** nMSE | $(\times 10^{-7})$ | $(\times 10^{-5})$ | $(\times 10^{-3})$ | $(\times 10^{-2})$ |
| U-Net | 3.431 ±0.113 | 4.347 ±0.334 | 2.083 ±0.214 | 1.395 ±0.042 |
| FNO (modes=16) | 63.153 ±0.790 | 4.694 ±0.073 | 1.066 ±0.035 | 0.911 ±0.008 |
| FNO (modes=32) | 14.962 ±0.252 | 0.970 ±0.033 | 1.598 ±0.690 | 1.264 ±0.277 |
| **PAIR** | **1.465** ±0.050 | **0.607** ±0.052 | **0.312** ±0.041 | **0.380** ±0.035 |
| **RD** raw MSE | $(\times 10^{-6})$ | $(\times 10^{-6})$ | $(\times 10^{-5})$ | $(\times 10^{-4})$ |
| U-Net | 2.598 ±0.147 | 1.542 ±1.607 | 3.200 ±2.804 | 1.027 ±0.854 |
| FNO (modes=16) | 148.233 ±7.475 | 44.891 ±5.309 | 7.496 ±1.534 | 3.676 ±1.026 |
| FNO (modes=32) | 50.138 ±0.739 | 1.912 ±0.101 | 0.750 ±0.073 | 0.426 ±0.050 |
| **PAIR** | **2.491** ±0.005 | **0.189** ±0.028 | **0.227** ±0.147 | **0.129** ±0.082 |
| **Acoustics** scaled MSE | $(\times 10^{-4})$ | $(\times 10^{-3})$ | $(\times 10^{-2})$ | $(\times 10^{0})$ |
| U-Net | 2.310 ±0.132 | 0.731 ±0.046 | 0.691 ±0.210 | 0.0446 ±0.0208 |
| FNO (modes=16) | 36.552 ±0.235 | 15.058 ±0.519 | 18.247 ±3.509 | 19.809 ±19.480 |
| FNO (modes=32) | 15.903 ±0.080 | 2.404 ±0.064 | 4.158 ±1.401 | 2.892 ±2.430 |
| **PAIR** | **1.590** ±0.065 | **0.432** ±0.008 | **0.291** ±0.047 | **0.0122** ±0.0024 |

# 5 Experiments

In this section, we evaluate PAIR's long-horizon accuracy on three PDE tasks and use controlled experiments to examine the roles of intermediate readouts and joint training. The main experiments use U-Net, with periodic-flow extensions to ResNet and ResNet18-FPN (He et al., 2016; Lin et al., 2017) in Appendix C.5.

## 5.1 Tasks, baselines, and evaluation

Periodic Navier–Stokes forecasting uses $256^2$ velocity fields generated by Py2D at $\mathrm{Re} = 5000$ (Jakhar et al., 2024), with IC1 for training and validation and an independently initialized IC2 trajectory for testing. Reaction–diffusion uses the $128^2$ activator–inhibitor fields from PDEBench (Takamoto et al., 2022). Acoustic scattering uses The Well's discontinuous-medium data at $256^2$, predicting pressure and velocity conditioned on density and sound speed (Ohana et al., 2024).

The baselines are the unmodified U-Net and FNO with 16 or 32 Fourier modes per spatial axis. Rollouts cover 100 steps for periodic flow and acoustics, and 99 for reaction–diffusion. We use nMSE for periodic flow, raw MSE for reaction–diffusion, and MSE scaled by training-channel variance for acoustics. All checkpoints are selected on validation data. Appendix A provides metric definitions, data splits, and training configurations.

## 5.2 Long-horizon forecasting accuracy

We first test whether the added coarse prediction path improves accuracy over autoregressive rollouts. PAIR achieves the lowest mean endpoint error on all three PDE tasks, reducing U-Net's error by 72.8%, 87.5%, and 72.6% for periodic flow, reaction–diffusion, and acoustics, respectively (Table 1). For the periodic-flow model, the added readout increases the parameter count from 8.914M to 8.915M. Batch-4 FP32 inference time on a single H800 rises from 15.66 to 15.78 ms, with the timing protocol provided in Appendix A.

In reaction–diffusion, PAIR and U-Net begin with nearly equal one-step errors, but their errors separate as predictions are repeatedly fed back into the models (Figure 3). In the endpoint example in Figure 4, U-Net expands the region enclosed by the reference activator contour, while both FNO configurations contract it. PAIR places the boundary closer to the reference. In acoustics, PAIR also follows the curved reference wavefront more closely. Across all 200 acoustic test trajectories, its endpoint physical-energy error is 5.23%, compared with 11.49% for U-Net (Appendix E).

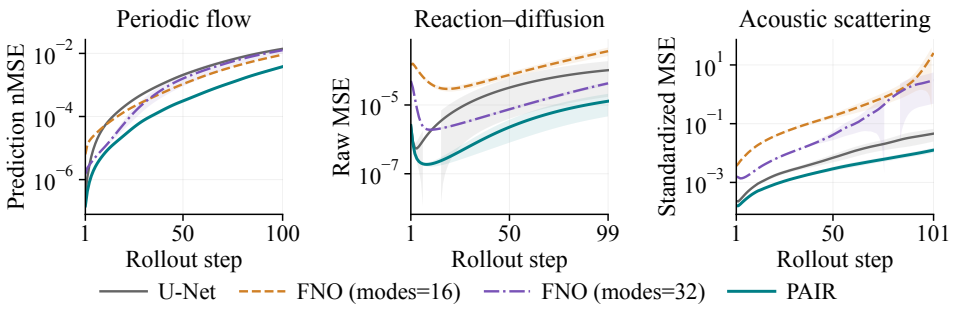



Figure 3: **Autoregressive rollout error.** Curves show mean test-set error across three training seeds, with shading for ± one sample SD. Metrics follow Table 1. Acoustic curves extend to step 101.

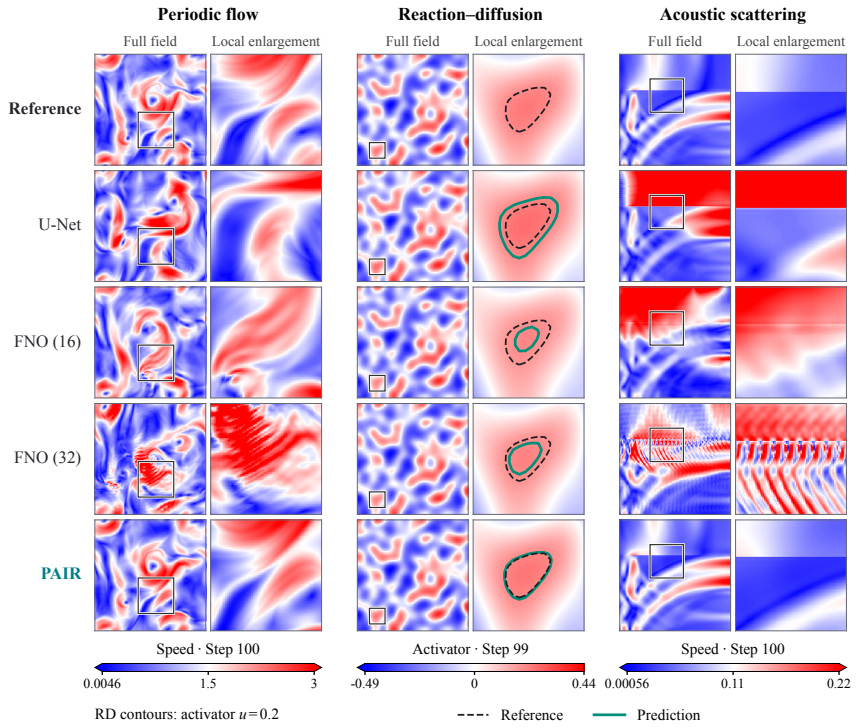



Figure 4: **Predicted fields and enlargements at step 100 (99 for RD).** Periodic flow and acoustics show speed, and RD shows activator concentration. Each task uses the full reference field's color range, with triangles marking out-of-range values. RD contours at $u = 0.2$ are black dashed for the reference and teal solid for predictions.

In periodic flow at step 100, PAIR's mean relative Fourier error in $0 \le k < 32$ over all 100 test starts is 0.456, compared with 0.907 for U-Net and 0.734 for FNO with 16 modes (Table 15). Figure 5 shows lower velocity-field error for PAIR at low wavenumbers, even where the models' energy spectra are close. FNO with 16 modes retains an advantage in the middle and high bands. Across the same 100 test starts, PAIR also reduces U-Net's mean energy-spectrum discrepancy from 6.56% to 2.66% (Appendix E).

### 5.3 Intermediate readouts and joint learning

To understand these long-horizon gains, we compare PAIR's joint prediction loss with auxiliary supervision, using the same $64^2$ feature grid, $3 \times 3$ head, and initialization. The auxiliary model's coarse output fits the Fourier-restricted target, while its full-resolution output fits the reference field.

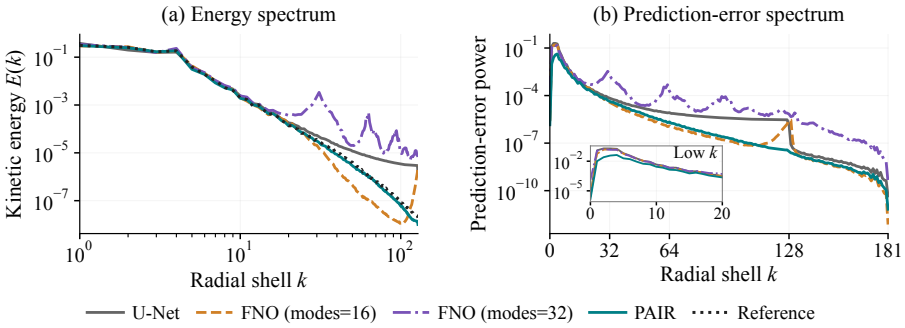



Figure 5: **Periodic-flow energy and velocity prediction-error spectra at step 100.** Curves average four common IC2 test starts, with neural-model spectra also averaged over three training seeds.

Table 2: **Intermediate-readout controls and full-resolution path diagnostics.** Mean ± sample SD across three training seeds. (a–c) Endpoint and time-averaged periodic nMSE over 100 IC2 starts, at step 100 and over steps 1–100, respectively ($\times 10^{-3}$). (d) One-step error power in $k \geq 64$ from common true inputs ($\times 10^{-6}$). Bold marks the lowest mean per group and metric.

| **(a) Supervision** (default init., AMP) | Endpoint | Time average |
|---|---|---|
| U-Net | 13.949 ±0.416 | 3.864 ±0.250 |
| Auxiliary supervision | 14.894 ±0.494 | 4.293 ±0.283 |
| **PAIR** | **3.801** ±0.347 | **0.816** ±0.080 |
| **(b) Readout placement** (zero-init. heads, AMP) | Endpoint | Time average |
| U-Net | 13.949 ±0.416 | 3.864 ±0.250 |
| Final-layer readout | 14.475 ±0.746 | 4.084 ±0.257 |
| Final-layer readout, matched support | 9.143 ±0.445 | 2.273 ±0.085 |
| Intermediate readout | **6.767** ±0.101 | **1.562** ±0.015 |
| **(c) Gradient connectivity** (default init., FP32) | Endpoint | Time average |
| Joint training | **3.831** ±0.323 | **0.821** ±0.071 |
| Stopped readout-to-feature gradient | 19.420 ±1.281 | 6.826 ±0.610 |

| **(d) Full-resolution path** | $\mathbf{u}_0$ | $\mathbf{u}_9$ | $\mathbf{u}_{49}$ | $\mathbf{u}_{99}$ |
|---|---|---|---|---|
| U-Net | 5.538 ±0.151 | 5.366 ±0.147 | 5.345 ±0.142 | 5.342 ±0.134 |
| **PAIR** | **1.967** ±0.128 | **1.982** ±0.140 | **1.987** ±0.140 | **1.958** ±0.140 |

Both losses update shared features, but only the full-resolution output supplies the forecast. With the auxiliary weight selected on IC1 validation data, this model has higher endpoint and time-averaged errors than PAIR (Table 2(a)). The objective and weight selection are detailed in Appendix C.1.

Within the additive prediction design, the coarse contribution can be read from intermediate or final-layer features. We compare these placements in a separate control with zero-initialized, 130-parameter heads. The final-layer proposal is restricted to $64^2$ and passed through the same Fourier upsampling operator, matching the intermediate readout's output support. The intermediate attachment yields lower endpoint and time-averaged errors (Table 2(b)), with head details in Appendix C.2.

Moving the readout changes both its input features and the path through which gradients reach them. We therefore keep the intermediate attachment fixed and detach $\mathbf{z}$ at the readout input to examine the added learning route in Equation (6). The forward prediction remains $F_{\text{coarse}} + F_{\text{full}}$, and both branches remain trainable. In the matched FP32 comparison, detaching this gradient increases endpoint error by a factor of 5.07 and also raises time-averaged error (Table 2(c)).

### 5.4 Adaptation of the full-resolution path

The added readout sends gradients to intermediate features shared by both paths. We therefore examine whether joint training improves predictions made by the full-resolution path itself. For the

default $64^2$ Fourier head, the coarse contribution is zero in $k \geq 64$, so predictions in this band come entirely from $F_{\text{full}}$ (Equation (10)). Each model receives the same reference IC2 states at offsets $0, 9, 49, 99$ within each test window and predicts the next step. Using reference states makes this comparison independent of the models' preceding rollout trajectories.

Across the four sets of reference inputs, PAIR reduces mean error power in this band by 62.8–64.5% relative to U-Net (Table 2(d)). The reduction comes from the jointly trained full-resolution path, whose architecture is unchanged from the backbone. The gradient control supports the contribution of the added learning route to rollout accuracy, while this diagnostic shows improved predictions from the original path beyond the coarse readout's output support. Further true-input diagnostics and frozen-feature probes are provided in Appendix D.

### 5.5 Accuracy of learned perturbation responses

Using the perturbation protocol of Figure 1, we test whether the trained models reproduce the scale-dependent responses that motivated PAIR. At step 100, PAIR preserves Py2D's larger response to low-frequency than high-frequency initial perturbations, while U-Net and FNO with 32 modes reverse this ordering (Figure 6). For high-frequency perturbations, FNO with 16 modes produces a squared response magnitude of 0.0001551, below Py2D's 0.008678.

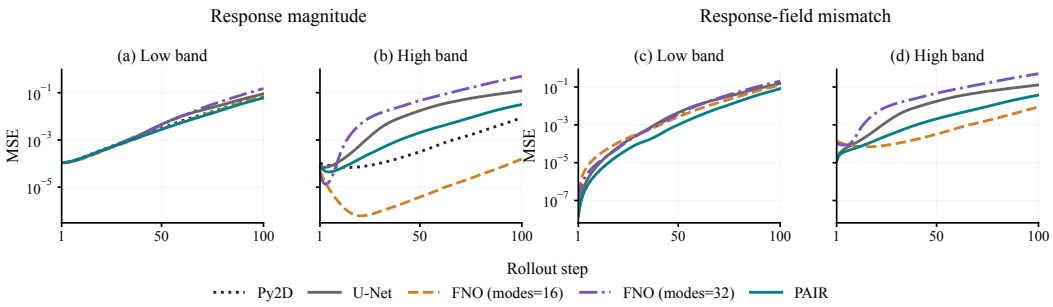



Figure 6: **Responses to initial perturbations.** Panels (a,b) show mean squared response magnitudes and (c,d) MSE between neural and Py2D response fields, computed from unnormalized velocities. Responses are differences from the corresponding clean rollouts. Low/high denote initial perturbation bands. Curves average paired 1% perturbations over 60 IC2 states, with three-seed averages for neural models.

