Coupled Horizons as Operator Coordinates for PDE Trajectory Prediction

Anonymous authors
Paper under double-blind review

Abstract

Neural PDE surrogates aim to predict finite-time trajectories from observed states. Existing approaches either advance trajectories sequentially or tie joint prediction to a prescribed temporal discretization, making cross-horizon interaction and flexible temporal queries difficult to realize simultaneously. We introduce CURVO, a horizon-coupled neural operator that treats prediction horizons as operator coordinates and represents finite-time evolution through interval-average responses. A multiscale spatial–horizon architecture with shared interval-conditioned refinement couples the queried horizons and reconstructs all requested states from a single observed state in one pass. Trained with a modest fixed query cardinality, the same operator directly evaluates query sets of different cardinalities and placements and supports dense readout of trajectories with up to one hundred states. Across five PDE configurations, CURVO reduces trajectory-averaged error by 4.6–37.4% over the strongest one-pass comparator. The formulation further extends to teacher-free one-step image generation with jointly predicted intermediate transport states. CURVO thus makes temporal resolution an inference-time choice while retaining explicit cross-horizon interaction.

1 Introduction

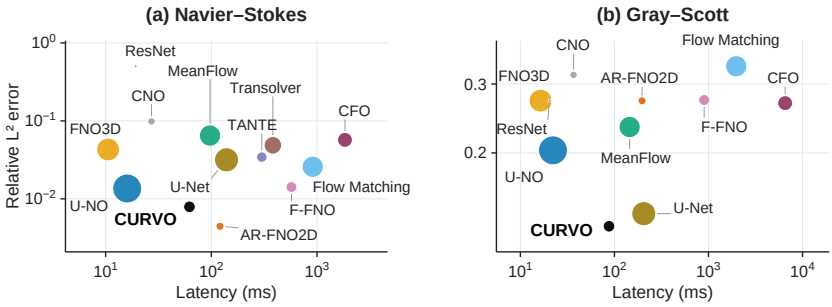

Figure 1: Accuracy–efficiency trade-offs on (a) Navier–Stokes ($\nu = 10^{-3}$) and (b) Gray–Scott. Each point denotes one comparison method. The axes report trajectory-averaged relative L^2 error and inference latency for $K = 8$ queried horizons; bubble area is proportional to parameter count. Lower-left is better. CURVO lies on the accuracy–latency Pareto frontier in both settings.

Many physical systems governed by time-dependent PDEs combine gradual evolution with localized events: spots grow and split in reaction–diffusion systems (Pearson, 1993), while defects form and annihilate in active nematics (Giomi et al., 2013). Characterizing these dynamics requires resolving transitions at their characteristic temporal scales. Adaptive numerical solvers decide temporal resolution as the solution unfolds, selecting time steps from local error estimates (Hairer et al., 1993; Hindmarsh et al., 2005), but each trajectory requires a full numerical evolution, and repeated solves dominate the cost of design and inference pipelines (Li et al., 2021; Wu et al., 2024).

Neural operators accelerate repeated solves by learning reusable field mappings (Li et al., 2021; Lu et al., 2021), but differ in how they represent and query temporal evolution. Autoregressive

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.

predictors construct trajectories through successive learned transitions (Brandstetter et al., 2022), while continuous-time vector-field models recover trajectories through numerical integration (Chen et al., 2018; Hou et al., 2026). Space–time operators such as FNO3D jointly predict trajectory segments on regular space–time grids (Li et al., 2021). Coordinate-based architectures such as DeepONet and CViT permit flexible output queries, but decode them query-wise from a shared input representation without explicit feature exchange across the query set (Lu et al., 2021; Wang et al., 2025). How can a neural surrogate bypass sequential rollout, couple predictions across horizons, and evaluate temporal query sets reconfigured after training in a single pass?

We introduce CURVO, a *horizon-coupled neural operator* for one-pass PDE trajectory prediction with temporal resolution chosen at inference, in which predictions at different horizons interact within a shared spatial–horizon representation rather than being computed sequentially or independently (Figure 2). Given an observed state u_t, a *query context* $\Xi = (h_1 < \cdots < h_K)$ specifies the future time offsets to be evaluated. CURVO treats these horizons as coordinates and jointly approximates the *interval-average responses* $\bar{v}_t(h) = [u(t+h) - u_t]/h$, the average state change per unit time over $[t, t+h]$, from which the queried states are recovered algebraically. To model scale-dependent interval responses, CURVO uses a U-shaped multiscale neural-operator backbone that couples spatial and horizon features across scales and refines the resulting latent field through repeated application of a shared interval-conditioned operator. This yields a coupled response representation over the query context, from which all requested states are reconstructed jointly from the same anchor in one forward pass.

Across five configurations, CURVO reduces trajectory-averaged error by 4.6–37.4% over the strongest one-pass comparator and by 10.1–60.0% over the best sequential method on three of four PDE families, while remaining among the fastest and most compact models evaluated (Figure 1). Trained with a modest fixed query cardinality, CURVO generalizes without retraining to unseen cardinalities, irregular layouts, and off-grid horizons; fixed-target predictions remain stable under context changes, and the same checkpoint reads a complete 100-state Gray–Scott trajectory in one pass. The formulation extends to teacher-free one-step CIFAR-10 generation, jointly producing terminal and intermediate transport states without distillation or cached teacher trajectories (Appendix 7.1). CURVO makes temporal resolution an inference-time choice while retaining explicit cross-horizon interaction.

Our main contributions are:

- **Prediction horizons as operator coordinates.** We cast finite-horizon PDE prediction as learning the interval-average responses of an observed state over the full range of horizons, so that the entire finite-time evolution becomes a single operator output, evaluated at inference-chosen horizons rather than propagated on a prescribed temporal grid.
- **Horizon-coupled neural operator.** We develop a multiscale spatial–horizon architecture with explicit cross-horizon interaction and shared interval-conditioned latent refinement, enabling all queried states to be reconstructed jointly from a single observed state in one pass.
- **Accurate and efficient PDE prediction.** Across five configurations, CURVO attains the lowest trajectory-averaged error among the evaluated one-pass predictors at single-evaluation latency; ablations isolate the gains from horizon coupling and interval-conditioned refinement. The same formulation further extends to teacher-free one-step CIFAR-10 generation.

2 Related Work

Temporal representations in neural operators. Neural operators differ in how they represent and query temporal evolution. Space–time FNO predicts trajectories on regular space–time grids (Li et al., 2021), while DeepONet and CViT support flexible output coordinates through query-wise decoding (Lu et al., 2021; Wang et al., 2025). VCNeF evaluates multiple temporal coordinates in parallel without explicitly modeling dependencies across them (Hagnberger et al., 2024), whereas P-STMAE uses temporal attention to jointly reconstruct missing and future states from multi-frame sequences (Zhu et al., 2026). Continuous-time methods instead parameterize or advance temporal dynamics through solution maps, learned vector fields, or local expansions (Biloš et al., 2021; Hou et al., 2026; Wu et al., 2026); lead-time-conditioned operators such as Poseidon and RIGNO directly predict states at specified horizons (Herde et al., 2024; Mousavi et al., 2025). CURVO combines

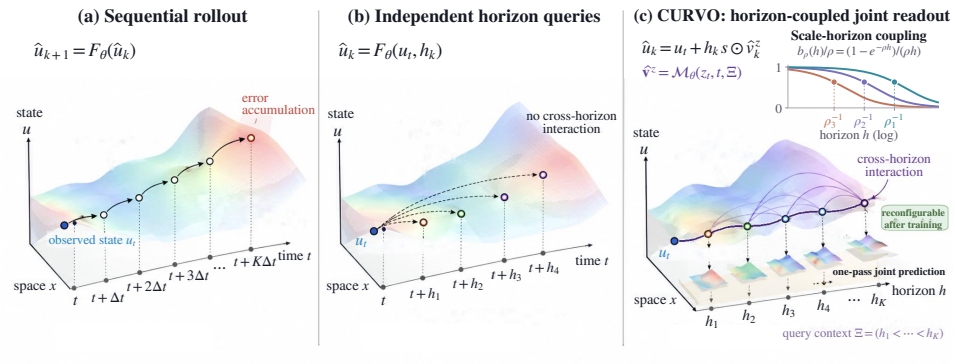

Figure 2: (a) Sequential rollout propagates predictions recursively. (b) Independent queries predict horizons separately from the anchor. (c) Driven by rate-dependent transition scales $h \sim \rho^{-1}$ in $b_\rho(h) = (1 - e^{-\rho h})/h$, CURVO jointly models the query context through spatial–horizon coupling and reconstructs all requested states in one forward pass, with query cardinality and placement chosen at inference. CURVO-specific notation is defined in Section 3.

flexible temporal queries with explicit cross-horizon interaction from a single observed state, with query cardinality and placement chosen after training.

Efficient trajectory inference. Autoregressive surrogates construct trajectories through repeated state transitions, with temporal bundling reducing network calls and pushforward training improving rollout stability (Brandstetter et al., 2022). Other approaches reduce spatial or iterative inference cost through adaptive meshes, trajectory-level generative models, rectified flows, or distillation (Wu et al., 2023; Hu et al., 2025; Rühling Cachay et al., 2023; Armegioiu et al., 2025; Zhang & Zou, 2026). CURVO instead directly regresses finite-interval responses from recorded trajectories and reconstructs all requested states from a single observed state in one forward pass.

One-step generative transport. One-step generators obtain fast sampling through distillation from pretrained multi-step models (Song et al., 2023; Salimans & Ho, 2022) or through teacher-free endpoint and interval objectives (Song & Dhariwal, 2024; Geng et al., 2025), in both cases returning the terminal sample alone; DSNO outputs the teacher's trajectory, but on a fixed grid and from cached rollouts (Zheng et al., 2023). CURVO needs neither a teacher nor a fixed readout grid: intermediate states and the terminal sample are read jointly in one evaluation, at query context chosen after training.