PAIR reduces U-Net's step-100 response-field MSE against Py2D from 0.1527 to 0.08153 for low-frequency perturbations and from 0.1296 to 0.03720 for high-frequency perturbations. PAIR also has lower low-frequency response-field error than both FNO configurations, while FNO with 16 modes has the lowest high-frequency error. Ten-state tests at additional perturbation amplitudes preserve the same ordering (Appendix F).

## 6 Conclusion

In this work, we introduced PAIR for long-horizon PDE forecasting. Periodic-flow solver experiments showed larger average long-horizon responses to low-frequency initial perturbations than to equal-norm high-frequency perturbations, motivating a direct learning path for coarse-scale prediction. PAIR reads a coarse physical contribution from intermediate features, applies unlearnable upsampling, and adds it to the full-resolution output. One loss on the sum jointly trains both paths and their shared features.

Local training-dynamics analysis establishes conditions for faster training-error decay in readout-supported directions. Experiments on three PDE tasks show improved long-horizon accuracy over U-Net. Periodic-flow controls further show that joint training improves the full-resolution path's high-frequency predictions beyond the readout's support. Future work will explore joint readouts at multiple intermediate resolutions to understand how learning prediction contributions across scales affects long-horizon forecasting.

### AI USE STATEMENT

We used generative AI tools to assist with language editing, translation, and relevant literature searches, and used tools such as Codex to run code following the authors' instructions. All AI-generated or AI-modified content was reviewed by the authors before inclusion in the manuscript. The authors take responsibility for the final content of the paper, including its scientific claims, experimental results, and references.

### REPRODUCIBILITY STATEMENT

Appendix A details the datasets, data splits, preprocessing, model configurations, training procedures, and checkpoint selection. Appendix B provides the assumptions and proofs for the theoretical results. Appendices C–F specify the ablation protocols, diagnostic evaluations, and perturbation experiments.

## REFERENCES

Johannes Brandstetter, Daniel E. Worrall, and Max Welling. Message passing neural PDE solvers. In *International Conference on Learning Representations*, 2022.

Gideon Dresdner, Dmitrii Kochkov, Peter Christian Norgaard, Leonardo Zepeda-Nunez, Jamie Smith, Michael Brenner, and Stephan Hoyer. Learning to correct spectral methods for simulating turbulent flows. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856.

Zi-Hao Guo and Hou-Biao Li. MgFNO: Multi-grid Architecture Fourier Neural Operator for Parametric Partial Differential Equations. *arXiv preprint arXiv:2407.08615*, 2024.

Jayesh K Gupta and Johannes Brandstetter. Towards multi-spatiotemporal-scale generalized PDE modeling. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition . In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

Zhicheng Hu, Jiacheng Li, and Min Yang. HiLNO: A hierarchical latent neural operator with multi-scale supervision for PDEs on general geometries. *arXiv preprint arXiv:2609.18419*, 2026.

Karan Jakhar, Yifei Guan, Rambod Mojgani, Ashesh Chattopadhyay, and Pedram Hassanzadeh. Learning closed-form equations for subgrid-scale closures from high-fidelity data: Promises and challenges. *Journal of Advances in Modeling Earth Systems*, 16(7):e2023MS003874, 2024.

Ruoxi Jiang, Peter Y. Lu, Elena Orlova, and Rebecca Willett. Training neural operators to preserve invariant measures of chaotic attractors. In *Advances in Neural Information Processing Systems*, volume 36, pp. 27645–27669. Curran Associates, Inc., 2023.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Felix Koehler, Simon Niedermayr, Rüdiger Westermann, and Nils Thuerey. Apebench: A benchmark for autoregressive neural emulators of pdes. In *Advances in Neural Information Processing Systems*, volume 37, pp. 120252–120310. Curran Associates, Inc., 2024.

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.

Chen-Yu Lee, Saining Xie, Patrick Gallagher, Zhengyou Zhang, and Zhuowen Tu. Deeply-Supervised Nets. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, volume 38 of *Proceedings of Machine Learning Research*, pp. 562–570. PMLR, 2015.

Jiayi Li, Penghao Jiang, Hira Saleem, Zhaonan Wang, Piotr Koniusz, and Flora D. Salim. SGNO: Spectral Generator Neural Operators for Stable Long Horizon PDE Rollouts. *arXiv preprint arXiv:2602.18801*, 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.

Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

Phillip Lippe, Bas Veeling, Paris Perdikaris, Richard Turner, and Johannes Brandstetter. Pde-refiner: Achieving accurate long rollouts with neural pde solvers. In *Advances in Neural Information Processing Systems*, volume 36, pp. 67398–67433. Curran Associates, Inc., 2023.

Xin-Yang Liu, Min Zhu, Lu Lu, Hao Sun, and Jian-Xun Wang. Multi-resolution partial differential equations preserved learning framework for spatiotemporal dynamics. *Communications Physics*, 7 (1):31, 2024.

Yu Liu, Yanming Guo, and Michael S. Lew. On the exploration of convolutional fusion networks for visual recognition. In *MultiMedia Modeling*, pp. 277–289, 2017.

Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3431–3440, 2015.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

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

Michael McCabe, Peter Harrington, Shashank Subramanian, and Jed Brown. Towards stability of autoregressive neural operators. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856.

Ruben Ohana, Michael McCabe, Lucas Meyer, Rudy Morel, Fruzsina J. Agocs, Miguel Beneitez, Marsha Berger, Blakesley Burkhart, Stuart B. Dalziel, Drummond B. Fielding, Daniel Fortunato, Jared A. Goldberg, Keiya Hirashima, Yan-Fei Jiang, Rich R. Kerswell, Suryanarayana Maddu, Jonah Miller, Payel Mukhopadhyay, Stefan S. Nixon, Jeff Shen, Romain Watteaux, Bruno Régaldo-Saint Blancard, François Rozet, Liam Parker, Miles Cranmer, and Shirley Ho. The well: a large-scale collection of diverse physics simulations for machine learning. In *Advances in Neural Information Processing Systems*, volume 37, pp. 44989–45037, 2024.

Shaoxiang Qin, Fuyuan Lyu, Wenhui Peng, Dingyang Geng, Ju Wang, Xing Tang, Sylvie Leroyer, Naiping Gao, Xue Liu, and Liangzhu Leon Wang. Toward a Better Understanding of Fourier Neural Operators from a Spectral Perspective. *arXiv preprint arXiv:2404.07200*, 2024.

Md Ashiqur Rahman, Zachary E Ross, and Kamyar Azizzadenesheli. U-NO: U-shaped neural operators. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856.

Bogdan Raonic, Roberto Molinaro, Tim De Ryck, Tobias Rohner, Francesca Bartolucci, Rima Alaifari, Siddhartha Mishra, and Emmanuel de Bézenac. Convolutional neural operators for robust and accurate learning of pdes. In *Advances in Neural Information Processing Systems*, volume 36, pp. 77187–77200. Curran Associates, Inc., 2023.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pp. 234–241, Cham, 2015. Springer International Publishing. ISBN 978-3-319-24574-4.

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

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

Qi Wang, Pu Ren, Hao Zhou, Xin-Yang Liu, Zhiwen Deng, Yi Zhang, Ruizhi Chengze, Hongsheng Liu, Zidong Wang, Jian-Xun Wang, Ji-Rong Wen, Hao Sun, and Yang Liu. Pˆ2cˆ2net: Pde-preserved coarse correction network for efficient prediction of spatiotemporal dynamics. In *Advances in Neural Information Processing Systems*, volume 37, pp. 68897–68925, 2024.

Saining Xie and Zhuowen Tu. Holistically-nested edge detection. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1395–1403, 2015.

Zilin You, Zhenli Xu, and Wei Cai. Mscalefno: Multi-scale fourier neural operator learning for oscillatory functions and wave scattering problems. *Journal of Computational Physics*, 547: 114530, 2026. ISSN 0021-9991.

Appendix A specifies the data and training protocols. Appendix B gives the derivations. Appendix C details the supervision and architecture controls, while Appendix D examines the full-resolution path and frozen representations. Additional forecast diagnostics, perturbation experiments, and field visualizations appear in Appendices E, F, and G, respectively.

# A Data, training, and evaluation

## A.1 Periodic Navier–Stokes flow

We use two-dimensional periodic incompressible flow at Reynolds number 5,000. Py2D (Jakhar et al., 2024)[1] generates the underlying $512^2$ trajectories with viscosity $\nu = 1/5000$, linear drag $0.1$, and vorticity forcing $-4\cos(4y)$. The numerical step is $10^{-4}$. Consecutive observations are separated by $0.05$, so 100 forecast steps cover physical time 5. The observation pipeline samples fine-grid vorticity to $256^2$ and recovers velocity there.

Each initial condition produces 20,000 saved frames, stored in ten consecutive parts numbered 00–09 with 2,000 frames each. Using one-based frame numbers, IC1 training uses frames 2,001–14,000 (parts 01–06), excluding the first 2,000 frames. Validation uses frames 16,001–20,000 (parts 08–09). Part 07 is unused, leaving a 2,000-frame gap between training and validation.

IC2 is generated from a different initial condition under the same physical and numerical settings. Formal rollout testing uses its last 4,000 frames, 16,001–20,000 (parts 08–09). From this segment, 100 starting indices are spaced uniformly over the valid range, each with 100 subsequent reference frames. Every model uses the same starts. The test windows overlap and belong to one trajectory. The three training seeds measure optimization variability on this common test set.

For start $i$, let $a_i$ be the range over both reference velocity channels, all spatial locations, the initial state, and the complete 100-step future reference. We report

$$\mathrm{nMSE}(h) = \frac{1}{N}\sum_{i=1}^{N} \frac{\mathrm{mean}_{j,m}\big[(\hat{u}_{i,h,j,m} - u_{i,h,j,m})^2\big]}{a_i^2}. \tag{13}$$

Here $N = 100$ is the number of test starts, $h$ is the forecast step, $j$ indexes velocity channels, and $m$ indexes grid points. The reference range $a_i$ is shared by every model evaluated from start $i$. Endpoint error uses the final reported step. Time-averaged error is the arithmetic mean of nMSE over steps 1–100. For each metric, cases are averaged within a training seed before computing the mean and sample SD across seeds 42, 43, and 44.

The U-Net backbone has base width 16, channel multipliers $(1, 2, 4, 8, 8, 8)$ over six resolutions, and two encoder residual blocks per level. All U-Net comparisons retain this configuration. The default intermediate readout is attached after the final decoder block at resolution $64^2$. Original feature skips remain available to the decoder. In the periodic U-Net, backbone convolutions use zero padding and the added spatial readout uses circular padding. The branch uses unlearnable Fourier upsampling unless a bilinear control is specified.

Training uses batch size 32, AdamW (Loshchilov & Hutter, 2019) with learning rate $5 \times 10^{-5}$, betas $(0.99, 0.999)$, weight decay $0.03$, ten warmup epochs, and cosine decay. The budget is 1,000 epochs, except for the 500-epoch placement/probe family. Automatic mixed precision, gradient clipping at norm 1, and an EMA with decay $0.9995$ are used. The periodic per-example loss is

$$\ell_{\mathrm{per}}(\hat{\mathbf{y}}, \mathbf{y}) = 6\,\mathrm{mean}\big[(\hat{\mathbf{y}} - \mathbf{y})^2 + |\hat{\mathbf{y}} - \mathbf{y}|\big]. \tag{14}$$

The loss is averaged over each minibatch, and EMA checkpoints are selected by IC1 validation prediction loss. Evaluation uses FP32 with TF32 disabled and batch size 4. Comparisons within each control group share precision and checkpoint selection. The output subgradient of this training loss enters Equation (6). Appendix B.2 specializes the local learning analysis to squared-loss gradient flow.

[1] Py2D code: `https://github.com/envfluids/py2d`.

### A.2 REACTION–DIFFUSION

We use the two-field PDEBench data (Takamoto et al., 2022) corresponding to `2D_diff-react_NA_NA.h5` (DaRUS file 133017), repacked from a community Parquet conversion. The activator and inhibitor have diffusion coefficients $D_u = 0.001$ and $D_v = 0.005$, with reaction parameter $k = 0.005$, on $[-1, 1]^2$ with zero-flux Neumann boundaries. Each of the 1,000 trajectories contains 101 observed frames at $128^2$, spaced by $0.05$ over $t \in [0, 5]$.

We permute trajectory indices using NumPy's `default_rng(20260909)` and assign the first 800, next 100, and final 100 to training, validation, and testing. All model seeds share this split. Channel means and standard deviations are fitted to all 101 frames of the training trajectories and held fixed. Training excludes transition $0 \to 1$. Tests start at true frame 1 and predict frames 2–100, giving 99 forecast steps. Models optimize standardized MSE. Table 1 reports errors in raw field units.

The optimizer, batch size, clipping, EMA, and warmup match the periodic settings, with a 200-epoch cosine schedule. Candidate checkpoints are completed epochs 10, 50, 100, and 200. They are selected by 20-step standardized validation rollout MSE from frames 1, 10, 30, and 50, across all 100 validation trajectories. All architectures use the same criterion. The U-Net backbone and readout use replicate padding. The default $3 \times 3$ readout maps 32 channels from the final $64^2$ decoder block to the two physical channels, then uses bilinear interpolation with `align_corners=False`.

### A.3 HETEROGENEOUS ACOUSTIC SCATTERING

The discontinuous acoustic scattering data come from The Well (Ohana et al., 2024). Its official 16/2/2 train/validation/test files each contain 100 trajectories at $256^2$, with 102 stored frames per trajectory. Inputs contain pressure, two velocity components, density, and sound speed. The network predicts pressure and velocity, and the static material maps are reattached at every autoregressive step. Stored coordinates provide positional features. Convolutions use replicate extension, and the PAIR contribution is upsampled by bilinear interpolation with `align_corners=False`.

Training includes all 101 adjacent-frame pairs from each training trajectory. Normalization is fitted once using 64 fixed training trajectories (indices 0, 25, 50, and 75 in each training file), frames 0, 25, 50, 75, and 101, and every eighth spatial sample. This subsampling estimates normalization statistics. Training and evaluation retain full spatial resolution.

All models train for 100 epochs with global batch 32. U-Net and PAIR use two GPUs with local batch 16. FNO uses one GPU with batch 32. Optimizer, warmup, EMA, and clipping match the settings above. The objective is final-output standardized MSE.

Validation considers completed epochs 10, 25, 50, 75, and 100, using all 200 validation trajectories, starts 0/10/30/50, and 20 forecast steps. Testing starts at frame 0 and predicts all 101 subsequent frames for 200 test trajectories. The reported standardized MSE averages squared physical errors divided by each dynamic channel's training variance. Checkpoint selection and evaluation therefore use a common scale across pressure and velocity.

### A.4 ARCHITECTURES AND COMPUTATION

The periodic and RD U-Nets use three decoder blocks per resolution with concatenated feature skips, two bottleneck blocks, and two-head attention at the coarsest decoder stage. Residual blocks apply GroupNorm, SiLU, and a $3 \times 3$ convolution twice, with dropout $0.2$ before the second convolution. GroupNorm uses learned affine parameters and $\epsilon = 10^{-5}$, choosing the largest group count that divides the channel count, up to 32 groups with at least four channels per group. Downsampling uses $2 \times 2$ averaging, and upsampling uses unlearnable pixel replication, each followed by a learned convolution. The output head is GroupNorm, SiLU, and a $3 \times 3$ convolution. Two uniform positional channels span $[0, 1]^2$.

The readout is an affine convolution from intermediate feature channels to physical output channels. In the default models, a randomly initialized $1 \times 1$ head supplies the center coefficients and bias of the larger readout kernel, with all off-center coefficients set to zero. Kernel comparisons therefore start from the same readout function and backbone initialization. Placement controls instead initialize every

added-head weight and bias to zero. Table 3 summarizes the periodic-flow training configurations for the main comparisons and feature probes.