3 Method

CURVO learns a finite segment of PDE dynamics through jointly estimated interval-average responses. The horizon enters as a coordinate of the learned operator, and the set of horizons is chosen at evaluation time. A multiscale spatial–horizon representation couples the requested responses, and a shared interval-conditioned update refines their latent features before algebraic state readout.

3.1 Finite-interval operator formulation

We consider a state field $u(x,t)$ on a spatial domain $\Omega \subset \mathbb{R}^d$, evolving according to

$$\partial_t u(x,t) = \mathcal{N}_\lambda[u,t](x), \qquad u(x,0) = u_0(x), \qquad t \in [0,T], \tag{1}$$

with appropriate boundary conditions. The parameters λ are fixed along each trajectory and suppressed below. We write $u(t) = u(\cdot,t)$ for the state field.

Given an observed anchor $u_t = u(t)$, we seek a representation of its forward solution segment $h \mapsto u(t+h)$ on $[0, H_t]$, where $H_t > 0$ and $t+H_t \le T$. An ordered query context $\Xi = (h_1 < \cdots < h_K)$, with $0 < h_k \le H_t$, specifies the states $\{u(t+h_k)\}_{k=1}^K$ to be estimated jointly from the anchor. Both the cardinality K and the placement of Ξ are inference-time choices.

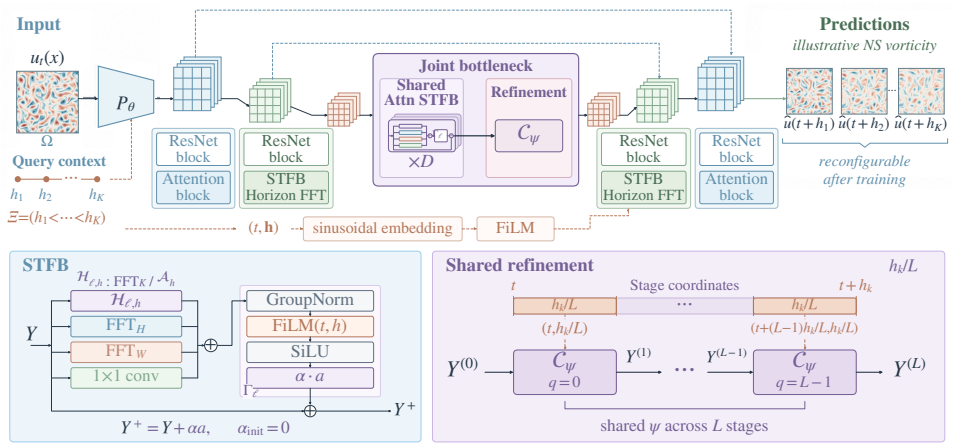

Figure 3: CURVO architecture. *Top:* Given u_t and query context $\Xi = (h_1 < \cdots < h_K)$, the lift $Y_0(x, h_k) = P_\theta(z_t(x), \eta(t, h_k))$ introduces horizons as operator coordinates. The query axis is retained across the multiscale hierarchy, where $\mathcal{H}_{\ell,h}$ captures scale-dependent horizon interactions and bottleneck attention $\mathcal{A}_h$ incorporates the physical offsets $h_k - h_j$. The decoder returns $\widehat{v}^{\,z}$ and reconstructs all queried states in one pass. *Bottom left:* Each coupled block $\mathcal{T}_\ell$ combines spatial mixing $\mathcal{K}_{\ell,x}$ with horizon interaction $\mathcal{H}_{\ell,h}$, coupling temporal dependence to spatial scale. *Bottom right:* The shared refinement cell C_ψ is reused at $(t + q\boldsymbol{h}/L, \boldsymbol{h}/L)$ to refine interval dependence across latent stages before decoding.

CURVO parameterizes this segment by its interval-average response $\bar{v}_t(h) = [u(t + h) - u_t]/h$ for $0 < h \le H_t$. For classical solutions, the evolution law gives

$$\bar{v}_t(h) = \frac{1}{h}\int_0^h \mathcal{N}[u(t + r), t + r]\,\mathrm{d}r, \tag{2}$$

with $\bar{v}_t(h) \to \mathcal{N}[u_t, t]$ as $h \downarrow 0$, and at every positive horizon the state is recovered algebraically as $u(t + h) = u_t + h\,\bar{v}_t(h)$ (Geng et al., 2025).

We organize these responses as a horizon-indexed *response curve* of spatial fields,

$$\mathcal{G}_t(u_t) = [h \mapsto \bar{v}_t(h)], \qquad \mathcal{G}_t(u_t)|_\Xi = \big(\bar{v}_t(h_1), \ldots, \bar{v}_t(h_K)\big), \tag{3}$$

for $0 < h \le H_t$. CURVO estimates this finite restriction jointly: the query coordinates specify both the response locations and the interactions within a shared spatial–horizon representation.

3.2 Horizon-coupled neural operator

To capture the inherent coupling between spatial scales and temporal dynamics, CURVO represents finite-interval responses as a unified multiscale field across space and prediction horizons. For a dissipative mode with spatial Fourier coefficient $\widetilde{u}(\omega, t)$ satisfying $\partial_t\widetilde{u} = -\rho(\omega)\widetilde{u}$ and $\rho(\omega) \ge 0$, Equation 2 gives

$$\widetilde{\bar{v}}(\omega, t, h) = -b_{\rho(\omega)}(h)\,\widetilde{u}(\omega, t), \qquad b_\rho(h) = \frac{1 - e^{-\rho h}}{h}. \tag{4}$$

Here $0 \le b_\rho(h) \le \min\{\rho, h^{-1}\}$. Each positive rate sets a transition horizon $h \sim \rho^{-1}$, from the local response $-\rho\widetilde{u}$ at $\rho h \ll 1$ to the interval-averaged change $-\widetilde{u}/h$ at $\rho h \gg 1$. A separated description of these responses uses shared temporal profiles $g_a(h)$ with rate-dependent coefficients c_{ia}.

Proposition 1 (Scale–horizon separation rank)**.** *Let $I \subset (0, \infty)$ be a nonempty open interval. For distinct positive rates $\rho_1, \ldots, \rho_n$, any representation $b_{\rho_i}(h) = \sum_{a=1}^{R} c_{ia} g_a(h)$ valid on I, with $g_1, \ldots, g_R$ independent of ρ, requires $R \ge n$. Consequently, no fixed finite ρ-independent temporal basis represents this response family exactly when ρ ranges over a nondegenerate interval of positive rates (Appendix 7.5).*

Motivated by this scale-dependent horizon structure, CURVO uses coordinate-conditioned features in a U-shaped hierarchy. The query axis is retained at each spatial resolution, while *horizon-coupled operator blocks* jointly transform spatial and horizon features. The shared lift $Y_0(x, h_k) = P_\theta(z_t(x), \eta(t, h_k))$ combines the normalized anchor with physical interval embeddings. A coupled block indexed by ℓ applies $Y_\ell^+ = \mathcal{T}_\ell(Y_\ell; t, \Xi)$, where

$$\mathcal{T}_\ell(Y_\ell; t, \Xi) := Y_\ell + \Gamma_\ell\big(\mathcal{K}_{\ell,x} Y_\ell + \mathcal{H}_{\ell,h} Y_\ell + W_\ell Y_\ell;\ t, \Xi\big)\,. \tag{5}$$

Here $\mathcal{K}_{\ell,x}$ uses factorized spatial Fourier branches (Li et al., 2021; Tran et al., 2023), W_ℓ mixes channels pointwise, and $\mathcal{H}_{\ell,h}$ is the horizon interaction, applied within each spatial scale rather than as a scale-independent temporal basis. Their outputs enter a conditioned normalization–activation–gate update Γ_ℓ. At intermediate encoder and decoder resolutions, the horizon branch is spectral:

$$\mathcal{H}_{\ell,h} Y = \mathcal{F}_K^{-1}\big[\mathbf{1}_{\{m \in \mathcal{M}_\ell\}} R_{\ell,h}(m)(\mathcal{F}_K Y)(m)\big]\,. \tag{6}$$

The transform $\mathcal{F}_K$ acts on the ordered query index $h_1 < \cdots < h_K$. Each cardinality uses the representable subset of the same learned mode bank $\mathcal{M}_\ell$, with all other coefficients zero: parameters attach to modes rather than output slots. Jointly processed encoder features reach the decoder through skip connections, where further spatial–horizon updates refine the reconstruction.

Within each bottleneck block, the horizon branch instead uses a *physical-coordinate attention kernel*, retaining the parallel spatial and pointwise branches of Equation 5. For one attention head at a fixed spatial location x, suppressing the block and location indices, write $Y_j = Y(x, h_j)$ and $\xi_j = (h_j, Y_j)$. These tokens define the empirical context $\nu_{\Xi,Y} = K^{-1} \sum_{j=1}^K \delta_{\xi_j}$ on the coordinate–feature domain E. The normalized interaction is (Chen & Wu, 2024; Calvello et al., 2025; Castin et al., 2024)

$$A_\theta(\xi; \nu) = \frac{\int_E e^{S_\theta(\xi,\zeta)} v_\theta(\zeta)\, d\nu(\zeta)}{\int_E e^{S_\theta(\xi,\zeta)}\, d\nu(\zeta)}, \qquad [\mathcal{A}_h Y]_k = A_\theta(\xi_k; \nu_{\Xi,Y}). \tag{7}$$

Here $\mathcal{A}_h$ is the per-head attention interaction of the bottleneck horizon branch $\mathcal{H}_{\ell,h}$, v_θ is the value map, and the score combines feature compatibility, individual horizon embeddings, and a bias in the physical offset $h_k - h_j$. Representing the context through $\nu_{\Xi,Y}$ separates the attention operator from a particular discrete query cardinality.