Periodic Fourier transfer uses float32 `rfft2`/`irfft2` with PyTorch's default normalization. It zero-pads or crops the retained frequency blocks and scales coefficients by the target-to-source grid-area ratio. Nyquist coefficients are copied directly in native rFFT indexing, without additional halving, merging, or zeroing.

Table 3: Periodic training configurations used in the main comparisons and feature probes. All use seeds 42, 43, and 44 and IC1 training and validation data. Forecasting comparisons select checkpoints by validation loss. Feature probes use fixed training epochs. The placement row describes the intermediate head. Its matched final-layer head is $2 \times 2$ with the same 130 parameters.

| Comparison | Interface / kernel | Head initialization | Precision | Epochs |
|---|---|---|---|---|
| Main PAIR and auxiliary supervision | $64^2/3 \times 3$ | Random center | AMP | 1,000 |
| Matched-support placement | $64^2/1 \times 1$ | Zero | AMP | 1,000 |
| Gradient connectivity | $64^2/3 \times 3$ | Random center | FP32 | 1,000 |
| Feature probes and secondary placement | $128^2/3 \times 3$ | Zero | AMP | 500 |

**Matched Fourier-support placement controls.** Table 2(b) compares zero-initialized intermediate and final-layer readouts with 130 parameters each. The intermediate head maps 64 feature channels to two physical channels with a $1 \times 1$ convolution on the $64^2$ grid. The final-layer readout maps 16 final-feature channels to two physical channels with a $2 \times 2$ convolution. For the matched-support final-layer control, its $256^2$ proposal is spectrally restricted to $64^2$ and then upsampled using the same Fourier operator as the intermediate head. Both contributions are added to the original full-resolution output.

Figure 2(a) schematically illustrates the U-Net interface. The encoder and early decoder produce $\mathbf{z}$, while later skip inputs are collected in the side features $\mathbf{s}$ supplied to the full-resolution decoder.

The constant-resolution ResNet uses eight stride-one BasicBlocks (He et al., 2016), 64 channels, and a $3 \times 3$ circular readout after block four. Both added-head controls are zero-initialized. The low-pass variant masks its physical output to $|k_x| \leq 32$ and $|k_y| \leq 32$, including DC, while the latent features continue through the remaining blocks. The multiscale comparison uses a periodic ResNet18-FPN, combining a ResNet-18 encoder with a feature pyramid network (He et al., 2016; Lin et al., 2017). The encoder has base width 64, and the pyramid features have 128 channels. Heads at P5/P4/P3 read $32^2/64^2/128^2$ features, respectively, and use unlearnable Fourier upsampling to the $256^2$ output grid.

FNO uses four Fourier blocks, hidden width 48, a 128-channel projection, and positional coordinates. The $16 \times 16$ and $32 \times 32$ mode settings share the grid, task-specific training budget, normalization, global batch size 32, loss, and validation selection. Increasing the mode count increases the number of Fourier coefficients and parameters. Periodic FNO operates directly on the $256^2$ grid. For nonperiodic tasks, its lifted feature domain is padded on the right and bottom by 8 cells for RD and 16 cells for acoustics, then cropped before projection. Acoustic FNO receives the same density and sound-speed fields as U-Net and PAIR. All backbones are trained from initialization.

Timing uses one H800, common $256^2$ inputs, five warmup iterations, and twenty measured iterations. Median batch-4 FP32 inference times are 15.66 ms for U-Net and 15.78 ms for PAIR. Median batch-32 AMP forward/backward times are 109.7 and 112.1 ms, respectively. These measurements exclude data loading and optimizer updates.

## B THEORETICAL DETAILS

The output-space identities support the full-resolution path diagnostic in Section 5.4. The matched-state and finite-time analyses describe the added parameter channel under squared-loss gradient flow. We then give the approximation-capacity and fixed-feature results, followed by the rollout-error expansion used to distinguish local prediction errors from their propagation.

### B.1 Output-space identities and joint adaptation

Let $\mathcal{V} = \mathrm{Range}(\mathbf{U})$ for the unlearnable linear upsampling operator $\mathbf{U}$, and let $\mathbf{P}$ be its orthogonal projector under the vectorized field inner product, as defined in Section 4.3. Since $\mathbf{PU} = \mathbf{U}$, the direct contribution satisfies $\mathbf{P}F_{\mathrm{coarse}} = F_{\mathrm{coarse}}$. Applying $\mathbf{P}$ and $\mathbf{I} - \mathbf{P}$ to $F_{\mathrm{PAIR}} = F_{\mathrm{coarse}} + F_{\mathrm{full}}$ gives

$$\mathbf{P}F_{\mathrm{PAIR}} = F_{\mathrm{coarse}} + \mathbf{P}F_{\mathrm{full}}, \qquad (\mathbf{I} - \mathbf{P})F_{\mathrm{PAIR}} = (\mathbf{I} - \mathbf{P})F_{\mathrm{full}}. \tag{15}$$

The two projected residuals are orthogonal, so

$$\|F_{\mathrm{PAIR}} - \mathbf{y}\|^2 = \|F_{\mathrm{coarse}} + \mathbf{P}F_{\mathrm{full}} - \mathbf{P}\mathbf{y}\|^2 + \|(\mathbf{I} - \mathbf{P})(F_{\mathrm{full}} - \mathbf{y})\|^2 . \tag{16}$$

Within $\mathcal{V}$, $F_{\mathrm{coarse}}$ and $\mathbf{P}F_{\mathrm{full}}$ may reinforce or cancel each other. Neither contribution is assigned a separate target.

Because $F_{\mathrm{coarse}} = \mathbf{U}q_{\boldsymbol{\phi}}(\mathbf{z})$ and $\mathbf{U}$ is unlearnable and linear, $\mathbf{P}\mathbf{J}_{\mathrm{coarse}} = \mathbf{J}_{\mathrm{coarse}}$. For $\mathbf{g} = \nabla_{\widehat{\mathbf{y}}}\ell(\widehat{\mathbf{y}}, \mathbf{y})$, the chain rule with the side features $\mathbf{s}$ and static inputs held fixed gives

$$\nabla_{\mathbf{z}}\ell = (\mathbf{J}_{\mathrm{coarse}} + \mathbf{J}_{\mathrm{full}})^\top \mathbf{g} = \mathbf{J}_{\mathrm{coarse}}^\top \mathbf{P}\mathbf{g} + \mathbf{J}_{\mathrm{full}}^\top \mathbf{g}.$$

This recovers Equation (6) and shows that the added route depends only on the projected output gradient $\mathbf{P}\mathbf{g}$. Upstream parameter gradients also contain the dependence through the side features $\mathbf{s}$.

For independently trained predictors, adding and subtracting $F^\star_{\mathrm{base}}$ gives

$$F^\star_{\mathrm{PAIR}} - F^\star_{\mathrm{base}} = F^\star_{\mathrm{coarse}} + (F^\star_{\mathrm{full}} - F^\star_{\mathrm{base}}) . \tag{17}$$

Applying $\mathbf{I} - \mathbf{P}$ removes $F^\star_{\mathrm{coarse}}$ and gives Equation (10).

### B.2 Matched-state tangent contribution

Stack predictions on a fixed batch and let $\mathbf{e}$ be the common residual at the matched zero-readout state. With all readout weights and biases set to zero, $F_{\mathrm{coarse}} = 0$ and $\partial F_{\mathrm{coarse}}/\partial \boldsymbol{\psi} = 0$. The prediction and the Jacobian with respect to the baseline parameters $\boldsymbol{\psi}$ therefore equal those of $F_{\mathrm{base}}$. The augmented Jacobian is $[\mathbf{J}_{\mathrm{base}}, \mathbf{J}_{\boldsymbol{\phi}}]$, and

$$\mathbf{K}_{\mathrm{PAIR}} = [\mathbf{J}_{\mathrm{base}}, \mathbf{J}_{\boldsymbol{\phi}}][\mathbf{J}_{\mathrm{base}}, \mathbf{J}_{\boldsymbol{\phi}}]^\top = \mathbf{K}_{\mathrm{base}} + \mathbf{K}_{\mathrm{coarse}},$$

which gives Equation (7). On stacked batch outputs, $\mathbf{P}$ acts block-diagonally, applying the field projector to each example. Since $\mathbf{P}\mathbf{J}_{\boldsymbol{\phi}} = \mathbf{J}_{\boldsymbol{\phi}}$, the added tangent block satisfies $\mathbf{K}_{\mathrm{coarse}} = \mathbf{P}\mathbf{K}_{\mathrm{coarse}}\mathbf{P}$. For $L_m = \frac{1}{2}\|F_m - \mathbf{y}\|^2$, $m \in \{\mathrm{base}, \mathrm{PAIR}\}$, unit-rate squared-loss gradient flow gives $\dot{\mathbf{e}} = -\mathbf{K}\mathbf{e}$ and $\dot{L} = -\mathbf{e}^\top \mathbf{K}\mathbf{e}$. At the matched state,

$$\dot{L}_{\mathrm{PAIR}} - \dot{L}_{\mathrm{base}} = -\mathbf{e}^\top \mathbf{K}_{\mathrm{coarse}}\mathbf{e} = -\left\|\mathbf{J}_{\boldsymbol{\phi}}^\top \mathbf{e}\right\|^2 \le 0. \tag{18}$$

The additional term updates $\boldsymbol{\phi}$ at the matched state. As the readout grows, Equation (6) couples the added channel to the shared state.

For a distribution-level form, write the static problem-specific inputs explicitly as $\boldsymbol{\xi}$ and let $\mathbf{x} = (\mathbf{u}, \boldsymbol{\xi})$. Let $(\mathbf{x}, \mathbf{y})$ follow the training distribution. At training time $\tau$, define

$$\mathbf{e}_\tau(\mathbf{x}, \mathbf{y}) = F_{\boldsymbol{\omega}(\tau)}(\mathbf{u}; \boldsymbol{\xi}) - \mathbf{y}, \qquad \mathbf{J}_\tau(\mathbf{x}) = \left.\frac{\partial F_{\boldsymbol{\omega}}(\mathbf{u}; \boldsymbol{\xi})}{\partial \boldsymbol{\omega}}\right|_{\boldsymbol{\omega} = \boldsymbol{\omega}(\tau)} .$$

Under squared-loss gradient flow, the population tangent kernel $\mathbf{K}_\tau(\mathbf{x}, \mathbf{x}') = \mathbf{J}_\tau(\mathbf{x})\mathbf{J}_\tau(\mathbf{x}')^\top$ gives

$$\partial_\tau \mathbf{e}_\tau(\mathbf{x}, \mathbf{y}) = -\mathbb{E}_{(\mathbf{x}', \mathbf{y}')}[\mathbf{K}_\tau(\mathbf{x}, \mathbf{x}')\mathbf{e}_\tau(\mathbf{x}', \mathbf{y}')].$$

The kernel couples residuals across inputs, while the added readout's spatial output support is determined by $\mathcal{V}$.

### B.3 FINITE-TIME RESULT UNDER COMMUTING TANGENT DYNAMICS

We prove Theorem 1. Commuting positive-semidefinite matrices $\mathbf{K}_{\text{base}}$ and $\mathbf{K}_{\text{coarse}}$ admit a common orthonormal eigenbasis $\{\mathbf{v}_j\}$. Along $\mathbf{v}_j$, the residual dynamics are

$$\dot{e}_{\text{base},j} = -\mu_j^{\text{base}} e_{\text{base},j}, \qquad \dot{e}_{\text{PAIR},j} = -(\mu_j^{\text{base}} + \mu_j^{\text{coarse}}) e_{\text{PAIR},j}.$$

Starting from the same $e_j(0)$ gives

$$e_{\text{base},j}(\tau) = e^{-\mu_j^{\text{base}}\tau} e_j(0), \qquad e_{\text{PAIR},j}(\tau) = e^{-(\mu_j^{\text{base}}+\mu_j^{\text{coarse}})\tau} e_j(0).$$

Taking squared magnitudes proves Equation (9). Define $\Lambda_{\mathbf{U}} = \{j : \mu_j^{\text{coarse}} > 0\}$. Summing the coordinate-wise reduction gives

$$\|\mathbf{e}_{\text{base}}(\tau)\|^2 - \|\mathbf{e}_{\text{PAIR}}(\tau)\|^2 = \sum_{j\in\Lambda_{\mathbf{U}}} |e_j(0)|^2 e^{-2\mu_j^{\text{base}}\tau} \left(1 - e^{-2\mu_j^{\text{coarse}}\tau}\right) \geq 0. \tag{19}$$

With Fourier upsampling, $\mathcal{V}$ is a coarse Fourier-support subspace. Under an additional Fourier-mode-decoupled approximation, positive $\mu_j^{\text{coarse}}$ correspond to supported low-to-intermediate wavenumbers. In general, the tangent directions may mix multiple modes within this support.

### B.4 UNIVERSAL OPERATOR APPROXIMATION

**Theorem 2** (Universal operator approximation)**.** *Let $\mathcal{X}$ and $\mathcal{Y}$ be normed spaces and let $\mathcal{K} \subset \mathcal{X}$ be compact. If $\mathcal{F}_{\text{base}}$ is dense in $C(\mathcal{K};\mathcal{Y})$ and the readout can represent the zero map, then $\mathcal{F}_{\text{PAIR}}$ is dense in $C(\mathcal{K};\mathcal{Y})$.*

*Proof.* Write $\mathbf{x} = (\mathbf{u}, \boldsymbol{\xi})$ and $\boldsymbol{\psi} = (\boldsymbol{\eta}, \boldsymbol{\theta})$, and let $\mathbf{s}_{\boldsymbol{\eta}}(\mathbf{x})$ collect the skip or pyramid features supplied to the remaining backbone layers. A member of the backbone family is

$$F_{\text{base},\boldsymbol{\psi}}(\mathbf{x}) = D_{\boldsymbol{\theta}}(E_{\boldsymbol{\eta}}(\mathbf{x}); \mathbf{s}_{\boldsymbol{\eta}}(\mathbf{x}))\,,$$

and the corresponding PAIR family is

$$\mathcal{F}_{\text{PAIR}} = \{\mathbf{x} \mapsto F_{\text{base},\boldsymbol{\psi}}(\mathbf{x}) + \mathbf{U} q_{\boldsymbol{\phi}}(E_{\boldsymbol{\eta}}(\mathbf{x})) : \boldsymbol{\psi} \in \Theta_{\text{base}},\ \boldsymbol{\phi} \in \Theta_{\text{coarse}}\}\,.$$

Here $\Theta_{\text{base}}$ and $\Theta_{\text{coarse}}$ denote the backbone and readout parameter sets. If $q_{\boldsymbol{\phi}_0} \equiv 0$, every $F_{\text{base},\boldsymbol{\psi}} \in \mathcal{F}_{\text{base}}$ is represented in $\mathcal{F}_{\text{PAIR}}$ by $(\boldsymbol{\psi}, \boldsymbol{\phi}_0)$. For every $G \in C(\mathcal{K};\mathcal{Y})$ and $\varepsilon > 0$, density of $\mathcal{F}_{\text{base}}$ supplies $F_{\text{base},\boldsymbol{\psi}}$ such that

$$\sup_{\mathbf{x}\in\mathcal{K}} \|G(\mathbf{x}) - F_{\text{base},\boldsymbol{\psi}}(\mathbf{x})\|_{\mathcal{Y}} < \varepsilon.$$

Choosing $\boldsymbol{\phi} = \boldsymbol{\phi}_0$ gives the same approximation in $\mathcal{F}_{\text{PAIR}}$. □

### B.5 FIXED-FEATURE INCREMENTAL READOUT

In the Hilbert space of square-integrable predictions, let $\mathcal{F}_{\text{full}}$ and $\mathcal{F}_{\text{coarse}}$ be finite-dimensional linear prediction spaces generated by fixed final-layer and intermediate features. For a target random variable $\mathbf{Y}$, define

$$R^*_{\text{full}} = \inf_{b\in\mathcal{F}_{\text{full}}} \mathbb{E}\,\|\mathbf{Y} - b\|^2\,, \qquad R^*_{\text{coarse+full}} = \inf_{a\in\mathcal{F}_{\text{coarse}},\ b\in\mathcal{F}_{\text{full}}} \mathbb{E}\,\|\mathbf{Y} - a - b\|^2\,.$$

Let $\Pi_{\text{full}}$ project onto $\mathcal{F}_{\text{full}}$, set $\boldsymbol{\rho}_{\text{full}} = (\mathbf{I} - \Pi_{\text{full}})\mathbf{Y}$, and define $\mathcal{F}_{\text{coarse|full}} = (\mathbf{I} - \Pi_{\text{full}})\mathcal{F}_{\text{coarse}}$. Since

$$a + b = (b + \Pi_{\text{full}} a) + (\mathbf{I} - \Pi_{\text{full}})a,$$

the sum space decomposes orthogonally as $\mathcal{F}_{\text{coarse}} + \mathcal{F}_{\text{full}} = \mathcal{F}_{\text{full}} \oplus \mathcal{F}_{\text{coarse|full}}$. If $\Pi_{\text{coarse|full}}$ denotes projection onto $\mathcal{F}_{\text{coarse|full}}$, Pythagoras gives

$$R^*_{\text{full}} - R^*_{\text{coarse+full}} = \left\|\Pi_{\text{coarse|full}}\boldsymbol{\rho}_{\text{full}}\right\|^2_{L^2}\,.$$

Appendix D.2 compares fitted final-layer and joint readouts on frozen features. These probes use validation-selected ridge regularization and are evaluated on held-out inputs.