Remark 2 (Variable-context attention). For a compact metric space (E, d_E) and Lipschitz maps $S_\theta : E \times E \to \mathbb{R}$ and $v_\theta : E \to \mathbb{R}^m$, standard regularity estimates for measure-valued attention give Lipschitz dependence of $A_\theta(\xi; \nu)$ on the target token ξ in d_E and on the empirical context ν in the Wasserstein-1 metric, with constants independent of the finite support cardinality (Vuckovic et al., 2020; Castin et al., 2024). Hence Equation 7 defines a single parameterized attention operator over finite query contexts of varying cardinality.

CURVO further organizes interval dependence through a *shared interval-conditioned latent operator composition*, built from the same bottleneck block structure. Its stage coordinates are derived from the telescoping decomposition along the reference trajectory:

$$\bar{v}_t(h) = \frac{1}{L} \sum_{q=0}^{L-1} \bar{v}_{\,t+\frac{q}{L}h}\left(\frac{h}{L}\right), \qquad L \in \mathbb{N}_{>0}. \tag{8}$$

Let $Y^{(0)}$ be the output of the ordinary bottleneck stack, and write $\boldsymbol{h} = (h_1, \ldots, h_K)$ for the horizon vector of Ξ. The refinement cell C_ψ is a separately parameterized copy of a bottleneck block, reused at stages $q = 0, \ldots, L-1$:

$$C_\psi^{[q]}(Y) := C_\psi\big(Y;\ t + \tfrac{q}{L}\boldsymbol{h},\ \tfrac{\boldsymbol{h}}{L}\big)\,, \qquad Y^{(q+1)} = C_\psi^{[q]}(Y^{(q)}). \tag{9}$$

The cell's attention uses the subinterval coordinates $\boldsymbol{h}/L$ and their pairwise offsets, while its affine modulation is conditioned on $(t + q\boldsymbol{h}/L, \boldsymbol{h}/L)$. Each stage therefore jointly updates spatial and horizon features at the corresponding interval coordinates, without decoding intermediate states. A single decoder combines $Y^{(L)}$ with the multiscale skip features and returns all normalized interval-average responses.

3.3 Training and inference

The joint operator learns the response curve from recorded state pairs, with supervision and state reconstruction expressed in the same interval-average coordinates. Let $z = (u - \mu) \oslash s$ and $z_t = z(t)$, with fixed training-set channelwise means μ and stabilized standard deviations $s > 0$. For each sampled trajectory, we draw an ordered context $\Xi = (h_1 < \cdots < h_{K_{\mathrm{tr}}})$ of physical offsets from the recorded grid, with K_{tr} fixed throughout training, then select a common anchor u_t for which all queried endpoints are recorded within $(t, t + H_t]$. Here $H_t = \min\{H, T - t\}$ for the dataset horizon scale H (Appendix 7.6). The target responses and joint prediction are

$$\bar{v}_k^{z,\star} = \frac{z(t + h_k) - z_t}{h_k}, \qquad \widehat{\boldsymbol{v}}^{\,z} = \mathcal{M}_\theta(z_t, t, \Xi). \tag{10}$$

To reduce the relative influence of large residuals on the shared representation, we use an elementwise adaptively reweighted regression loss (Geng et al., 2025). Writing $r_{kca} = \widehat{v}_{k,c}^{\,z}(x_a) - \bar{v}_{k,c}^{z,\star}(x_a)$, we minimize

$$\mathcal{L}(\theta) = \mathbb{E}_{u,t,\Xi}\left[\frac{1}{K_{\mathrm{tr}} C N_x} \sum_{k,c,a} \frac{r_{kca}^2}{\left(\mathrm{sg}(r_{kca}^2) + \varepsilon\right)^p}\right]. \tag{11}$$

Here C and N_x are the numbers of channels and spatial grid points; sg holds the residual-dependent weights fixed during backpropagation, while $p > 0$ controls reweighting and $\varepsilon > 0$ stabilizes the denominator.

At inference, the same trained operator evaluates an ordered context $0 < h_1 < \cdots < h_K \le H_t$ whose cardinality and placement may differ from those used in training; the horizons are continuous coordinates and need not lie on the recorded grid. One joint evaluation reconstructs all physical states through

$$\widehat{u}(t + h_k; \Xi) = u_t + h_k\, s \odot \widehat{v}_k^{\,z}, \qquad k = 1, \ldots, K. \tag{12}$$

For the corresponding reference response, $\widehat{u}(t + h_k; \Xi) - u(t + h_k) = h_k\, s \odot (\widehat{v}_k^{\,z} - \bar{v}_k^{z,\star})$, which exactly connects response error to physical-state error. Every state is read from the observed anchor rather than propagated from an earlier prediction.

4 Experiments

We evaluate CURVO on four time-dependent PDE benchmarks: two-dimensional Navier–Stokes (NS) at $\nu \in \{10^{-3}, 10^{-4}\}$ (Li et al., 2021), radial dam-break shallow water (SWE) from PDEBench (Takamoto et al., 2022), and Gray–Scott (GS) and Active Matter (AM) from The Well (Ohana et al., 2024; Maddu et al., 2024). Spanning vortex evolution, wave propagation, nonlinear pattern formation, and active-fluid dynamics, these systems probe finite-horizon trajectory approximation under complementary regimes. All methods predict states at the same physical horizons from a single observed snapshot u_t, on 64×64 spatial grids.

We compare CURVO against established neural-operator baselines and recent temporal prediction methods under a common finite-horizon evaluation: (i) direct spatial and space–time models (FNO3D, F-FNO (Li et al., 2021; Tran et al., 2023), U-NO, CNO (Rahman et al., 2023; Raonić et al., 2023), U-Net, ResNet (Ronneberger et al., 2015; He et al., 2016)); (ii) direct single-interval prediction (MeanFlow (Geng et al., 2025)); and (iii) sequential and continuous-time predictors (AR-FNO2D, Flow Matching (Lipman et al., 2023), CFO (Hou et al., 2026), TANTE (Wu et al., 2026)). Appendix 7.6 details datasets, adaptations, and training.

4.1 Main Results

Across all five configurations, CURVO attains the lowest trajectory-averaged relative L^2 error among the evaluated predictors without sequential rollout, including spatial-backbone predictors and spectral space–time operators (Table 1). Under the eight-query workload, it is both more accurate and faster than U-Net and MeanFlow with fewer parameters, while U-NO is faster but less accurate (Figure 1). Against the evaluated sequential predictors, CURVO yields lower mean errors on SWE, Gray–Scott, and Active Matter. On NS, AR-FNO2D remains more accurate at higher latency; on Gray–Scott,

Table 1: Predictive performance across PDE benchmarks. Relative L^2 errors evaluated as the trajectory mean (Rel. L^2) and at the final endpoint ($@H_{\max}$).

	NS, $\nu = 10^{-3}$		NS, $\nu = 10^{-4}$		SWE		Gray–Scott		Active Matter	
Method	Rel. L^2	$@H_{\max}$	Rel. L^2	$@H_{\max}$	Rel. L^2	$@H_{\max}$	Rel. L^2	$@H_{\max}$	Rel. L^2	$@H_{\max}$
Spatial-backbone direct predictors										
ResNet	0.4914	0.5838	0.4975	0.7482	0.0045	0.0039	0.2580	0.3201	<u>0.7376</u>	<u>0.8697</u>
UNet	0.0291	0.0570	0.0973	0.2830	0.0087	0.0097	<u>0.1294</u>	<u>0.1918</u>	0.7581	0.9725
UNO	0.0115	0.0368	0.0795	0.2891	<u>0.0032</u>	<u>0.0024</u>	0.1901	0.2534	0.7791	1.0222
CNO	0.0890	0.2126	0.1937	0.4677	0.0066	0.0070	0.2938	0.4544	0.7684	**0.8541**
Spectral space–time operators										
FFNO	0.0140	0.0442	0.0724	0.2737	0.0037	0.0208	0.2586	0.3217	0.7778	0.9811
FNO3D	0.3301	0.8517	0.3520	0.7601	0.0501	0.1082	0.2578	0.3196	0.9531	1.8092
Single-interval response prediction										
MeanFlow	0.0602	0.1136	0.1963	0.4245	0.0046	0.0030	0.2198	0.2814	0.7937	0.9551
Sequential evolution										
AR-FNO2D	**0.0063**	**0.0105**	**0.0327**	**0.1102**	0.0084	0.0095	0.2509	0.3967	0.9424	2.2504
Flow Matching	0.0243	0.0411	0.1093	0.3398	0.0070	0.0070	0.3076	0.4829	0.7835	1.1041
TANTE	0.0333	0.0521	0.1084	0.3167	0.0424	0.0573	–	–	0.8866	1.1961
CFO	0.0526	0.1005	1.2653	3.5868	0.0445	0.1403	0.2469	0.3847	1.8442	5.0315
CURVO (ours)	<u>0.0072</u>	<u>0.0144</u>	<u>0.0618</u>	<u>0.1943</u>	**0.0028**	**0.0022**	**0.1199**	**0.1816**	**0.7040**	0.8712

Bold and underline indicate the best and second-best results, respectively; –: training diverged.

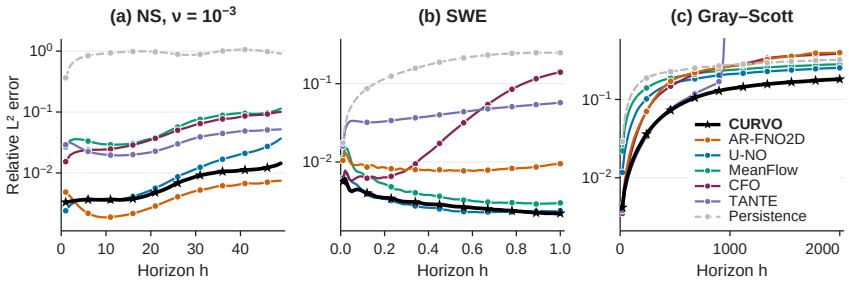

Figure 4: Horizon-wise relative L^2 error on (a) NS ($\nu = 10^{-3}$), (b) SWE, and (c) Gray–Scott, comparing CURVO with competing predictors throughout the finite prediction window.