### B.6 A SCALAR LEARNING EXAMPLE

Let $\widehat{y}_{\text{base}} = bz$ and $\widehat{y}_{\text{PAIR}} = (b + a)z$, hold $b$ fixed, and initialize $a(0) = 0$. Starting from the same $z_0$ and residual $e_{\text{init}} = bz_0 - y$, squared-loss gradient flow with rates $\alpha_z, \alpha_a > 0$ gives

$$\dot{e}_{\text{base}}(0) = -\alpha_z b^2 e_{\text{init}}, \qquad \dot{e}_{\text{PAIR}}(0) = -(\alpha_z b^2 + \alpha_a z_0^2) e_{\text{init}}.$$

The additional instantaneous term comes from updating $a$, although the new route's derivative with respect to $z$ vanishes at $a = 0$.

### B.7 ERROR PROPAGATION OVER LONG HORIZONS

For fixed learned parameters and static inputs $\boldsymbol{\xi}$, define the rollout error and true-input one-step error along a reference trajectory by

$$\mathbf{r}_t = \widehat{\mathbf{u}}_t - \mathbf{u}_t, \qquad \boldsymbol{\epsilon}_t = F_{\boldsymbol{\omega}}(\mathbf{u}_t; \boldsymbol{\xi}) - \mathbf{u}_{t+1}.$$

Using $\widehat{\mathbf{u}}_t = \mathbf{u}_t + \mathbf{r}_t$ and adding and subtracting $F_{\boldsymbol{\omega}}(\mathbf{u}_t; \boldsymbol{\xi})$ gives

$$\mathbf{r}_{t+1} = F_{\boldsymbol{\omega}}(\mathbf{u}_t + \mathbf{r}_t; \boldsymbol{\xi}) - F_{\boldsymbol{\omega}}(\mathbf{u}_t; \boldsymbol{\xi}) + \boldsymbol{\epsilon}_t.$$

Let $\mathbf{M}_t = D_{\mathbf{u}} F_{\boldsymbol{\omega}}(\mathbf{u}_t; \boldsymbol{\xi})$ and define $\boldsymbol{\Phi}_{T,t} = \mathbf{M}_{T-1} \cdots \mathbf{M}_t$ for $0 \le t < T$, with $\boldsymbol{\Phi}_{T,T} = \mathbf{I}$. A first-order expansion about $\mathbf{u}_t$, starting from $\mathbf{r}_0 = 0$, yields

$$\mathbf{r}_{t+1} \approx \mathbf{M}_t \mathbf{r}_t + \boldsymbol{\epsilon}_t, \qquad \mathbf{r}_T \approx \sum_{t=0}^{T-1} \boldsymbol{\Phi}_{T,t+1} \boldsymbol{\epsilon}_t. \tag{20}$$

The corresponding squared error contains individual and cross-time terms,

$$\|\mathbf{r}_T\|^2 \approx \sum_t \|\boldsymbol{\Phi}_{T,t+1} \boldsymbol{\epsilon}_t\|^2 + 2 \sum_{t' < t} \langle \boldsymbol{\Phi}_{T,t'+1} \boldsymbol{\epsilon}_{t'}, \boldsymbol{\Phi}_{T,t+1} \boldsymbol{\epsilon}_t \rangle .$$

True-input evaluation measures $\boldsymbol{\epsilon}_t$, whereas free rollout measures the accumulated error $\mathbf{r}_t$. The cross-time terms also depend on how successive local errors align after propagation.

For a time-invariant propagation model diagonal in the spatial Fourier basis, let $\lambda_k$ denote the propagation factor at mode $k$. If the injected errors are zero mean, uncorrelated across time, and have constant variance $\sigma_k^2$, then

$$\mathbb{E}|r_T(k)|^2 \approx \sigma_k^2 \sum_{j=0}^{T-1} |\lambda_k|^{2j}.$$

An initial perturbation $\boldsymbol{\delta}$ contributes the additional term $\boldsymbol{\Phi}_{T,0} \boldsymbol{\delta}$ to the linearized forecast error. Appendix F evaluates finite responses directly from paired clean and perturbed trajectories for both learned and numerical propagators.

## C SUPERVISION AND ARCHITECTURE CONTROLS

Bold method names identify PAIR. Bold numerical entries mark the lowest reported mean error within each comparison group.

### C.1 INTERMEDIATE AUXILIARY SUPERVISION

This control asks whether a separately supervised intermediate prediction yields the same benefit as an coarse contribution trained through the final sum. It uses the default U-Net, its last $64^2$ decoder state, and the same affine $3 \times 3$ head and parameter initialization as PAIR. Both losses update the shared features. With $\mathbf{R}$ denoting Fourier restriction to the intermediate grid, the training objective is

$$\mathcal{L}_{\text{aux}} = \ell(F_{\text{full}}, \mathbf{y}) + \lambda \ell(q_{\boldsymbol{\phi}}(\mathbf{z}), \mathbf{R}\mathbf{y}).$$

Each term uses the periodic loss in Appendix A, including its factor of six. Validation and inference use the full-resolution output $F_{\text{full}}$.

Three seed-42 runs compare $\lambda \in \{0.1, 1, 10\}$ under the 1,000-epoch default AMP protocol. IC1 validation loss of the EMA predictor $F_{\text{full}}$ selects $\lambda = 0.1$. Two further runs use this weight with seeds 43 and 44. The selected models are evaluated on the common 100-start IC2 test, giving the comparison in Table 2(a). Table 4 records validation selection and the individual candidate results.

Table 4: Auxiliary-weight selection with seed 42. IC1 validation loss selects $\lambda = 0.1$. The IC2 columns report periodic nMSE for the corresponding selected checkpoints. These candidate results each use one training seed.

| $\lambda$ | IC1 validation loss | Selected epoch | IC2 step 100 | IC2 time average |
|---|---|---|---|---|
| 0.1 | 0.012235 | 920 | 0.014368 | 0.004036 |
| 1 | 0.012724 | 900 | 0.014606 | 0.004114 |
| 10 | 0.015250 | 980 | 0.021517 | 0.007085 |

### C.2 Readout placement

Readout placement changes the features supplied to the head and the gradient route to the backbone. The comparison in Table 2(b) controls the added parameter count and output support using the heads in Appendix A.4. Each head has 130 parameters and starts at zero, recovering the same initial U-Net prediction. Training lasts 1,000 epochs. Table 5 provides the early and intermediate forecast errors.

Table 5: **Readout placement at early and intermediate horizons.** Periodic nMSE, mean $\pm$ training-seed SD, for the zero-initialized configurations in Table 2(b). Endpoint and time-averaged errors appear in Table 2(b).

| Model | Step 1 ($\times 10^{-7}$) | Step 10 ($\times 10^{-5}$) | Step 50 ($\times 10^{-3}$) |
|---|---|---|---|
| U-Net | $3.431 \pm 0.113$ | $4.347 \pm 0.334$ | $2.083 \pm 0.214$ |
| Final-layer readout | $3.349 \pm 0.249$ | $4.409 \pm 0.234$ | $2.225 \pm 0.208$ |
| Final-layer, matched support | $3.186 \pm 0.255$ | $2.683 \pm 0.086$ | $1.089 \pm 0.030$ |
| Intermediate $64^2$ | $\mathbf{2.492 \pm 0.082}$ | $\mathbf{1.619 \pm 0.059}$ | $\mathbf{0.671 \pm 0.012}$ |

A second placement family uses 500 epochs and a zero-initialized $128^2$ intermediate readout. Its 100-step nMSE improves by 62.4% over its matched-budget baseline. Final-layer $3 \times 3$ and $5 \times 5$ readouts give higher endpoint errors than the intermediate readout. The final-layer $5 \times 5$ head has more parameters than the intermediate head. The frozen-feature probes in Appendix D.2 use this configuration.

### C.3 Gradient control at the added readout

To test the added gradient connection, we detach the input to the default $64^2/3 \times 3$ head. The forward prediction remains $F_{\text{coarse}} + F_{\text{full}}$, and the head parameters continue to receive gradients. The full-resolution path still updates the shared features through the decoder. Both configurations use FP32, seeds 42/43/44, 1,000 epochs, identical data and validation selection, and the full 100-start IC2 test. Table 2(c) reports endpoint and time-averaged errors. Table 6 shows earlier horizons. The default main models use AMP, as recorded in Table 3.

Table 6: **Gradient connectivity at early and intermediate horizons.** Periodic nMSE for the matched FP32 configurations in Table 2(c). Means are shown above their training-seed SD. Endpoint and time-averaged errors appear in Table 2(c).

| Training path | Step 1 | Step 10 | Step 50 |
|---|---|---|---|
| **Joint training** | $\mathbf{1.458 \times 10^{-7}}$ $\pm 5.079 \times 10^{-9}$ | $\mathbf{6.015 \times 10^{-6}}$ $\pm 4.606 \times 10^{-7}$ | $\mathbf{0.0003123}$ $\pm 3.428 \times 10^{-5}$ |
| Stop gradient through $F_{\text{coarse}}$ | $1.684 \times 10^{-6}$ $\pm 6.23 \times 10^{-7}$ | $0.0001474$ $\pm 1.829 \times 10^{-5}$ | $0.004801$ $\pm 0.0004875$ |

### C.4 Readout, upsampling, and full-resolution path controls

We vary the information used by the added head, the transfer to the output grid, and the presence of the full-resolution decoder in separate comparisons. The kernel sweep keeps the interface at $64^2$ and uses Fourier upsampling throughout.

Table 7: Readout kernel size on periodic flow. $64^2$ interface, Fourier upsampling, 1,000 epochs. Periodic nMSE, mean ± seed SD.

| Readout kernel | Step 1 ($\times 10^{-7}$) | Step 10 ($\times 10^{-5}$) | Step 50 ($\times 10^{-3}$) | Step 100 ($\times 10^{-2}$) |
|---|---|---|---|---|
| $1 \times 1$ | $2.124 \pm 0.059$ | $1.370 \pm 0.061$ | $0.596 \pm 0.031$ | $0.617 \pm 0.008$ |
| $3 \times 3$ | $1.465 \pm 0.050$ | $0.607 \pm 0.052$ | $0.312 \pm 0.041$ | $0.380 \pm 0.035$ |
| $5 \times 5$ | $1.211 \pm 0.018$ | $0.459 \pm 0.028$ | $0.244 \pm 0.029$ | $0.299 \pm 0.015$ |
| $7 \times 7$ | $1.088 \pm 0.046$ | $0.379 \pm 0.015$ | $0.211 \pm 0.008$ | $0.271 \pm 0.012$ |
| $9 \times 9$ | $1.026 \pm 0.065$ | $0.341 \pm 0.027$ | $0.186 \pm 0.018$ | $0.253 \pm 0.013$ |
| $11 \times 11$ | $0.961 \pm 0.062$ | $0.314 \pm 0.019$ | $0.176 \pm 0.018$ | $0.236 \pm 0.006$ |
| $13 \times 13$ | $0.922 \pm 0.057$ | $0.291 \pm 0.025$ | $0.152 \pm 0.014$ | $0.218 \pm 0.003$ |
| $15 \times 15$ | $0.884 \pm 0.070$ | $0.271 \pm 0.038$ | $0.134 \pm 0.005$ | $0.199 \pm 0.009$ |
| $31 \times 31$ | $0.853 \pm 0.020$ | $0.227 \pm 0.013$ | $0.113 \pm 0.009$ | $0.164 \pm 0.018$ |
| $63 \times 63$ | $\mathbf{0.815} \pm 0.045$ | $\mathbf{0.203} \pm 0.013$ | $\mathbf{0.089} \pm 0.012$ | $\mathbf{0.150} \pm 0.020$ |

Larger dense readouts lower endpoint error across the sweep in Table 7. A sparse $3 \times 3$ head with dilation 15 has the same nominal $31 \times 31$ span as a dense head but samples nine positions. Its 100-step nMSE is 0.006203, above the dense $31 \times 31$ result. Thus, matching the spatial span does not reproduce the performance of dense sampling.

To examine the mapping from features to physical channels, we also use a readout that directly selects two channels from the trainable upstream representation. Its 100-step nMSE is 0.006325, higher than those of the learned dense readouts in Table 7.

Table 8: Band-limited and bilinear upsampling of the coarse prediction, with a $3 \times 3$ readout. Band-limited upsampling uses Fourier zero-padding and confines $F_{\mathrm{coarse}}$ to the coarse grid's Fourier support. $F_{\mathrm{full}}$ remains unrestricted. Each pair uses the same interface resolution and training protocol. Bold values compare upsampling methods within each resolution. Periodic nMSE.

| Interface / interpolation | Step 1 ($\times 10^{-7}$) | Step 10 ($\times 10^{-5}$) | Step 50 ($\times 10^{-3}$) | Step 100 ($\times 10^{-2}$) |
|---|---|---|---|---|
| $32^2$, Band-limited | $\mathbf{1.767} \pm 0.021$ | $\mathbf{0.781} \pm 0.034$ | $\mathbf{0.337} \pm 0.011$ | $\mathbf{0.419} \pm 0.013$ |
| $32^2$, Bilinear | $1.983 \pm 0.046$ | $0.967 \pm 0.015$ | $0.394 \pm 0.010$ | $0.455 \pm 0.006$ |
| $64^2$, Band-limited | $\mathbf{1.465} \pm 0.050$ | $\mathbf{0.607} \pm 0.052$ | $\mathbf{0.312} \pm 0.041$ | $\mathbf{0.380} \pm 0.035$ |
| $64^2$, Bilinear | $1.658 \pm 0.065$ | $0.666 \pm 0.037$ | $0.328 \pm 0.019$ | $0.398 \pm 0.038$ |
| $128^2$, Band-limited | $\mathbf{1.540} \pm 0.086$ | $\mathbf{0.708} \pm 0.057$ | $\mathbf{0.369} \pm 0.013$ | $\mathbf{0.427} \pm 0.011$ |
| $128^2$, Bilinear | $1.597 \pm 0.072$ | $0.709 \pm 0.079$ | $0.372 \pm 0.036$ | $0.434 \pm 0.036$ |

Changing the transfer tests whether the improvement requires a sharp Fourier cutoff. At $64^2/3 \times 3$, bilinear upsampling gives endpoint nMSE 0.003979 versus 0.003801 for band-limited upsampling (Table 8).