CURVO is both more accurate and faster than every sequential alternative, and at the native 128^2 resolution its margin over FNO3D and U-NO widens to 2.5–3× (Appendix 7.7).

Figure 4 shows that this accuracy holds across the prediction window: CURVO's margin over U-NO widens at longer NS horizons, its Gray–Scott advantage persists through intermediate and late times, and its SWE errors remain close to those of the best non-sequential comparator throughout. Sustained over the whole window and obtained in a single evaluation, this accuracy makes joint prediction of the solution segment a compact and efficient alternative to sequential state construction for finite-horizon PDE simulation. Representative field trajectories across PDE families, error percentiles, and comparison methods are provided in Appendix 7.4.

4.2 Finite-Restriction Generalization

CURVO generalizes without retraining across both query cardinality and placement. Uniform, perturbed, and random layouts preserve endpoint accuracy, while denser uniform contexts maintain comparable grid-mean error (Figure 5). Concentrated layouts show greater task dependence, particularly for NS and SWE, whereas Gray–Scott is comparatively insensitive. Holding a target $h^\star$ fixed while changing only its companion queries further yields stable predictions (Appendix 7.3).

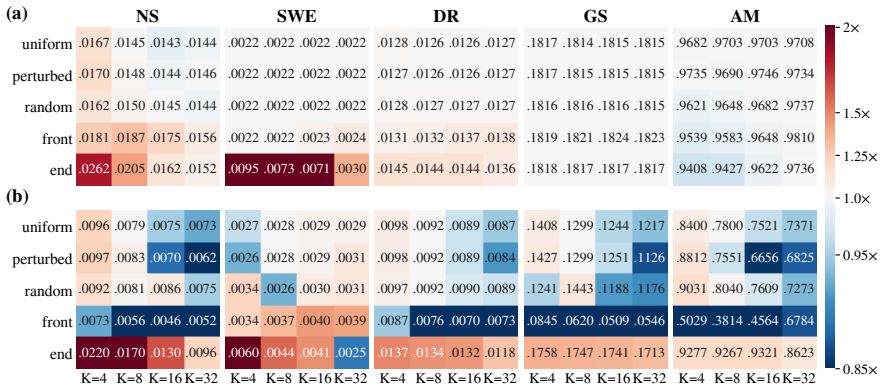

Figure 5: Query cardinality and placement transfer ($K_{\text{tr}} = 8$): (a) relative L^2 error at the common final horizon $H_{\max}$; (b) mean relative L^2 error over the queried horizons. Color indicates the ratio to the uniform K=8 context for each dataset.

Queries need not coincide with horizons seen during training. On Gray–Scott, a model trained on a subsampled temporal grid directly predicts withheld intermediate states without retraining or interpolation, with errors comparable to neighboring on-grid horizons and nearly identical at late times (Table 2). Thus inference-time query flexibility extends to unseen temporal coordinates.

Table 2: Off-grid temporal prediction on Gray–Scott. Relative L^2 errors at withheld intermediate horizons h and neighboring on-grid horizons $h \pm 10$.

Query	Requested off-grid horizon h					
	10	50	250	490	750	990
On-grid, $h - 10$	0.0042*	0.0071	0.0364	0.0766	0.1134	0.1373
Off-grid, h	0.0036	0.0102	0.0418	0.0807	0.1148	0.1380
On-grid, $h + 10$	0.0042	0.0097	0.0398	0.0798	0.1157	0.1386

*For $h = 10$, both on-grid entries use $h = 20$.

4.3 Ablation Study

Table 3: Ablations of (a) horizon coupling $\mathcal{H}_{\ell,h}$ and latent refinement $\{C_\psi^{[q]}\}_{q=0}^{L-1}$, (b) refinement depth and interval conditioning on NS, and (c) training query cardinality K_{tr} on GS.

(a) Component ablations

PDE	D-I	D-J	L-I	L-J (ours)
NS	0.010575	0.008694	0.009950	**0.007221**
GS	0.13854	0.12405	0.13645	**0.11991**
AM	0.81354	0.78443	0.81767	**0.70158**
SWE	0.00381	0.00285	0.00343	**0.00280**

D/L: direct/latent-refined;
I/J: independent/joint horizon prediction.

(b) Refinement controls (NS)

Variant	Rel. $L^2 \downarrow$
Direct (no refinement)	0.009890
Depth-matched direct[1]	0.008970
Full-h repeated[2]	0.008750
Shared refinement, $L = 4$	0.007240
Shared refinement, $L = 2$	**0.007221**

[1] Matched parameter count and depth.
[2] One shared cell applied L times at (t, h).

(c) Training query cardinality (GS)

K_{tr}	2	4	8	16
Rel. $L^2 \downarrow$	0.14810	0.13838	0.12991	**0.12435**

Gray shading marks the default configuration.

Horizon coupling improves both the direct and latent-refined predictors across all four systems, while refinement is most effective when combined with coupling: L-I only modestly improves over D-I, whereas L-J is the best variant throughout (Table 3(a)).

The refinement controls isolate the role of interval conditioning. Conditioning the shared cell on $(t + \frac{q}{L}\boldsymbol{h}, \boldsymbol{h}/L)$ reduces NS error by 17.5% relative to repeating it at the full coordinates $(t, \boldsymbol{h})$; matching depth and parameter count without refinement closes even less of the gap (Table 3(b)). Two and four stages give nearly identical errors, supporting the default choice of $L = 2$.

Training cardinality matters less: on GS, error decreases monotonically from $K_{\text{tr}} = 2$ to 16 under a fixed step budget, and doubling the supervised queries beyond the default $K_{\text{tr}} = 8$ yields a further 4.3% (Table 3(c)). Overall, the largest gains arise from horizon coupling and interval-conditioned refinement rather than added depth, repeated computation, or increased training cardinality.

Extension to image generation. On unconditional CIFAR-10, CURVO achieves an FID of 9.24 with a single network evaluation and without a pretrained teacher or trajectory distillation, while jointly producing the terminal sample and intermediate transport states. For comparison, iCT attains the best FID of 2.83 but predicts only the terminal sample; Flow Matching and Score Matching achieve 6.35 and 19.94 with 142 and 242 evaluations, respectively, while one-step DDIM Distillation reaches 9.36 through teacher distillation. The same CURVO checkpoint also supports inference-time changes in query cardinality: trained with $K_{\text{tr}} = 4$, it operates at $K = 2$–32, with FID changing from 9.24 to 9.83. Full comparisons and trajectory visualizations are provided in Appendix 7.1.

5 Discussion

CURVO is strongest on Gray–Scott, shallow water, and Active Matter, whereas AR-FNO2D remains more accurate on Navier–Stokes. This pattern suggests that cross-horizon coupling is most useful when future states share exploitable trajectory structure, while strong local transition models can remain advantageous when short-step evolution is highly predictive. Query-layout results show a related dependence: concentrated contexts reduce accuracy on NS and SWE, whereas controlled changes to companion queries leave a fixed target largely stable (Appendix 7.3).

Increasing query cardinality K enlarges the horizon dimension of the latent operator, increasing spectral computation and introducing quadratic attention cost in K. CURVO limits this growth through factorized spatial–horizon spectral operators at higher resolutions and bottleneck-only physical-coordinate attention, supporting denser query sets at inference (Appendix 7.2).

The current formulation is deterministic, conditioned on a single state, and operates within the trained horizon window. Short histories, physical parameters, and distribution-valued responses provide natural extensions to partially observed, parameterized, and stochastic dynamics. The CIFAR-10 results further show that the same horizon-coupled representation transfers to generative transport, where intermediate transport states and the terminal sample are produced jointly with the query set chosen after training. Finite evolution segments, rather than next states or endpoints alone, may thus serve as the prediction object for a broader class of trajectory-valued problems.

6 Conclusion

We introduced CURVO, a horizon-coupled neural operator that treats prediction horizons as operator coordinates and represents finite-time evolution through interval-average response curves. A multiscale spatial–horizon architecture with shared interval-conditioned refinement jointly models the queried trajectory, enabling all requested states to be reconstructed from a single anchor in one forward pass. Across five PDE configurations, CURVO achieves the lowest trajectory-averaged error among the evaluated one-pass methods in a comparison spanning established neural-operator baselines and recent state-of-the-art temporal predictors, while supporting sparse, dense, irregular, and off-grid readouts without retraining. The same formulation also extends to teacher-free one-step CIFAR-10 generation with joint intermediate transport states. CURVO thus shifts finite-time PDE prediction from state propagation on a prescribed temporal grid to evaluation of a coupled trajectory operator at inference-chosen horizons.

AI USE STATEMENT

We used generative AI tools to provide feedback on experimental design, assist in interpreting results, refine conceptual framing and terminology, and check mathematical statements and derivations. We did not use these tools to generate data, implement methods, or develop critical ingredients for proofs. We also used them to assist with drafting and editing text, proposing titles, identifying and formatting references, and suggesting document structure. All AI-assisted content was reviewed by the authors; numerical claims were checked against experiment logs, and AI-suggested references were verified against the original sources. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Victor Armegioiu, Yannick Ramic, and Siddhartha Mishra. Rectified flows for fast multiscale fluid flow modeling. *arXiv preprint arXiv:2506.03111*, 2025.

Marin Biloš, Johanna Sommer, Syama Sundar Rangapuram, Tim Januschowski, and Stephan Günnemann. Neural flows: Efficient alternative to neural odes. In *Advances in Neural Information Processing Systems*, volume 34, pp. 21325–21337, 2021.

Johannes Brandstetter, Daniel Worrall, and Max Welling. Message passing neural PDE solvers. In *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2202.03376.

Edoardo Calvello, Nikola B. Kovachki, Matthew E. Levine, and Andrew M. Stuart. Continuum attention for neural operators. *Journal of Machine Learning Research*, 26(300):1–52, 2025.