The direct-only models train the intermediate path as the entire predictor. Computation after the selected interface and the original output head are removed, and the upsampled intermediate prediction is supervised against the full target. Table 9 compares each interface with a separately trained full PAIR model.

At $32^2$ and $64^2$, the full PAIR model has lower error at every reported horizon. At $128^2$, PAIR has lower step-1 error, while direct-only prediction has lower errors at steps 10, 50, and 100.

### C.5 Constant-resolution ResNet and ResNet18-FPN

The ResNet comparison keeps the feature grid fixed and varies the output bandwidth of the added head. The ResNet18-FPN comparison uses intermediate features from a spatial pyramid. Both use the backbone definitions in Appendix A and the same periodic test set.

Filtering $F_{\mathrm{coarse}}$ reduces mean step-100 field nMSE by 4.60% relative to the unrestricted readout (Table 10). The gain comes from the lowest frequency shells. In the unrestricted model, roughly 88%

Table 9: Full and separately trained direct-only models at different interface resolutions. Bold values compare models within each resolution. Periodic nMSE, $3 \times 3$ readouts.

| Interface / model | Step 1 ($\times 10^{-7}$) | Step 10 ($\times 10^{-5}$) | Step 50 ($\times 10^{-3}$) | Step 100 ($\times 10^{-2}$) |
|---|---|---|---|---|
| $32^2$, **PAIR** | $\mathbf{1.767} \pm 0.021$ | $\mathbf{0.781} \pm 0.034$ | $\mathbf{0.337} \pm 0.011$ | $\mathbf{0.419} \pm 0.013$ |
| $32^2$, direct only | $265.924 \pm 0.259$ | $6.311 \pm 0.159$ | $1.180 \pm 0.077$ | $0.962 \pm 0.043$ |
| $64^2$, **PAIR** | $\mathbf{1.465} \pm 0.050$ | $\mathbf{0.607} \pm 0.052$ | $\mathbf{0.312} \pm 0.041$ | $\mathbf{0.380} \pm 0.035$ |
| $64^2$, direct only | $32.222 \pm 0.124$ | $1.205 \pm 0.062$ | $0.432 \pm 0.049$ | $0.498 \pm 0.036$ |
| $128^2$, **PAIR** | $\mathbf{1.540} \pm 0.086$ | $0.708 \pm 0.057$ | $0.369 \pm 0.013$ | $0.427 \pm 0.011$ |
| $128^2$, direct only | $3.111 \pm 0.015$ | $\mathbf{0.662} \pm 0.007$ | $\mathbf{0.351} \pm 0.005$ | $\mathbf{0.396} \pm 0.009$ |

Table 10: Constant-resolution ResNet readout controls. The low-pass acts only on $F_{\mathrm{coarse}}$. Periodic nMSE, three seeds, 1,000 epochs.

| Model | Step 1 ($\times 10^{-6}$) | Step 10 ($\times 10^{-3}$) | Step 50 ($\times 10^{-2}$) | Step 100 ($\times 10^{-2}$) |
|---|---|---|---|---|
| ResNet | $7.713 \pm 0.423$ | $1.344 \pm 0.176$ | $2.079 \pm 0.114$ | $2.913 \pm 0.184$ |
| Unrestricted readout | $\mathbf{7.653} \pm 0.406$ | $1.345 \pm 0.151$ | $2.067 \pm 0.110$ | $2.904 \pm 0.055$ |
| Low-pass readout | $7.820 \pm 0.558$ | $\mathbf{1.254} \pm 0.016$ | $\mathbf{1.958} \pm 0.022$ | $\mathbf{2.771} \pm 0.119$ |

of the step-100 raw error lies in shells 1–4, where the improvement outweighs the deterioration in shells 5–20.

Over shells 0–20, the ratio of seed-averaged raw MSEs gives a 4.43% reduction. The effect varies across training seeds: error decreases by 3.88% and 10.26% for seeds 42 and 44, but increases by 1.05% for seed 43.

On ResNet18-FPN, a $64^2/15 \times 15$ readout lowers the 100-step nMSE by 25.2% relative to the backbone (Table 11). We also vary the attachment level with a $3 \times 3$ head to examine how the feature hierarchy affects the result.

Table 11: PAIR on ResNet18-FPN with a $64^2$ interface and $15 \times 15$ readout. Both models use the same 1,000-epoch protocol and full 100-start periodic test. Periodic nMSE, mean $\pm$ seed SD.

| Model | Step 1 ($\times 10^{-6}$) | Step 10 ($\times 10^{-4}$) | Step 50 ($\times 10^{-3}$) | Step 100 ($\times 10^{-2}$) |
|---|---|---|---|---|
| ResNet18-FPN | $2.675 \pm 0.229$ | $2.121 \pm 0.075$ | $8.301 \pm 0.284$ | $2.443 \pm 0.044$ |
| ResNet18-FPN + **PAIR** | $\mathbf{1.002} \pm 0.100$ | $\mathbf{0.830} \pm 0.102$ | $\mathbf{3.958} \pm 0.423$ | $\mathbf{1.828} \pm 0.085$ |

Table 12: ResNet18-FPN $3 \times 3$ readout position controls. All direct outputs use Fourier upsampling. Periodic nMSE.

| Interface / kernel | Step 1 ($\times 10^{-6}$) | Step 10 ($\times 10^{-4}$) | Step 50 ($\times 10^{-3}$) | Step 100 ($\times 10^{-2}$) |
|---|---|---|---|---|
| $32^2$, $3 \times 3$ | $\mathbf{1.907} \pm 0.008$ | $1.738 \pm 0.026$ | $7.383 \pm 0.262$ | $2.319 \pm 0.051$ |
| $64^2$, $3 \times 3$ | $1.937 \pm 0.106$ | $\mathbf{1.683} \pm 0.067$ | $\mathbf{7.090} \pm 0.241$ | $\mathbf{2.307} \pm 0.045$ |
| $128^2$, $3 \times 3$ | $2.295 \pm 0.040$ | $1.927 \pm 0.077$ | $7.867 \pm 0.266$ | $2.364 \pm 0.011$ |

In Table 12, the $64^2$ position has the lowest mean multistep error among the three configurations, with a small late difference from $32^2$. Changing the pyramid level changes both the features read by the head and its output support.

# D FULL-RESOLUTION PATH DIAGNOSTICS AND REPRESENTATION PROBES

## D.1 TRUE-INPUT PREDICTION OUTSIDE THE ADDED SUPPORT

We select frequencies where the added contribution is identically zero, so the measured output comes from the full-resolution path. On a $256^2$ output grid, the $64^2$ Fourier head has coordinate-wise support with corner radius about 45.3. We therefore evaluate the radial band $k \geq 64$. A $128^2$ head has corner radius about 90.5, for which we use $k \geq 96$. Both follow the native Nyquist convention. Fourier normalization is specified in Appendix E.

For every formal IC2 test start, each model receives the reference states at offsets $0, 9, 49, 99$ and predicts one subsequent step. The 100 windows give 400 input–target pairs per seed and 362 distinct inputs. Averaging the 100 pairs at each offset and then the training seeds gives Table 2(d). IC2 comes from an initial condition distinct from IC1 training and validation.

The same diagnostic on 32 common IC1 validation pairs lowers error power in $k \geq 64$ from $5.028 \times 10^{-6}$ for U-Net to $1.814 \times 10^{-6}$ for the default PAIR model. With a $128^2$ interface, error power in $k \geq 96$ decreases by 65.6%.

## D.2 COMMON-READOUT PROBES

The probes compare predictive information in frozen features under a common readout-fitting procedure. We use the 500-epoch placement configuration in Appendix C.2, freezing U-Net and PAIR at epochs 50, 100, 200, and 500. At each checkpoint, linear $3 \times 3$ readouts are fitted to reconstruct the full reference field, using either final-layer features alone or final-layer and $128^2$ intermediate features together. Fits use 128 IC1 training inputs and 64 IC1 validation inputs. A common ridge coefficient, $10^{-8}$, is selected by mean validation error across the candidate fits, then all probes are evaluated on the same 256 IC2 inputs.

Table 13: Held-out raw MSE of readouts fitted to frozen features. Means are shown above their training-seed SD. Bold compares U-Net and PAIR features within each readout type and epoch. The final-layer probe is refitted, as are both heads in the joint probe.

| | Final-layer readout | | Joint readouts | |
|---|---|---|---|---|
| Epoch | U-Net features | PAIR features | U-Net features | PAIR features |
| 50 | $\mathbf{0.0001004}$ $\pm 2.688 \times 10^{-6}$ | $0.005666$ $\pm 0.0002201$ | $9.654 \times 10^{-5}$ $\pm 2.744 \times 10^{-6}$ | $\mathbf{6.338 \times 10^{-5}}$ $\pm 3.557 \times 10^{-6}$ |
| 100 | $\mathbf{6.356 \times 10^{-5}}$ $\pm 2.687 \times 10^{-6}$ | $0.005759$ $\pm 0.0003283$ | $6.17 \times 10^{-5}$ $\pm 2.658 \times 10^{-6}$ | $\mathbf{3.77 \times 10^{-5}}$ $\pm 1.339 \times 10^{-6}$ |
| 200 | $\mathbf{4.387 \times 10^{-5}}$ $\pm 2.729 \times 10^{-6}$ | $0.005975$ $\pm 0.0003601$ | $4.268 \times 10^{-5}$ $\pm 2.711 \times 10^{-6}$ | $\mathbf{2.449 \times 10^{-5}}$ $\pm 6.114 \times 10^{-7}$ |
| 500 | $\mathbf{3.17 \times 10^{-5}}$ $\pm 1.719 \times 10^{-6}$ | $0.006303$ $\pm 0.0003826$ | $3.08 \times 10^{-5}$ $\pm 1.74 \times 10^{-6}$ | $\mathbf{1.682 \times 10^{-5}}$ $\pm 3.133 \times 10^{-7}$ |

Joint probes fitted to PAIR features have lower held-out error at every measured checkpoint (Table 13). At epoch 500, their MSE is $1.682 \times 10^{-5}$, compared with $3.080 \times 10^{-5}$ for U-Net features. Final-layer probes give the opposite ordering, with errors of $6.303 \times 10^{-3}$ for PAIR features and $3.170 \times 10^{-5}$ for U-Net features. Under this linear fitting procedure, the combined PAIR features predict the full target more accurately, while its final-layer features alone are less predictive. This pattern is consistent with a redistribution of predictive information across depth. Appendix B.5 gives the unregularized fixed-feature projection identity motivating this comparison.

## D.3 CONTRIBUTIONS LEARNED BY THE TWO PATHS

The contribution spectra use the default $64^2/3 \times 3$ PAIR on 32 common IC1 validation inputs. Both $F_{\text{coarse}}$ and $F_{\text{full}}$ are taken from the same forward pass, and their spectra are averaged over three seeds. Both contain low-frequency energy (Figure 8), consistent with their overlapping output spaces in Section 4.3. The energy of $F_{\text{coarse}} + F_{\text{full}}$ also includes cross terms between the two contributions.

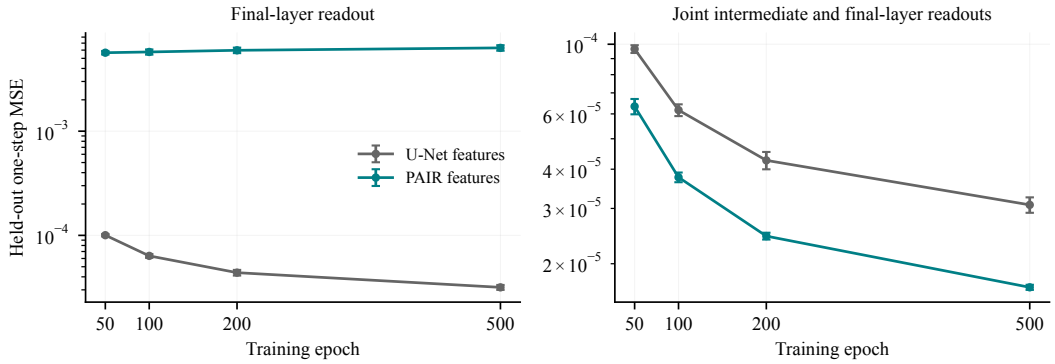



Figure 7: **Predictive information in frozen features.** Identical final-layer probes (left) and joint probes (right) are fitted to the two trained models in the 500-epoch $128^2$ configuration. Error bars show training-seed SD on held-out IC2 inputs. Both panels use logarithmic vertical axes.

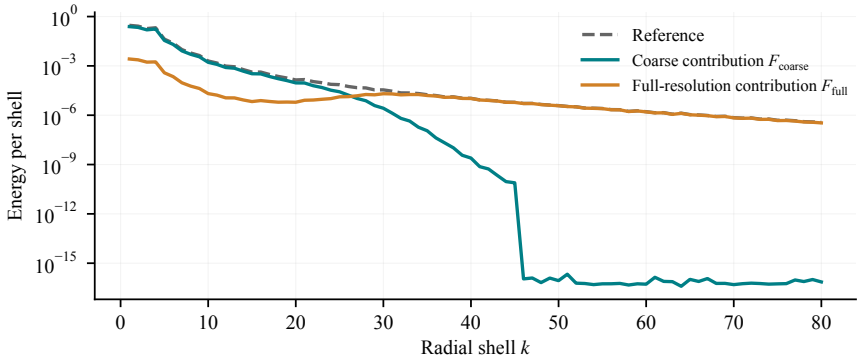



Figure 8: **Energy spectra of the learned contributions.** Coarse contribution $F_{\text{coarse}}$, full-resolution contribution $F_{\text{full}}$, and the reference target on 32 common IC1 inputs. Curves average three training seeds of the default $64^2/3 \times 3$ model.

# E Additional forecast diagnostics

Figure 3 uses logarithmic vertical axes. Shading for the mean $\pm$ one sample SD is shown only where the lower bound is positive.

## E.1 Energy and frequency-resolved fidelity

Long-horizon neural models can be trained to preserve invariant statistics of chaotic dynamics (Jiang et al., 2023). Here we evaluate energy distributions and field errors at specified forecast horizons. Energy-spectrum discrepancy measures how predicted energy is distributed across wave numbers. Fourier error power is computed from the predicted-minus-reference velocity field, so it also captures phase differences and locates forecast error by spatial scale.

Let $E_{\text{pred}}(k)$ and $E_{\text{ref}}(k)$ be radial kinetic-energy spectra. For each training seed, bars denote means over the 100 periodic starts. The percentage discrepancy is $100 \sum_{k=1}^{128} |\bar{E}_{\text{pred}}(k) - \bar{E}_{\text{ref}}(k)| / \sum_{k=1}^{128} \bar{E}_{\text{ref}}(k)$. Table 14 reports its mean and SD over seeds. The spatial ensemble is averaged before computing this spectrum discrepancy.

Acoustic physical-energy relative error is computed per trajectory before averaging, using the spatial material coefficients and all three dynamic channels. The physical energy is

$$\mathcal{E} = \left\langle \frac{p^2}{2\rho c_s^2} + \frac{\rho}{2}|\boldsymbol{v}|^2 \right\rangle, \tag{21}$$

where $p$ is pressure, $\boldsymbol{v}$ is velocity, $\rho$ and $c_s$ are the static density and sound-speed fields, and $\langle\cdot\rangle$ denotes the spatial mean. Relative error is computed against the reference energy at each forecast step.

For a two-component velocity error $\mathbf{e}$, the error power in radial shell $k$ is

$$E_{\mathrm{err}}(k) = \sum_{\mathrm{round}(\|\boldsymbol{\kappa}\|)=k} \frac{1}{2} \sum_{j=1}^{2} |\mathcal{F}(\mathbf{e}_j)(\boldsymbol{\kappa})|^2. \tag{22}$$

Here $\boldsymbol{\kappa} = (k_x, k_{\mathbf{y}})$ is the integer Fourier wavevector, $k$ is the radial shell index, and $\mathcal{F}$ is the two-dimensional discrete Fourier transform with forward normalization. We retain DC and all corner modes and compute statistics in float64. Summing all shells recovers raw field MSE.

The multistep spectral figure uses four fixed IC2 starts (indices 0, 39, 78, and 118) for all models and seeds at steps 1, 10, and 100. The step-100 data are also used in Figure 5, whose error panel covers all radial shells. Full-cohort spectral statistics use all 100 test starts. Appendix D.3 reports the contribution spectra, and Appendix D.1 gives the true-input diagnostics.

Table 14: **Energy-spectrum discrepancy in periodic flow and physical-energy relative error in acoustics.** Mean ± seed SD. Lower is better. Periodic errors compare mean energy spectra over 100 starts using shells 1–128. Acoustic errors average per-trajectory relative physical-energy errors over 200 trajectories. Definitions and averaging conventions appear in Appendix E.

| Model | Step 1 | Step 10 | Step 50 | Step 100 |
|---|---|---|---|---|
| *Periodic: relative discrepancy of mean energy spectra (%)* | | | | |
| U-Net | $0.062 \pm 0.046$ | $0.780 \pm 0.487$ | $2.413 \pm 1.775$ | $6.561 \pm 1.655$ |
| FNO (modes=16) | $0.132 \pm 0.011$ | $0.814 \pm 0.093$ | $2.672 \pm 0.413$ | $3.942 \pm 0.693$ |
| FNO (modes=32) | $0.082 \pm 0.003$ | $0.660 \pm 0.022$ | $5.092 \pm 1.364$ | $11.587 \pm 4.823$ |
| PAIR | $\mathbf{0.042 \pm 0.010}$ | $\mathbf{0.541 \pm 0.099}$ | $\mathbf{1.876 \pm 0.496}$ | $\mathbf{2.663 \pm 0.799}$ |
| *Acoustic: mean relative physical-energy error (%)* | | | | |
| U-Net | $0.147 \pm 0.014$ | $1.080 \pm 0.352$ | $4.690 \pm 2.237$ | $11.494 \pm 3.245$ |
| PAIR | $\mathbf{0.067 \pm 0.009}$ | $\mathbf{0.371 \pm 0.005}$ | $\mathbf{1.718 \pm 0.368}$ | $\mathbf{5.231 \pm 1.516}$ |

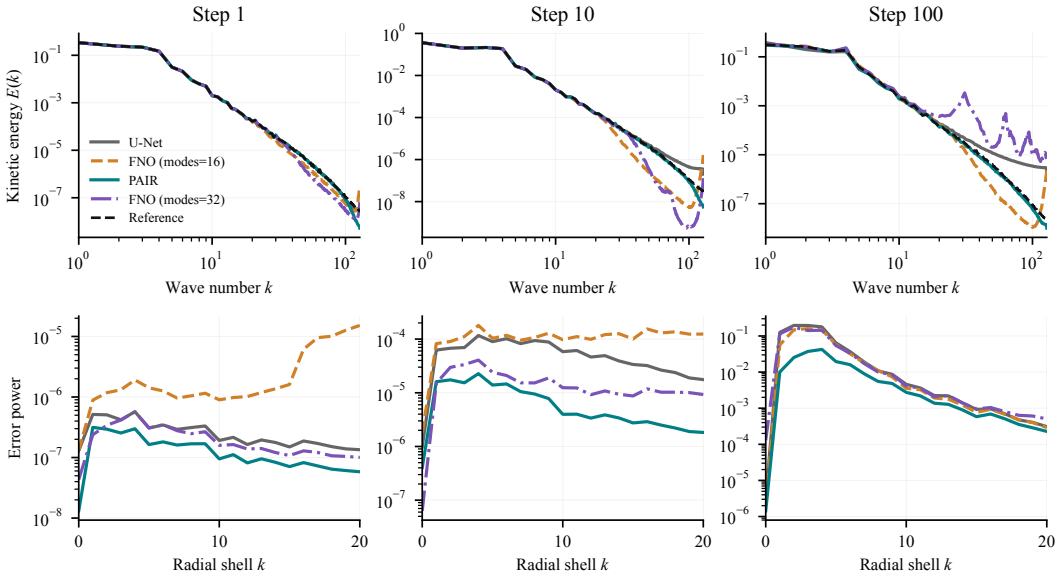



Figure 9: **Energy and prediction-error spectra.** Top: kinetic-energy spectra. The reference is black dashed. Bottom: low-wave-number error power, including amplitude and phase error. Both rows average the same four IC2 starts and three seeds. Error scales vary across time. Table 14 and Appendix E report full-cohort statistics.

Table 15: Relative Fourier error after 100 autoregressive steps over all 100 test starts. For each test start and band, take the square root of the ratio of Fourier error power to reference power, then average starts and training seeds. Values are mean ± seed SD.

| Model | $0 \le k < 32$ | $32 \le k < 64$ | $k \ge 64$ |
|---|---|---|---|
| U-Net | $0.907 \pm 0.016$ | $1.946 \pm 0.049$ | $4.312 \pm 0.269$ |
| FNO (modes=16) | $0.734 \pm 0.006$ | $\mathbf{1.084} \pm 0.014$ | $\mathbf{1.274} \pm 0.029$ |
| FNO (modes=32) | $0.840 \pm 0.095$ | $4.468 \pm 1.736$ | $6.076 \pm 3.625$ |
| **PAIR** | $\mathbf{0.456} \pm 0.025$ | $1.338 \pm 0.006$ | $1.341 \pm 0.018$ |

Table 15 reports relative Fourier errors over all 100 periodic starts. At step 100, the error in $0 \le k < 32$ is 0.456 for PAIR, compared with 0.907 for U-Net and 0.734 for FNO (modes=16). FNO (modes=16) has lower relative error than the default PAIR model in the middle and high bands, while PAIR has lower low-band error and lower whole-field nMSE.

### E.2 One-step accuracy and error accumulation

Increasing FNO's mode count from 16 to 32 lowers periodic step-1 error by 76.3%, but raises step-100 error by 38.7%. The same change lowers endpoint error by 88.4% on reaction–diffusion and 85.4% on acoustics. The effect of this configuration change therefore depends on the task and forecast horizon.

Table 16 compares whole-field true-input error at the 100-step target with free-rollout error at that target. Values are raw MSEs. Table 1 reports periodic nMSE.

Table 16: Periodic true-input and free-rollout errors at the 100-step target. Means are shown above their training-seed SD.

| Model | True-input one-step raw MSE | 100-step rollout raw MSE |
|---|---|---|
| U-Net | $2.252 \times 10^{-5}$ $\pm 8.528 \times 10^{-7}$ | $0.9589$ $\pm 0.02336$ |
| FNO (modes=16) | $0.0004188$ $\pm 5.896 \times 10^{-6}$ | $0.6393$ $\pm 0.004503$ |
| FNO (modes=32) | $9.862 \times 10^{-5}$ $\pm 1.775 \times 10^{-6}$ | $0.8988$ $\pm 0.1914$ |
| **PAIR** | $\mathbf{9.531 \times 10^{-6}}$ $\pm 3.969 \times 10^{-7}$ | $\mathbf{0.28}$ $\pm 0.02583$ |

FNO (modes=32) remains more accurate than modes=16 on true inputs at every evaluated horizon, yet its mean free-rollout endpoint error is larger.

### E.3 Distribution of acoustic endpoint errors

To distinguish typical acoustic errors from the contribution of high-error trajectories, we summarize the 200 step-100 test errors for each model and seed. Table 17 gives their mean, median, and the fraction of summed error contributed by the five largest errors. FNO with 32 modes has similar medians across seeds while its means vary sharply. PAIR has lower medians as well as lower means in these comparisons.

Diagnostics on all 200 validation trajectories compare one-step predictions from true inputs with 100-step free rollouts starting at frame 0, using the same checkpoints, normalization, and static material fields. For FNO (modes=32), the final true-input standardized MSE is $7.95$–$8.25 \times 10^{-5}$ across seeds, whereas free-rollout means range from 0.230 to 5.725.

Table 17: Acoustic step-100 error distributions. The worst five trajectories constitute 2.5% of the test set. Mean and median errors are compared within each seed.

| Model | Seed | Mean | Median | Worst five share (%) |
|---|---|---|---|---|
| U-Net | 42 | 0.02087 | 0.007511 | 48.4 |
| U-Net | 43 | 0.05352 | 0.009822 | 62.79 |
| U-Net | 44 | 0.05941 | 0.008431 | 79.26 |
| FNO (modes=16) | 42 | 5.159 | 0.2643 | 77.7 |
| FNO (modes=16) | 43 | 12.35 | 0.242 | 96 |
| FNO (modes=16) | 44 | 41.92 | 0.3521 | 95.54 |
| FNO (modes=32) | 42 | 0.08962 | 0.01181 | 67.22 |
| FNO (modes=32) | 43 | 4.41 | 0.01149 | 95.91 |
| FNO (modes=32) | 44 | 4.176 | 0.01218 | 99.07 |
| **PAIR** | 42 | **0.01381** | **0.003433** | 50.02 |
| **PAIR** | 43 | **0.01336** | **0.005232** | 40.99 |
| **PAIR** | 44 | **0.00945** | **0.003316** | 46.2 |

# F Solver and model perturbation experiments

## F.1 Construction, propagators, and metrics

Starting at a true IC2 state, we construct zero-mean divergence-free velocity perturbations supported either in $0 < k \leq 8$ or $32 < k \leq 64$. Within each state, both perturbation bands are normalized to the same velocity RMS, set to the chosen fraction of that state's demeaned velocity RMS. Each state has three random directions per band, both signs, and a clean control: 13 trajectories in total. Perturbations are injected only at the initial state.

The reference study in Figure 1 and the model comparison in Figure 6 use the same 60 IC2 states at 1% amplitude. Additional amplitude checks use ten preselected states from this set at 0.3%, 3%, and 10%. Both studies sample the same held-out flow trajectory and forcing/viscosity regime.

All numerical references evolve $512^2$ states under the same equation and forcing and use the same $256^2$ observation map. Py2D applies CN/AB2 with $3/2$ dealiasing and $dt = 10^{-4}$, retaining integrator history across observations. JAX-CFD (Dresdner et al., 2023) applies CN/RK4 with native $2/3$ filtering. Exponax (Koehler et al., 2024)[2] applies ETDRK4 with either its native $2/3$ filter or the matched Py2D $3/2$ spatial operator. These alternatives use $dt = 0.001$.

The numerical checks use JAX float64 on GPU. References for 36 of the 60 Py2D states were integrated on GPU from the start. Those for the other 24 states were continued from verified CPU prefixes using the same algorithm. CPU/GPU comparisons hold algorithm and time step fixed. A four-state IC1 check halves the alternative time step to assess temporal sensitivity.

Let $G^{(h)}$ denote $h$ successive applications of a one-step propagator $G$. For an initial state $\mathbf{u}$ and perturbation $\boldsymbol{\delta}$, the response after $h$ steps is

$$\mathbf{d}_G(h; \mathbf{u}, \boldsymbol{\delta}) = G^{(h)}(\mathbf{u} + \boldsymbol{\delta}) - G^{(h)}(\mathbf{u}). \tag{23}$$

Write $\mathrm{MSE}(\mathbf{a}, \mathbf{b}) = \mathrm{mean}_{j,m} |a_{j,m} - b_{j,m}|^2$, averaging velocity channels and grid points. Squared response magnitude is $\mathrm{MSE}(\mathbf{d}_G, 0)$. Its normalized squared gain divides this value by the initial perturbation MSE within each state. The low/high RMS ratio in Table 18 is the square root of the ratio of the mean squared gains. Directions and signs are averaged within each state, followed by states and, for neural predictors, training seeds.

For each state and frequency band, the state-wise RMS gain is the square root of the normalized squared response gain averaged over the six signed perturbation directions. Figure 1(a) pairs these gains at step 100, with the high-frequency gain on the horizontal axis and the low-frequency gain on the vertical axis. For the curves in Figure 1(b), we average normalized squared gains over directions, signs, and states before taking the square root. The bars at steps 1, 10, 25, 50, 75, and 100 span the 16th–84th percentiles of state-wise RMS gains, showing variation across the 60 sampled IC2 states.

[2]Exponax code: `https://github.com/Ceyron/exponax`.

Let $F$ denote the learned predictor and $S$ the Py2D reference. Response-field mismatch is $\mathrm{MSE}(\mathbf{d}_F, \mathbf{d}_S)$, comparing the change induced by the perturbation after each propagator subtracts its own clean trajectory. Perturbed-forecast error is $\mathrm{MSE}(F^{(h)}(\mathbf{u}+\boldsymbol{\delta}), S^{(h)}(\mathbf{u}+\boldsymbol{\delta}))$, comparing the full fields from the same perturbed initial state. Tables 20 and 21 report these two errors separately.

Table 18: Low/high RMS response ratio at additional perturbation amplitudes on ten IC2 states. Values greater than one indicate a larger low-frequency response. Neural squared gains are averaged over three training seeds before taking the ratio and square root.

| Amplitude | Propagator | Step 1 | Step 10 | Step 50 | Step 100 |
|---|---|---|---|---|---|
| 0.3% | Py2D | 1.028 | 1.517 | 3.305 | 3.057 |
| 0.3% | JAX-CFD | 1.028 | 1.519 | 3.304 | 3.057 |
| 0.3% | Exponax | 1.028 | 1.519 | 3.306 | 3.063 |
| 0.3% | Matched Exponax | 1.028 | 1.517 | 3.305 | 3.057 |
| 0.3% | U-Net | 1.132 | 1.132 | 0.870 | 0.897 |
| 0.3% | **PAIR** | 1.204 | 2.180 | 2.978 | 2.410 |
| 3% | Py2D | 1.028 | 1.510 | 3.182 | 2.307 |
| 3% | JAX-CFD | 1.028 | 1.513 | 3.181 | 2.305 |
| 3% | Exponax | 1.028 | 1.513 | 3.181 | 2.307 |
| 3% | Matched Exponax | 1.028 | 1.510 | 3.182 | 2.307 |
| 3% | U-Net | 1.217 | 0.881 | 0.618 | 0.866 |
| 3% | **PAIR** | 1.338 | 2.200 | 1.949 | 1.676 |
| 10% | Py2D | 1.028 | 1.465 | 2.765 | 1.737 |
| 10% | JAX-CFD | 1.028 | 1.469 | 2.768 | 1.738 |
| 10% | Exponax | 1.028 | 1.469 | 2.768 | 1.737 |
| 10% | Matched Exponax | 1.028 | 1.465 | 2.765 | 1.737 |
| 10% | U-Net | 1.655 | 1.137 | 1.037 | 1.092 |
| 10% | **PAIR** | 2.554 | 4.468 | 4.100 | 2.437 |

### F.2 Results across states and amplitudes

In the 60-state Py2D study, low-frequency response exceeds high-frequency response in every state at steps 1, 10, 50, and 100 after averaging its directions and signs. The pooled step-100 low/high RMS ratio is approximately 2.89 (Figure 1(b)).