Valérie Castin, Pierre Ablin, and Gabriel Peyré. How Smooth Is Attention? In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 5817–5840. PMLR, 2024. URL https://proceedings.mlr.press/v235/castin24a.html.

Junfeng Chen and Kailiang Wu. Positional knowledge is all you need: Position-induced transformer (PiT) for operator learning. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, 2024.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David K. Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, volume 31, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/69386f6bb1dfed68692a24c8686939b9-Abstract.html.

Zhengyang Geng, Mingyang Deng, Xingjian Bai, J. Zico Kolter, and Kaiming He. Mean Flows for One-step Generative Modeling. In *Advances in Neural Information Processing Systems*, 2025.

Luca Giomi, Mark J. Bowick, Xu Ma, and M. Cristina Marchetti. Defect annihilation and proliferation in active nematics. *Physical Review Letters*, 110(22):228101, 2013. doi: 10.1103/PhysRevLett.110.228101. URL https://arxiv.org/abs/1303.4720.

Jan Hagnberger, Marimuthu Kalimuthu, Daniel Musekamp, and Mathias Niepert. Vectorized conditional neural fields: A framework for solving time-dependent parametric partial differential equations. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 17189–17223. PMLR, 2024.

Ernst Hairer, Syvert P. Nørsett, and Gerhard Wanner. *Solving Ordinary Differential Equations I: Nonstiff Problems*, volume 8 of *Springer Series in Computational Mathematics*. Springer, 2 edition, 1993. doi: 10.1007/978-3-540-78862-1.

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

Maximilian Herde, Bogdan Raonić, Tobias Rohner, Roger Käppeli, Roberto Molinaro, Emmanuel de Bézenac, and Siddhartha Mishra. Poseidon: Efficient foundation models for PDEs. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-2311. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/84e1b1ec17bb11c57234e96433022a9a-Abstract-Conference.html.

Alan C. Hindmarsh, Peter N. Brown, Keith E. Grant, Steven L. Lee, Radu Serban, Dan E. Shumaker, and Carol S. Woodward. SUNDIALS: Suite of nonlinear and differential/algebraic equation solvers. *ACM Transactions on Mathematical Software*, 31(3):363–396, 2005. doi: 10.1145/1089014.1089020.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising Diffusion Probabilistic Models. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851. Curran Associates, Inc., 2020.

Xianglong Hou, Xinquan Huang, and Paris Perdikaris. CFO: Learning continuous-time PDE dynamics via flow-matched neural operators. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/8bfbf4ec87e1e331f0b1adc483b53b6b-Abstract-Conference.html.

Peiyan Hu, Rui Wang, Xiang Zheng, Tao Zhang, Haodong Feng, Ruiqi Feng, Long Wei, Yue Wang, Zhi-Ming Ma, and Tailin Wu. Wavelet diffusion neural operator. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/20eadb6338a53114a6371f102c5ded95-Abstract-Conference.html.

Yifan Jiang, Shiyu Chang, and Zhangyang Wang. TransGAN: Two Pure Transformers Can Make One Strong GAN, and That Can Scale Up. In *Advances in Neural Information Processing Systems*, 2021.

Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training Generative Adversarial Networks with Limited Data. In *Advances in Neural Information Processing Systems*, volume 33. Curran Associates, Inc., 2020.

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. URL https://openreview.net/forum?id=c8P9NQVtmnO.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow Matching for Generative Modeling. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow Straight and Fast: Learning to Generate and Transfer Data with Rectified Flow. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=XVjTT1nw5z.

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. doi: 10.1038/s42256-021-00302-5.

Eric Luhman and Troy Luhman. Knowledge distillation in iterative generative models for improved sampling speed. *arXiv preprint arXiv:2101.02388*, 2021.

Suryanarayana Maddu, Scott Weady, and Michael J. Shelley. Learning fast, accurate, and stable closures of a kinetic theory of an active fluid. *Journal of Computational Physics*, 504:112869, 2024.

Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral Normalization for Generative Adversarial Networks. In *International Conference on Learning Representations*, 2018.

Sepehr Mousavi, Shizheng Wen, Levi Lingsch, Maximilian Herde, Bogdan Raonić, and Siddhartha Mishra. RIGNO: A graph-based framework for robust and accurate operator learning for PDEs on arbitrary domains. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://arxiv.org/abs/2501.19205.

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 H. 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 (Datasets and Benchmarks Track)*, volume 37, 2024.

John E. Pearson. Complex patterns in a simple system. *Science*, 261(5118):189–192, 1993. doi: 10.1126/science.261.5118.189. URL https://arxiv.org/abs/patt-sol/9304003.

Md Ashiqur Rahman, Zachary E. Ross, and Kamyar Azizzadenesheli. U-NO: U-shaped neural operators. *Transactions on Machine Learning Research*, 2023. URL https://openreview.net/forum?id=j3oQF9coJd.

Bogdan Raonić, 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, 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)*, pp. 234–241, 2015.

Salva Rühling Cachay, Bo Zhao, Hailey Joren, and Rose Yu. DYffusion: A dynamics-informed diffusion model for spatiotemporal forecasting. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-1961. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/8df90a1440ce782d1f5607b7a38f2531-Abstract-Conference.html.

Tim Salimans and Jonathan Ho. Progressive Distillation for Fast Sampling of Diffusion Models. In *International Conference on Learning Representations*, 2022.

Yang Song and Prafulla Dhariwal. Improved Techniques for Training Consistency Models. In *The Twelfth International Conference on Learning Representations*, 2024.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-Based Generative Modeling through Stochastic Differential Equations. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=PxTIG12RRHS.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency Models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 32211–32252. PMLR, 2023. URL https://proceedings.mlr.press/v202/song23a.html.

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

Alasdair Tran, Alexander Mathews, Lexing Xie, and Cheng Soon Ong. Factorized fourier neural operators. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2111.13802.

James Vuckovic, Aristide Baratin, and Remi Tachet des Combes. A Mathematical Theory of Attention. *arXiv preprint arXiv:2007.02876*, 2020.

Sifan Wang, Jacob H. Seidman, Shyam Sankaran, Hanwen Wang, George J. Pappas, and Paris Perdikaris. CViT: Continuous vision transformer for operator learning. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2405.13998.

Tailin Wu, Takashi Maruyama, Qingqing Zhao, Gordon Wetzstein, and Jure Leskovec. Learning controllable adaptive simulation for multi-resolution physics. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2305.01122.

Tailin Wu, Takashi Maruyama, Long Wei, Tao Zhang, Yilun Du, Gianluca Iaccarino, and Jure Leskovec. Compositional generative inverse design. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2401.13171.

Zhikai Wu, Sifan Wang, Shiyang Zhang, Sizhuang He, Min Zhu, Anran Jiao, Lu Lu, and David van Dijk. TANTE: Time-adaptive operator learning via neural Taylor expansion. *Journal of Computational Physics*, 562:115041, 2026. doi: 10.1016/j.jcp.2026.115041. URL https://www.sciencedirect.com/science/article/pii/S0021999126003943.

Yi Zhang and Difan Zou. Physics-informed distillation of diffusion models for PDE-constrained generation. In *Proceedings of the 43rd International Conference on Machine Learning (ICML)*, 2026.

Shengyu Zhao, Zhijian Liu, Ji Lin, Jun-Yan Zhu, and Song Han. Differentiable Augmentation for Data-Efficient GAN Training. In *Advances in Neural Information Processing Systems*, volume 33. Curran Associates, Inc., 2020.

Hongkai Zheng, Weili Nie, Arash Vahdat, Kamyar Azizzadenesheli, and Anima Anandkumar. Fast sampling of diffusion models via operator learning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 42390–42402. PMLR, 2023. URL https://proceedings.mlr.press/v202/zheng23d.html.

Kewei Zhu, Yanze Xin, Jinwei Hu, Xiaoyuan Cheng, Yiming Yang, and Sibo Cheng. Spatiotemporal system forecasting with irregular time steps via masked autoencoder. *Physica D: Nonlinear Phenomena*, 492:135189, 2026. doi: 10.1016/j.physd.2026.135189.

7 Appendix

7.1 Extension to Image Generation

CURVO extends to unconditional image generation without a pretrained teacher, trajectory distillation, or cached solver paths. Distilled one-step generators obtain fast sampling by transferring a pretrained multi-step model, while teacher-free endpoint methods learn individual finite transitions. CURVO instead learns directly from sampled data–noise pairs and, in one network evaluation, returns the states of a jointly estimated finite restriction of the transport response curve together with the terminal sample. The same checkpoint changes the number of requested intermediate states after training. CIFAR-10 therefore tests a distinct combination of properties: teacher-free learning, one-step generation, joint trajectory readout, and post-training query reconfiguration.

Table 4: **Training requirements and inference capabilities.** The comparison separates training dependencies from the prediction object available at inference.

	Training requirements			Inference capabilities		
Method	Teacher	Distill.	Cached traj.	1 eval.	Joint traj.	Reconfig. K
Multi-step diffusion / flow	×	×	×	×	×	–
Consistency Distillation	✓	✓	×	✓	×	–
DSNO	✓	✓	✓	✓	✓	×
Teacher-free endpoint / interval	×	×	×	✓	×	–
GAN	×	×	×	✓	–	–
CURVO	×	×	×	✓	✓	✓

"Joint traj." denotes joint readout of multiple transport states in one network evaluation; "Reconfig. K" denotes changing the cardinality of this joint readout after training; –: not applicable, as there is no joint readout to reconfigure.

Table 5: **Unconditional generation on CIFAR-10.** FID and network evaluations (NFE) for representative generative models.

Method	TF	NFE	FID ↓
Direct generation (multi-step)			
Flow Matching† (Lipman et al., 2023)	✓	142	6.35
DDPM† (Ho et al., 2020)	✓	274	7.48
Score Matching† (Song et al., 2021)	✓	242	19.94
Distillation (1-step student)			
Consistency Distillation (LPIPS) (Song et al., 2023)	×	1	3.55
DSNO (LPIPS) (Zheng et al., 2023)	×	1	3.78
2-Rectified Flow (+Distill) (Liu et al., 2023)	×	1	4.85
Progressive Distillation (Salimans & Ho, 2022)	×	1	9.12
Knowledge Distillation (Luhman & Luhman, 2021)	×	1	9.36
Endpoint / interval models			
iCT (Song & Dhariwal, 2024)	✓	1	2.83
MeanFlow* (Geng et al., 2025)	✓	1	3.81
Consistency Training (Song et al., 2023)	✓	1	8.70
GAN			
StyleGAN2 + ADA (Karras et al., 2020)	✓	1	2.92
StyleGAN2 + DiffAug (Zhao et al., 2020)	✓	1	5.79
TransGAN + DiffAug (Jiang et al., 2021)	✓	1	9.26
SNGAN (Miyato et al., 2018)	✓	1	21.70
CURVO (ours)	✓	1	9.24

TF denotes training without a pretrained generator or offline teacher trajectories. †: as reported by Lipman et al. (2023); *: our run with the official code; all other results are taken from the respective publications.

CURVO achieves an FID of 9.24 with a single network evaluation (Table 5); its operating point, teacher-free, one-step, and trajectory-valued, is the one Table 4 isolates.

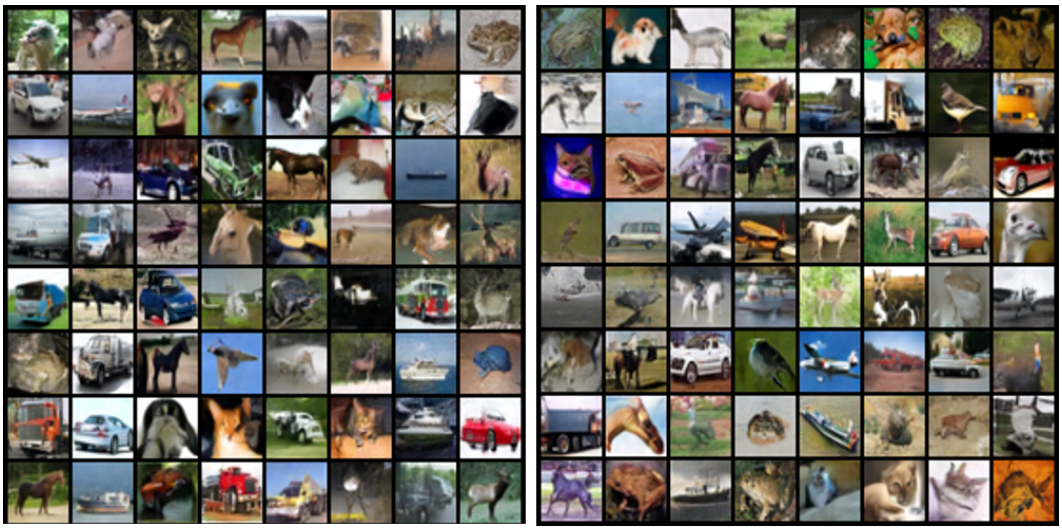

Figure 6: Uncurated CIFAR-10 samples generated by CURVO in one network evaluation.

The same checkpoint preserves inference-time query reconfiguration. Trained only with $K_{\text{tr}} = 4$, it is evaluated at $K \in \{2, 4, 8, 16, 32\}$ with unchanged parameters; an eightfold densification from K=4 to K=32 changes FID only from 9.24 to 9.83 (Table 6). Figures 7 and 8 show this at the trajectory level, and Figure 6 shows uncurated samples. Joint estimation of the response curve and reconfiguration of its finite restriction thus carry over from PDE evolution to generative transport.

Table 6: **Query-cardinality transfer on CIFAR-10.** A checkpoint trained with $K_{\text{tr}} = 4$ is evaluated at different query cardinalities with unchanged parameters.

K	2	4†	8	16	32
FID ↓	9.20	9.24	9.29	9.48	9.83

† Matches K_{tr}.

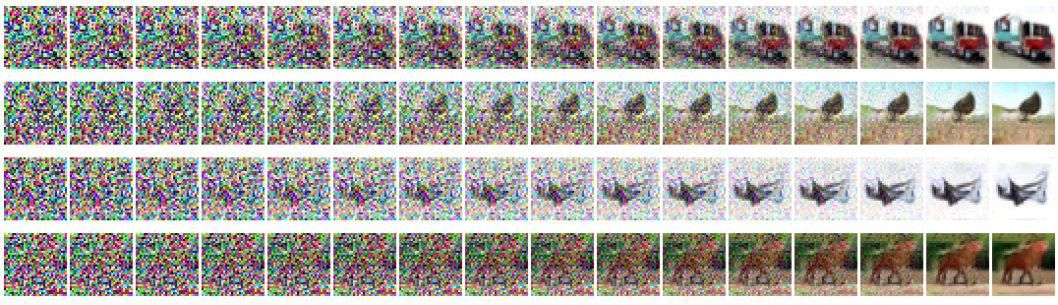

Figure 7: Generative trajectories from a checkpoint trained with $K_{\text{tr}} = 4$ and evaluated at $K = 16$. Each row is jointly reconstructed from one noise anchor in a single forward pass.

The image model retains the spectral horizon branch, the physical-coordinate attention, and the shared interval-conditioned refinement used for PDE prediction; its principal change is increased backbone capacity (Table 7). CIFAR-10 is modeled at its native 32×32 resolution with RGB values scaled to $[-1, 1]$. Query contexts are sampled continuously rather than from a recorded grid: the anchor time and the time offsets follow logit-normal distributions with location -2 and scale 2, ordered within the transport interval, and with probability 0.25 the shortest offset is collapsed to the instantaneous



Figure 8: One noise anchor read at $K \in \{2, 4, 8, 16, 32\}$ by the $K_{\text{tr}} = 4$ checkpoint; each row is one forward pass. All cardinalities trace the same transport path and agree at the endpoint.

limit $h \to 0$. The adaptive reweighting of Equation equation 11 uses $p = 0.75$ and $\varepsilon = 10^{-3}$. These changes adapt the data and query distribution while leaving the trajectory operator unchanged.

Table 7: **Architecture changes for CIFAR-10.** The image model scales the backbone while retaining the same horizon-coupled operator blocks and interval-conditioned refinement.

Setting	PDE model	CIFAR-10 model
Channel widths	32/64/128	256/256/256
Encoder / decoder blocks	2	3
Bottleneck coupled blocks	16	6
Attention heads	1	1
Refinement steps L	2	2
Parameters	4.94M	54.49M
Training queries K_{tr}	8	4

7.2 Efficiency Scaling with Query Cardinality

We measure the inference cost of CURVO as the queried trajectory becomes denser. Using the final Gray–Scott checkpoint, we vary K from 4 to 100 while fixing the batch size, spatial resolution, precision, and hardware; the timing protocol matches the main-text latency measurements. Table 8 shows that the same model jointly returns every recorded state of the $H = 100$ window in one forward pass. The full-grid readout takes 467.5 ms and 3.43 GB of peak memory, while per-state latency remains approximately stable from $K = 8$ onward.

Table 8: **Inference scaling with query cardinality.** The final CURVO checkpoint is evaluated on Gray–Scott with batch size 8, 64^2 resolution, and fp32 precision on one A100-PCIE-40GB. $K = 100$ queries every recorded state over the full $H = 100$ window.

K_{eval}	4	8	16	32	100
States returned	4	8	16	32	100
Latency (ms)	32.7	34.7	65.8	158.0	467.5
Peak memory (GB)	0.22	0.35	0.62	1.15	3.43

This scaling is consistent with the design of Section 3.2. At the higher-resolution stages, spatial and horizon interactions are modeled with factorized spectral operators rather than a joint spectral tensor over space and horizon. Physical-coordinate attention $\mathcal{A}_h$ is restricted to the bottleneck, so its quadratic interaction in K acts only on the lowest-resolution latent field.

Table 9 compares CURVO with independent parallel queries using MeanFlow under the same protocol. CURVO is faster from $K = 8$ onward and uses less peak memory at every tested cardinality.

Table 9: **Coupled and independent multi-query inference.** CURVO and MeanFlow are evaluated on Navier–Stokes with the same batch size, spatial resolution, precision, and hardware. MeanFlow evaluates the requested horizons independently in parallel, whereas CURVO processes them as one coupled query context.

	K	4	8	16	32
CURVO	Latency (ms)	39.6	43.5	77.0	150.7
	Peak memory (GB)	0.22	0.35	0.62	1.15
MeanFlow	Latency (ms)	28.9	54.5	104.5	204.6
	Peak memory (GB)	0.63	1.01	1.76	3.26

Denser readout therefore increases computation and memory, while the requested states remain jointly evaluated in a single forward pass. The same checkpoint scales from sparse query sets to the complete 100-state trajectory without changing the model or inference procedure.

7.3 Fixed-Target Sensitivity to Query Context

We evaluate whether a prediction at a fixed target horizon $h^\star$ remains stable when only its companion queries are changed. For each (anchor, $h^\star$) pair, we draw $M = 16$ nested contexts $\Xi_4 \subset \Xi_8 \subset \Xi_{16} \subset \Xi_{32}$, all containing the same $h^\star$. We use 200 Navier–Stokes ($\nu = 10^{-3}$) anchors with $h^\star \in \{12, 25, 49\}$ and 120 Gray–Scott anchors with $h^\star \in \{25, 50, 100\}$.

For a prediction $\widehat{u}_K(h^\star)$ obtained from a context of cardinality K, E_K denotes its relative L^2 error at the fixed target, while D_K measures its relative deviation from the $K_{\text{tr}} = 8$ prediction under the same nested context. S_8 measures variation across independently sampled $K = 8$ companion-query sets. All quantities are averaged over anchors and context draws.