At the three additional amplitudes, all four numerical implementations give the same average ordering throughout steps 1–100 (Table 18). U-Net's ordering at step 100 is reversed for 0.3% and 3%, but not for 10%. PAIR has a larger mean response to low-frequency initial perturbations across these amplitudes.

### F.3 Model responses and perturbed forecasts

U-Net, PAIR, and both FNO settings are evaluated on the same 60 states at 1% amplitude, using seeds 42/43/44 and 100-step FP32 rollouts with TF32 disabled. Each checkpoint receives the same saved perturbed inputs and Py2D reference trajectories. For each band, we average six signed directions per state, then states within each seed, and report the mean and sample SD across seeds. Table 19 gives the endpoint errors corresponding to Figure 6, together with the full perturbed-forecast errors.

The ten-state checks at 0.3%, 3%, and 10% retain this ordering of response-field errors. PAIR has lower error than both FNO settings for low-frequency initial perturbations, while FNO (modes=16) has lower error than PAIR for high-frequency initial perturbations (Table 20).

At 10% amplitude, PAIR's first-step normalized RMS response to high-frequency initial perturbations is about 0.39, compared with 0.97 for Py2D.

PAIR also has lower mean step-100 perturbed-forecast MSE than both FNO settings in both bands, on the 60-state comparison and the additional-amplitude checks (Tables 19 and 21). With 10% high-frequency initial perturbations, its first-step error is 12.2% higher than U-Net's, but its errors are lower at steps 10, 50, and 100.

Table 19: **Perturbation accuracy on 60 IC2 states at 1% amplitude.** Raw velocity-field MSE at step 100. Response-field MSE compares perturbation-induced changes. Forecast MSE compares full perturbed trajectories. Low/high identify the initial perturbation band. Means are shown above their sample SD across three training seeds. Bold marks the lowest mean in each column.

| | Response-field MSE | | Forecast MSE | |
|---|---|---|---|---|
| Model | Low | High | Low | High |
| U-Net | 0.152681 ±0.010675 | 0.129573 ±0.033900 | 0.960550 ±0.023238 | 0.978173 ±0.017367 |
| FNO (modes=16) | 0.127205 ±0.010625 | **0.008833** ±0.000056 | 0.662508 ±0.026038 | 0.659593 ±0.025934 |
| FNO (modes=32) | 0.204437 ±0.071638 | 0.505837 ±0.209710 | 0.879952 ±0.172540 | 1.019772 ±0.207570 |
| **PAIR** | **0.081529** ±0.003217 | 0.037199 ±0.027598 | **0.279198** ±0.016793 | **0.293143** ±0.027629 |

Table 20: Step-100 responses at additional perturbation amplitudes on ten IC2 states. Responses are differences from each propagator's own clean trajectory. Magnitude is the mean squared response, while mismatch is the MSE between the model and Py2D responses. Low and high identify the initial perturbation bands. Means are shown above their training-seed SD. Only response-field mismatch is ranked.

| Amplitude | Model | Response, low | Response, high | Mismatch, low | Mismatch, high |
|---|---|---|---|---|---|
| 0.3% | U-Net | 0.04941 ±0.02567 | 0.0611 ±0.00685 | 0.05969 ±0.02562 | 0.06217 ±0.006691 |
| 0.3% | FNO (modes=16) | 0.01494 ±0.008182 | $1.836 \times 10^{-5}$ $\pm 1.409 \times 10^{-5}$ | 0.02502 ±0.008015 | **0.001213** $\pm 1.229 \times 10^{-5}$ |
| 0.3% | FNO (modes=32) | 0.06048 ±0.04816 | 0.453 ±0.3067 | 0.0705 ±0.04727 | 0.4541 ±0.307 |
| 0.3% | **PAIR** | 0.008648 ±0.00102 | 0.00141 ±0.0007176 | **0.01339** ±0.0009134 | 0.002092 ±0.0006916 |
| 3% | U-Net | 0.4182 ±0.06103 | 0.5643 ±0.08779 | 0.7352 ±0.05154 | 0.6367 ±0.08643 |
| 3% | FNO (modes=16) | 0.3768 ±0.03931 | 0.002074 ±0.001404 | 0.6367 ±0.03369 | **0.07992** ±0.001501 |
| 3% | FNO (modes=32) | 0.5359 ±0.2458 | 0.8617 ±0.4196 | 0.818 ±0.2461 | 0.9389 ±0.4224 |
| 3% | **PAIR** | 0.3256 ±0.01315 | 0.1185 ±0.06517 | **0.3078** ±0.01658 | 0.1669 ±0.07457 |
| 10% | U-Net | 1.095 ±0.08006 | 0.9188 ±0.04702 | 1.647 ±0.07381 | 1.254 ±0.05348 |
| 10% | FNO (modes=16) | 1.072 ±0.03305 | 0.0354 ±0.01037 | 1.388 ±0.06147 | **0.4394** ±0.008984 |
| 10% | FNO (modes=32) | 1.198 ±0.3292 | 1.162 ±0.563 | 1.692 ±0.3268 | 1.486 ±0.5397 |
| 10% | **PAIR** | 1.097 ±0.01389 | 0.1899 ±0.09254 | **0.5723** ±0.04675 | 0.5073 ±0.09811 |

## G ADDITIONAL FIELD VISUALIZATIONS

Models share each input, reference, and zoom window. Periodic examples use seed 42. RD and acoustics use seed 43. Figure 4 shows periodic start 0, RD trajectory 922, and acoustic trajectory 10. RD and acoustic cases and windows were selected where PAIR has lower error and visible spatial differences.

Periodic and acoustic panels show speed. RD panels show activator concentration. The full reference field's range at each displayed time sets the colors for all models and enlargements, with zero at white for RD. Colorbar triangles indicate values outside this range. RD contours at $u = 0.2$ use black dashed reference lines and teal predictions. The main RD zoom covers $18 \times 18$ grid points, with linear contour interpolation and bilinear raster display. Periodic windows emphasize reference gradients, and acoustic windows show wavefront structure.

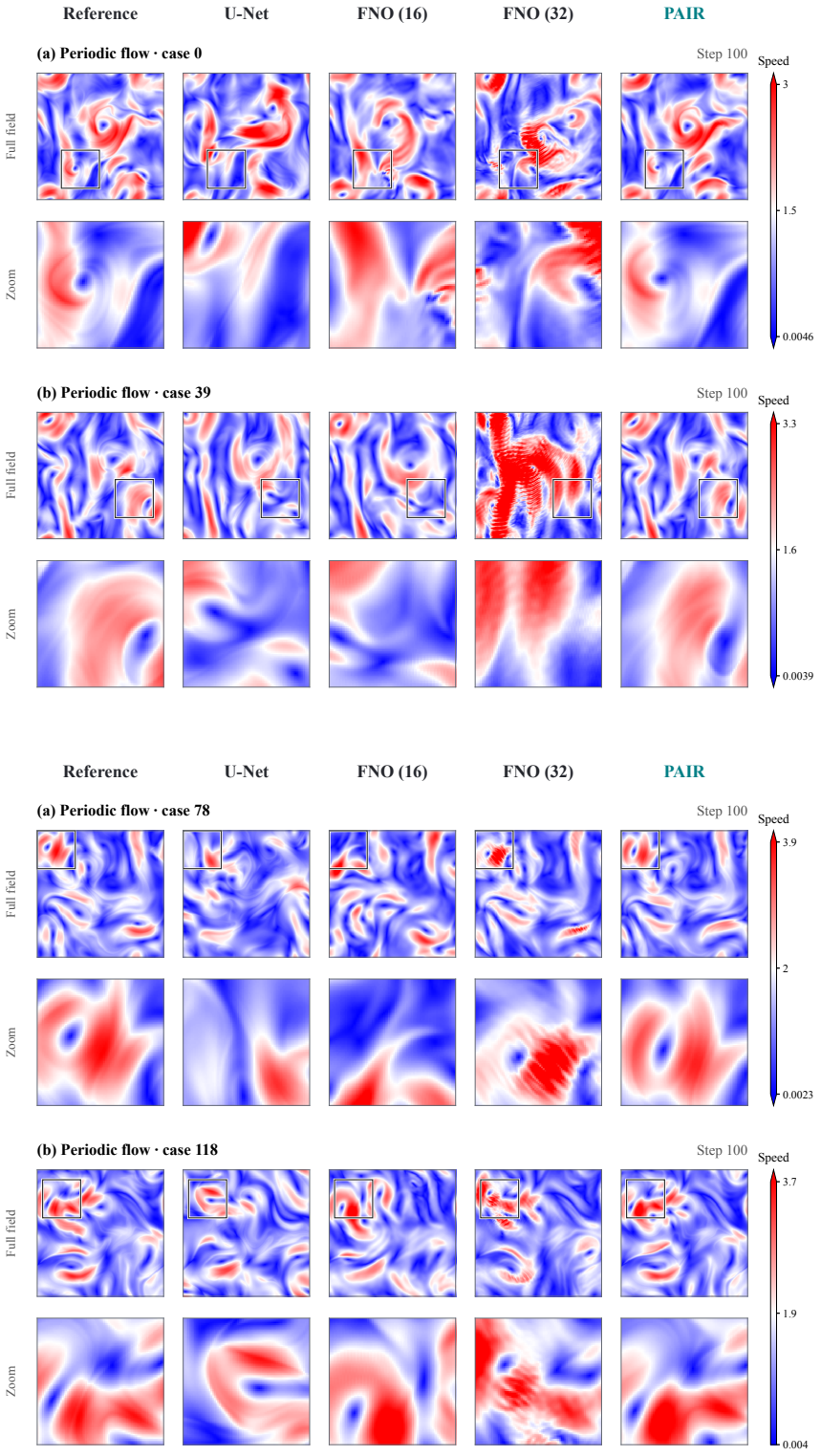



Figure 10: **Periodic flow: additional test examples.** Speed at step 100 for cases 0 and 39 (upper plate), and 78 and 118 (lower plate). Each case shows full fields and matched enlargements using the reference color range.

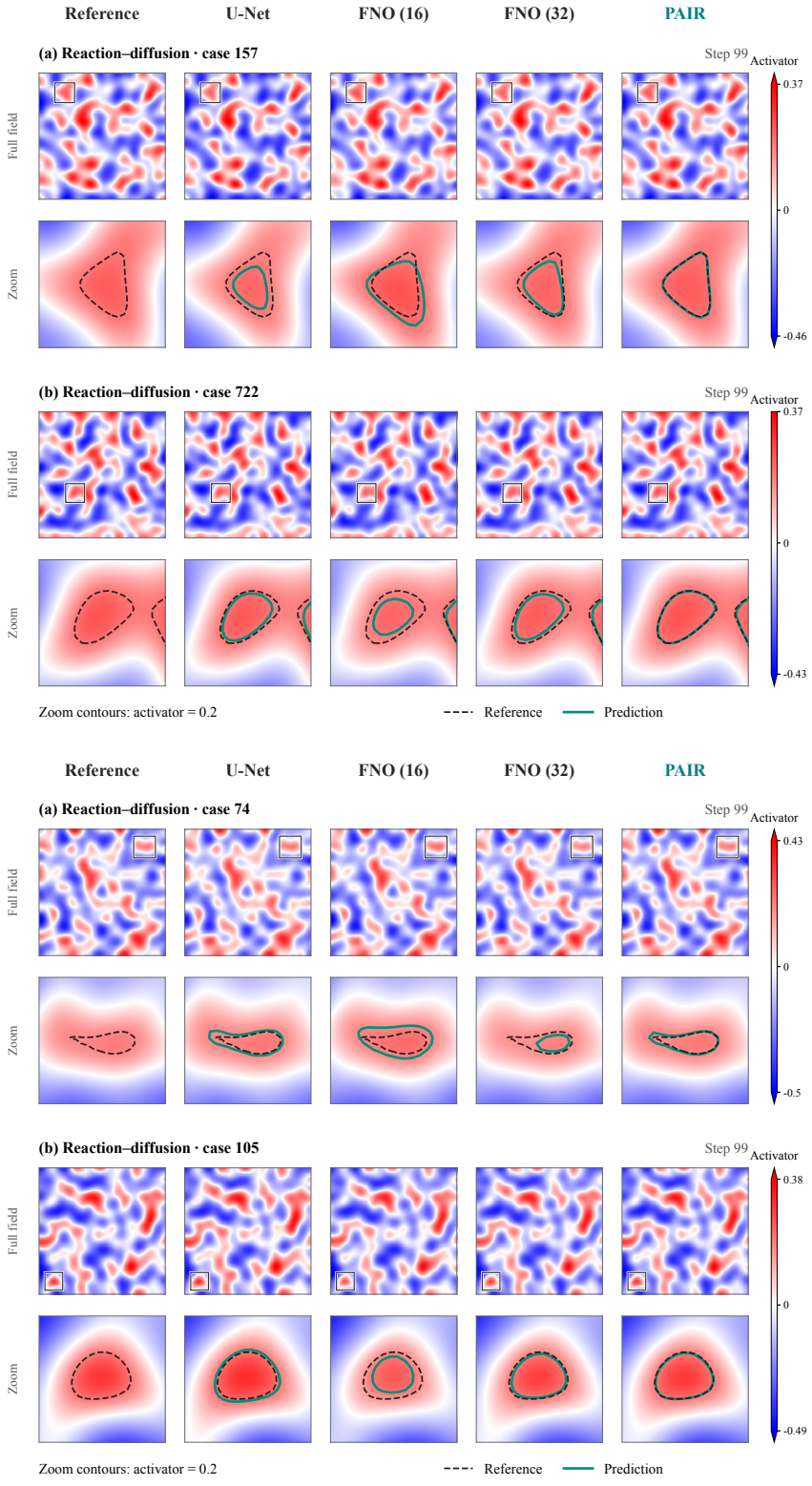



Figure 11: **Reaction–diffusion: additional test examples.** Activator concentration at step 99 for cases 157 and 722 (upper plate), and 74 and 105 (lower plate). Reference and predicted contours at $u = 0.2$ are black dashed and teal solid, respectively. Each case shows full fields and matched enlargements using the reference color range.