Table 10: **Fixed-target sensitivity to query context.** E_K is the relative error at the same target $h^\star$; D_K is the prediction drift from the training cardinality $K_{\text{tr}} = 8$; S_8 measures variation across different $K = 8$ companion-query sets. Context-sensitivity values are reported in units of 10^{-3}.

		Fixed-target error E_K				Context sensitivity ($\times 10^{-3}$)			
Dataset	$h^\star$	K=4	K=8	K=16	K=32	D_4	D_{16}	D_{32}	S_8
NS (10^{-3})	12	.0039	.0037	.0037	.0037	1.20	0.69	0.81	0.72
	25	.0083	.0068	.0064	.0063	4.20	1.90	2.20	2.10
	49	.0204	.0154	.0146	.0144	11.0	4.80	5.70	5.00
Gray–Scott	25	.0797	.0794	.0793	.0792	6.70	5.20	6.10	5.40
	50	.1378	.1375	.1373	.1372	9.80	7.20	8.40	7.70
	100	.1816	.1815	.1815	.1815	9.30	7.10	9.00	7.10

Table 10 shows that the same physical target remains stable under substantial reconfiguration of its finite query context. Gray–Scott is nearly unchanged from $K = 4$ to $K = 32$. Navier–Stokes exhibits greater context dependence at longer horizons, while denser contexts improve fixed-target accuracy beyond the training cardinality. In both systems, context-induced variation remains smaller than the underlying prediction error, supporting reconfigurable finite-restriction readout from a single learned operator.

7.4 Representative Trajectory Predictions

We complement the aggregate metrics with trajectory-level predictions over the queried finite interval. Figure 9 shows median-error examples from four PDE families with distinct spatial dynamics. From a single observed anchor, CURVO jointly reconstructs the requested solution segment at multiple horizons, capturing advective evolution, expanding wave structure, reaction–diffusion patterns, and active-matter dynamics within the same finite-restriction formulation.

Figure 10 resolves this behavior across the Gray–Scott test-error distribution. The q_{25} and q_{50} trajectories retain the evolving reaction–diffusion morphology throughout the interval, while the q_{90}

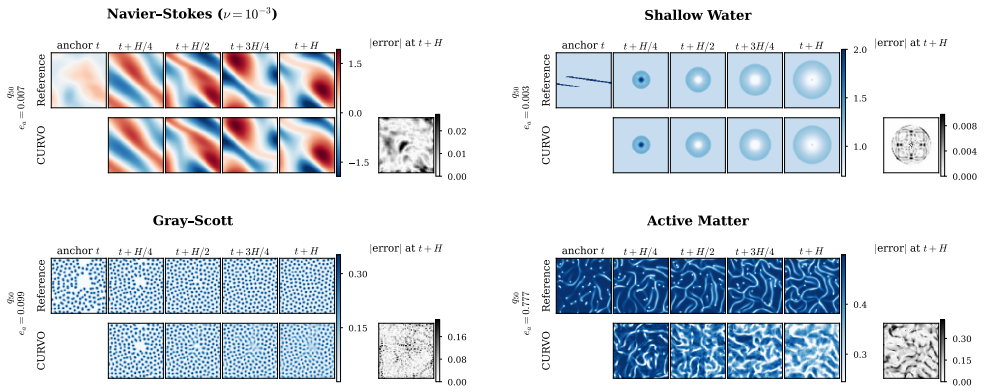

Figure 9: **Representative trajectories across PDE families.** Median-error test examples for Navier–Stokes ($\nu = 10^{-3}$), Shallow Water, Gray–Scott, and Active Matter. Each panel shows the reference solution, CURVO prediction from the same anchor at four horizons, and the absolute endpoint error at $t + H$.

example develops progressively larger structural mismatch at later horizons. The sequence exposes how prediction error evolves over the solution segment rather than only at its endpoint.

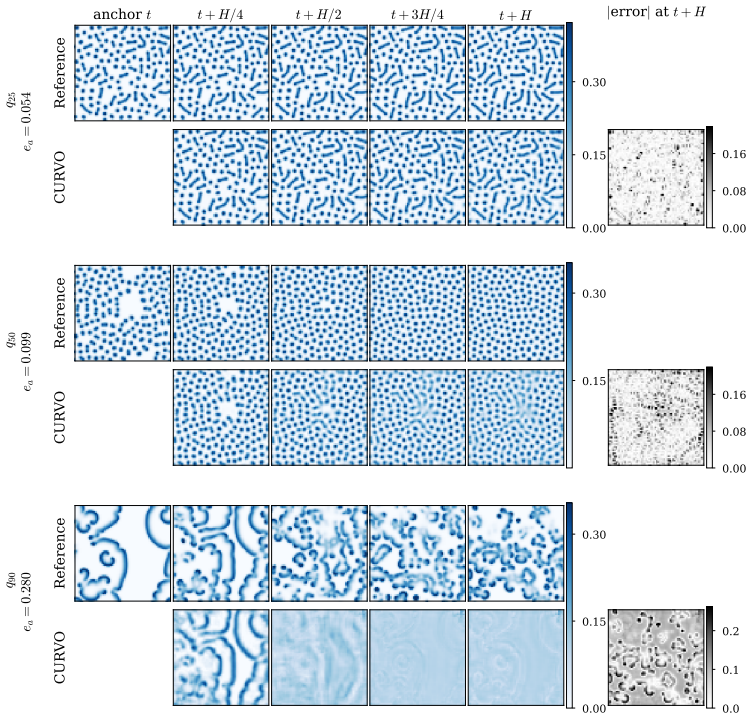

Figure 10: **Gray–Scott trajectories across prediction difficulty.** Examples nearest the 25th, 50th, and 90th percentiles of CURVO trajectory-averaged relative L^2 error are shown from top to bottom. All predictions use the same queried horizons, with the absolute endpoint error shown in the final column.

Figure 11 compares the same median Gray–Scott trajectory across representative direct, interval-based, and sequential predictors. CURVO attains the lowest trajectory error among the displayed methods and preserves the reaction–diffusion structure across the queried interval. The comparison complements the main quantitative results by showing how differences in trajectory error appear spatially and accumulate across horizons.

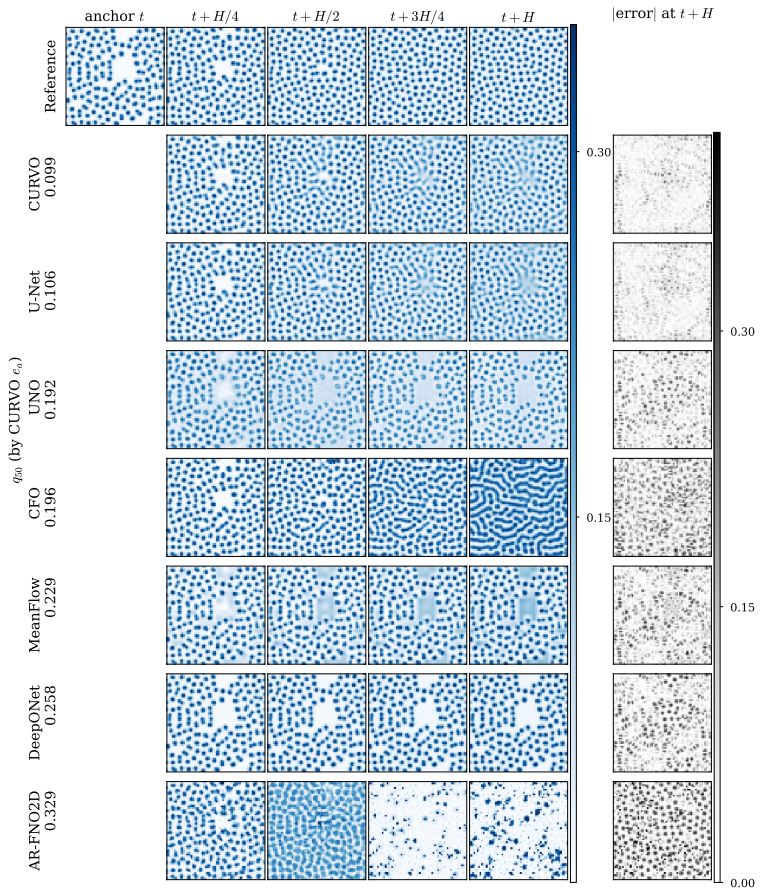

Figure 11: **Qualitative comparison on Gray–Scott.** The same median-error trajectory and queried horizons are evaluated with CURVO, U-Net, UNO, CFO, MeanFlow, DeepONet, and AR-FNO2D. Numbers beside each method report trajectory-averaged relative L^2 error; the final column shows the absolute endpoint error using a common scale.

7.5 Separation Dimension of Finite-Interval Responses

Recall $b_\rho(h) = (1 - e^{-\rho h})/h$ for $\rho > 0$ and $h > 0$, so that $\widehat{\dot{v}}(\omega, t, h) = -b_\rho(h)\widehat{u}(\omega, t)$.

Proof of Proposition 1. Let $\rho_1, \dots, \rho_n > 0$ be distinct and suppose $\sum_{i=1}^n c_i b_{\rho_i}(h) = 0$ for $h \in I$. Multiplying by h and differentiating gives

$$\sum_{i=1}^{n} c_i \rho_i e^{-\rho_i h} = 0, \qquad h \in I. \tag{13}$$

Fix $h_\star \in I$. Differentiating Equation equation 13 $a = 0, \dots, n-1$ times at $h_\star$ yields

$$\sum_{i=1}^{n} (-\rho_i)^a c_i \rho_i e^{-\rho_i h_\star} = 0, \qquad a = 0, \dots, n-1.$$

This is a Vandermonde system in $d_i = c_i \rho_i e^{-\rho_i h_\star}$. Since the ρ_i are distinct, the Vandermonde matrix is invertible, hence $d_i = 0$ and therefore $c_i = 0$ for every i. Thus $b_{\rho_1}, \dots, b_{\rho_n}$ are linearly independent on I.

If $b_{\rho_i}(h) = \sum_{a=1}^R c_{ia} g_a(h)$ with $g_1, \dots, g_R$ independent of ρ, then

$$\operatorname{span}\{b_{\rho_1}, \dots, b_{\rho_n}\} \subseteq \operatorname{span}\{g_1, \dots, g_R\}.$$

The left-hand side has dimension n, while the right-hand side has dimension at most R, so $R \geq n$. Since arbitrarily many distinct rates can be chosen from any nondegenerate interval of positive rates, no fixed finite ρ-independent temporal basis represents the full response family exactly. □

For finitely sampled rates and horizons, the corresponding approximation dimension is characterized by the singular spectrum of the response matrix. Given positive weights p_i, q_j, define $B_{ij} = \sqrt{p_i}\, b_{\rho_i}(h_j)\sqrt{q_j}$ and let $\sigma_1 \geq \cdots \geq \sigma_m \geq 0$, $m = \min\{n, K\}$, denote its singular values. By the Eckart–Young–Mirsky theorem,

$$\min_{\operatorname{rank}(C) \leq R} \|B - C\|_F^2 = \sum_{j > R} \sigma_j^2. \tag{14}$$

Hence, for a relative tolerance $\varepsilon \in (0, 1)$, the effective separation rank is

$$r_\varepsilon = \min\left\{ R : \frac{\sum_{j>R} \sigma_j^2}{\sum_{j=1}^m \sigma_j^2} \leq \varepsilon^2 \right\}. \tag{15}$$

The same diagnostic applies to nonlinear recorded responses by flattening spatial and field-component indices into α and forming $D_{\alpha k} = \sqrt{w_\alpha}\, \bar{v}_\alpha^{z,\star}(t, h_k)\sqrt{q_k}$. Its singular spectrum gives the effective horizon dimension in the chosen weighted normalized-response metric.

Table 11: Dataset protocols. T is the number of retained snapshots, Δt their temporal spacing, and H the maximum prediction horizon in retained steps.

Dataset	Source	Grid (native→used)	C	T	Δt	H	Train/Val/Test
NS ν=10^{-3}	Li et al. (2021)	64^2	1	50	1.0	49	1000/100/200
NS ν=10^{-4}	Li et al. (2021)	64^2	1	30	1.0	29	9000/1000/1000
SWE	PDEBench	$128^2 \to 64^2$	1	101	0.01	100	57/8/16 ICs
Gray–Scott	The Well	$128^2 \to 64^2$	2	200	20	100	960/120/120
Active Matter	The Well	$256^2 \to 64^2$	4	81	0.25	40	175/24/26

7.6 Experimental Details

Datasets. Table 11 summarizes the five benchmark configurations. Where the working grid differs from the native resolution, we use the data source's prescribed downsampling, namely stride-2 for PDEBench and the official MiniWell operator for The Well. Gray–Scott keeps every second snapshot

($\Delta t : 10 \to 20$), which also provides the withheld frames used in the off-grid study, and combines six (F, k) regimes. SWE splits are stratified over its 81 initial conditions, while Active Matter uses parameter-disjoint splits. Normalization is applied channelwise,

$$z = \frac{u - \mu}{s + 10^{-8}},$$

with (μ, s) estimated from the training split only.

Table 12: Optimization settings shared by CURVO and all retrained baselines.

Optimizer	AdamW	Iterations	150k
(β_1, β_2)	$(0.9, 0.999)$	LR schedule	cosine
Learning rate	10^{-3}	Batch size	8
Weight decay	10^{-5}	K_{tr}	8
Loss (ε, p)	$(10^{-3}, 0.75)$	Seed	42

Training. For each sample, K_{tr} horizons are drawn uniformly without replacement from $\{1, \ldots, H\}$ and sorted. The anchor is then sampled uniformly among all time indices for which every queried endpoint is available. All retrained models follow the optimization settings in Table 12. CURVO is trained with the loss in Equation equation 11 on normalized interval-average responses. All reported experiments use the final training checkpoint. Table 13 summarizes the CURVO architecture, which is shared across datasets apart from the physical channel count C and horizon scale H.

Table 13: CURVO architecture (4.94M parameters at C=1).

Levels (resolution)	64/32/16	Widths	32/64/128
Blocks per level	2	Bottleneck coupled blocks	16
Coupled-block levels	32^2 (encoder and decoder)	Horizon modes $\lvert\mathcal{M}_\ell\rvert$	3
Spatial modes ($\mathcal{K}_{\ell,x}$)	6×6	Spatial spectral weights	shared
Spatial self-attention	16^2 level	Coordinate attention ($\mathcal{A}_h$)	bottleneck, horizon axis
Interval embedding $\eta(t, h)$	sinusoidal	Refinement steps L	2
Horizon scale	H		

Baselines. Direct prediction baselines receive (u_t, t, h) and are trained on the same normalized finite-horizon target under Table 12. Official implementations are used where released. FNO3D, F-FNO, and AR-FNO2D follow the original FNO codebase, and ResNet is the 18-convolution baseline of Li et al. (2021). Independent-query models append (t, h) to the input and process the K requested horizons in parallel. Sequential models retain their native temporal formulation, roll out internally, and are read out at the requested horizons. Architecture hyperparameters are kept fixed across datasets rather than tuned separately for each benchmark.

Table 14: Baseline architectures. Parameter counts are reported at C=1 on NS. NFE denotes the number of sequential network calls per anchor.

Method	Architecture	Params	NFE
FNO3D	width 32, 4 layers, modes (4, 12, 12), padding 6	18.9M	1
F-FNO	width 128, 24 layers, modes (4, 16, 16), shared	4.4M	1
UNO	official 2-D core, width 32	30.5M	1
CNO	official core, width 16, 4 layers	2.0M	1
U-Net	conv U-Net, width 64, 2 blocks/level	21.1M	1
ResNet	width 43, 8 residual blocks (18 conv)	0.27M	1
MeanFlow	SongUNet, width 64	16.4M	1
AR-FNO2D	width 32, 4 layers; autoregressive	2.4M	$h/\Delta t$
Flow Matching	SongUNet, width 64; Euler	16.4M	$h/\Delta t$
TANTE	official, embed 256, 8 heads	4.0M	$h/\Delta t$
CFO	official U-Net; RK4, 2 steps/interval	8.1M	$8h/\Delta t$

Evaluation. All models are evaluated on frozen test sets with fixed trajectories, anchors, and query grids. Predicted normalized interval responses are mapped back to physical state space as

$$\hat{u}(t+h) = u_t + h\, s \odot \hat{v}^{\,z},$$

where s is the channelwise standard deviation estimated from the training split. The reported relative L^2 error is computed per sample in physical state space. For the main results, all recorded horizons between the anchor and $H_{\max}$ are evaluated, while the ablations use a fixed eight-horizon grid. Rel. L^2 averages the per-sample error over the evaluation horizons, and $@H_{\max}$ reports the error at the final horizon.

Latency is measured with CUDA events after 10 warm-up runs over 100 timed runs at batch size 8, K=8, and fp32 on one shared device within each comparison. NFE measures sequential network depth rather than total arithmetic cost, and complex-valued parameters are counted as two real parameters.

7.7 Higher Spatial Resolution

Table 15: Gray–Scott at native 128^2 resolution using the same test trajectories and anchors as the 64^2 benchmark.

Method	Macro (K=8)	$@H_{\max}$	Macro (full grid)	Params
CURVO	**0.1170**	**0.1727**	**0.1073**	7.83M
FNO3D	0.2791	0.3282	–	75.5M
U-NO	0.3543	0.5333	0.3249	30.5M

To test whether CURVO's performance depends on the 64^2 working resolution used in the main experiments, we retrain CURVO, FNO3D, and U-NO on Gray–Scott at its native 128^2 resolution. CURVO extends the multiscale hierarchy to 128/64/32/16 while retaining horizon coupling at the two coarsest resolutions; FNO3D increases its spatial Fourier modes from 12 to 24, and U-NO is unchanged. All models use the same test trajectories, anchors, and 150k-step training budget as in the main benchmark.

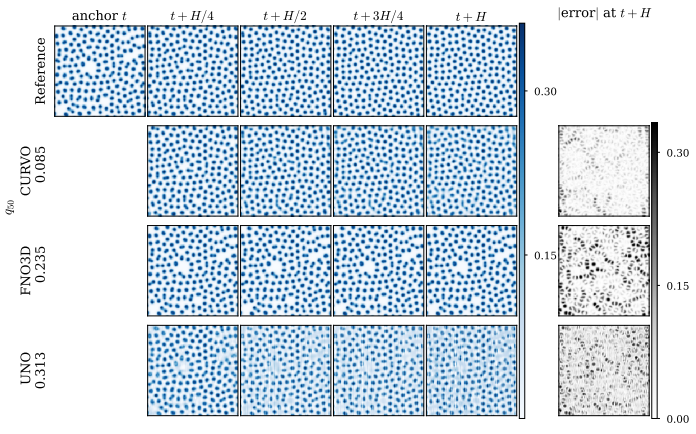

Figure 12: Representative Gray–Scott trajectory at native 128^2 resolution. Rows compare CURVO, FNO3D, and U-NO against the reference trajectory from the same anchor, with endpoint absolute error shown at right.

As shown in Table 15, CURVO remains the most accurate model at native resolution, achieving a trajectory-averaged relative L^2 error of 0.1170 under the standard $K = 8$ query protocol, compared with 0.2791 for FNO3D and 0.3543 for U-NO. Under full-grid readout, CURVO reaches 0.1073 and predicts all 100 recorded states in a single forward pass, using 13.5 GB of peak memory. The qualitative comparison in Figure 12 shows the same pattern across the trajectory: CURVO preserves the evolving Gray–Scott structures more faithfully and yields substantially smaller endpoint errors than the two comparison models.