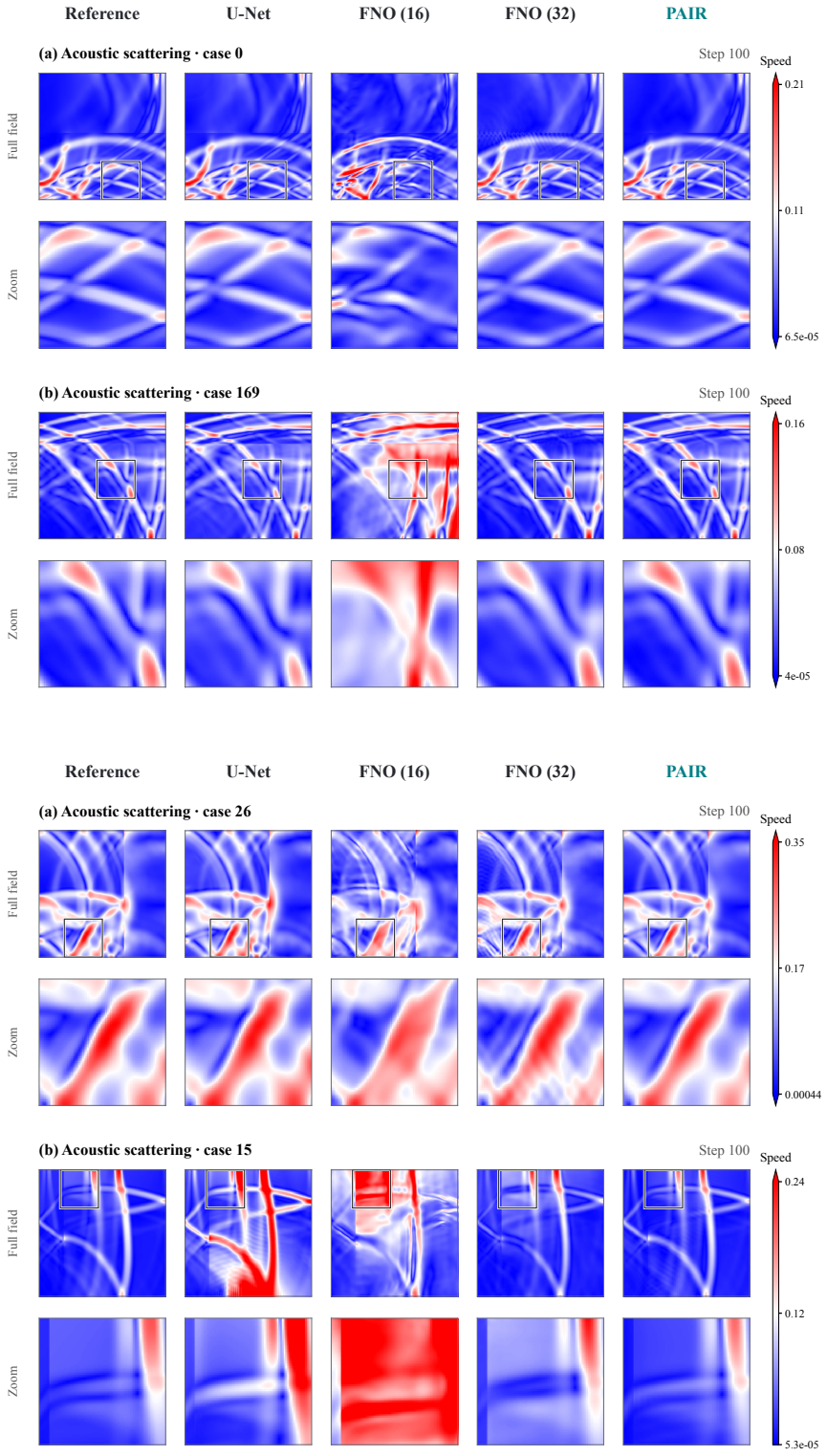



Figure 12: **Acoustic scattering: additional test examples.** Speed at step 100 for cases 0 and 169 (upper plate), and 26 and 15 (lower plate). Each case shows full fields and matched enlargements using the reference color range.

Table 21: Prediction error at step 100 for additional perturbation amplitudes. Raw velocity-field MSE against the corresponding perturbed Py2D trajectory, averaged over six signed directions and ten IC2 states within each training seed. Values are mean ± SD over three seeds. Columns identify the initial perturbation band. Errors cover the full velocity field.

| Amplitude | Model | Low-frequency perturbation | High-frequency perturbation |
|---|---|---|---|
| 0.3% | U-Net | $0.9929 \pm 0.02851$ | $0.9944 \pm 0.02571$ |
| 0.3% | FNO (modes=16) | $0.7143 \pm 0.05732$ | $0.7123 \pm 0.05942$ |
| 0.3% | FNO (modes=32) | $1.053 \pm 0.2769$ | $1.061 \pm 0.24$ |
| 0.3% | **PAIR** | $\mathbf{0.2825} \pm 0.03845$ | $\mathbf{0.2815} \pm 0.03789$ |
| 3% | U-Net | $1.002 \pm 0.03149$ | $1.113 \pm 0.03563$ |
| 3% | FNO (modes=16) | $0.7545 \pm 0.02859$ | $0.7233 \pm 0.05199$ |
| 3% | FNO (modes=32) | $1.059 \pm 0.2467$ | $1.252 \pm 0.2827$ |
| 3% | **PAIR** | $\mathbf{0.3029} \pm 0.02343$ | $\mathbf{0.316} \pm 0.06359$ |
| 10% | U-Net | $1.065 \pm 0.03644$ | $1.26 \pm 0.06216$ |
| 10% | FNO (modes=16) | $0.7917 \pm 0.01828$ | $0.8046 \pm 0.02731$ |
| 10% | FNO (modes=32) | $1.043 \pm 0.1698$ | $1.421 \pm 0.2837$ |
| 10% | **PAIR** | $\mathbf{0.3392} \pm 0.03882$ | $\mathbf{0.4961} \pm 0.06911$ |

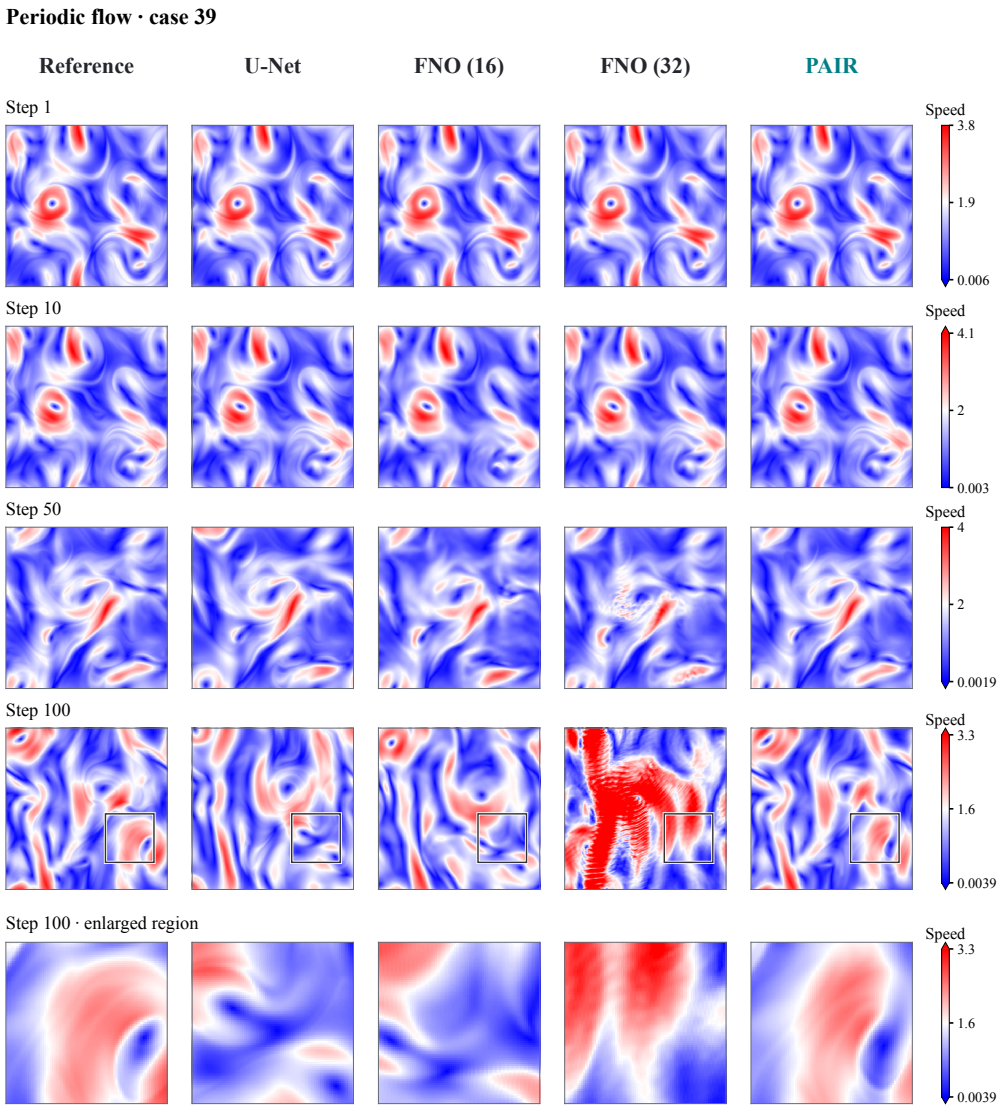



Figure 13: **Periodic flow: evolution over time.** Start 39 at steps 1, 10, 50, and 100, followed by an enlargement of the boxed step-100 region. Each row compares the same reference and all four predictions using the reference speed range. The final full-field and zoom rows share a color scale.

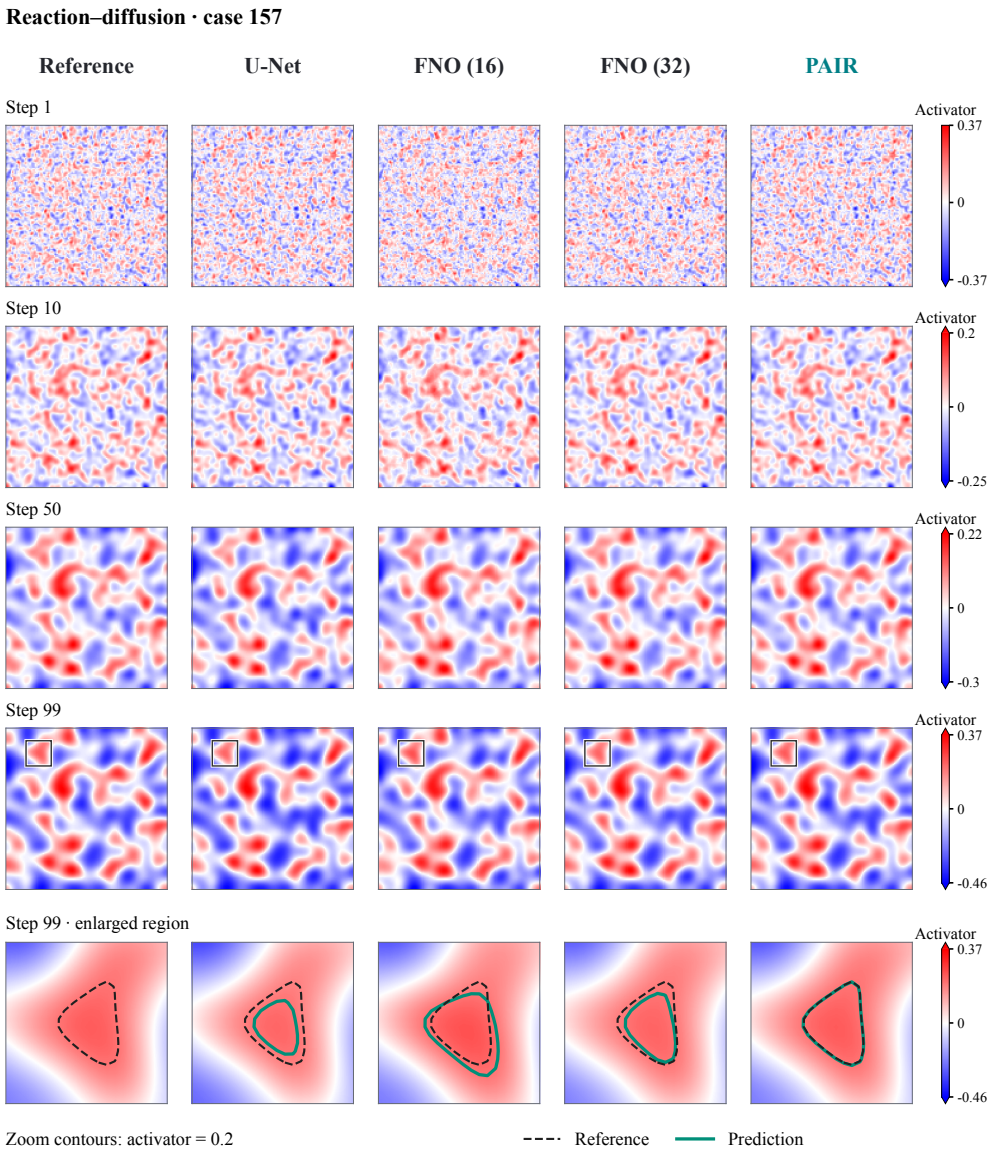



Figure 14: **Reaction–diffusion: evolution over time.** Activator concentration on trajectory 157 at steps 1, 10, 50, and 99. The last row enlarges the boxed step-99 region, with matched $u = 0.2$ contours. Each row uses the reference range. The final full field and enlargement share it.

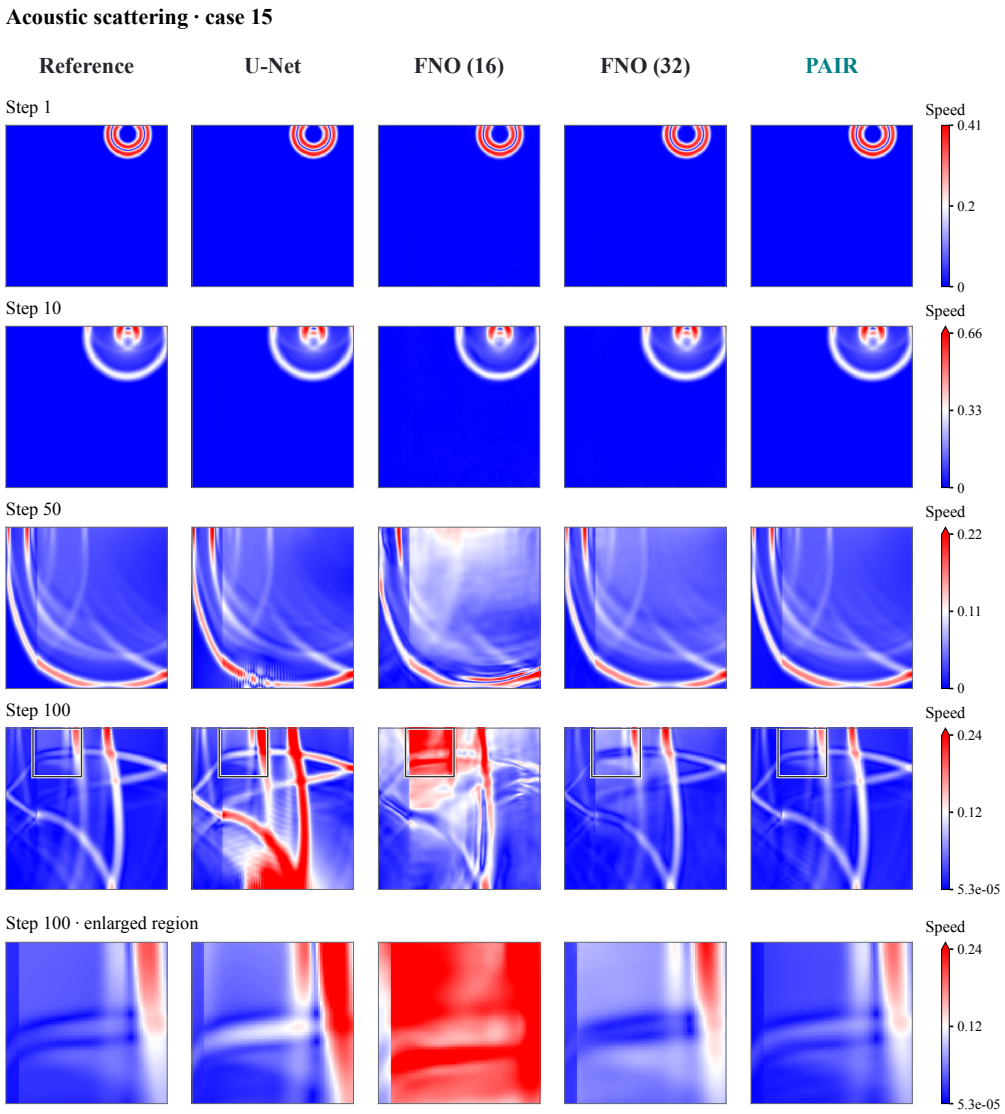



Figure 15: **Acoustic scattering: evolution over time.** Trajectory 15 at steps 1, 10, 50, and 100, followed by an enlargement of the boxed step-100 wavefront region. Each row uses the reference speed range. The final full field and enlargement share it.