LEARNING CHAOS WITHOUT SEEING CHAOS: EXTRAPOLATION OF GLOBAL DYNAMICS IN AUTOREGRESSIVE TRANSFORMERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Autoregressive models are trained to predict a system's behavior one step at a time, and recursive generation allows the learned dynamics to unfold over long horizons. To what extent can such dynamics learned from local observations recover broader organization of an underlying system that was only partially observed during training? We study small autoregressive transformers trained from scratch on trajectories sampled from restricted parameter regimes of several nonlinear dynamical systems, including logistic and sine maps, the Lorenz system, and the generalized Hopf system, with control parameters and state trajectories represented as sequences of continuous tokens. Under closed-loop evaluation at parameter ranges far outside the training distribution, the models recover correct self-similar period-doubling cascades, chaotic dynamics, and attractor structures with remarkable visual and numerical fidelity. For the logistic map, a transformer reproduces successive period doublings up to period 128, yielding a finite-order scaling ratio of 4.6687, matching the Feigenbaum constant to within 5×10^{-4}. We further investigate how these structures emerge over the course of training, and reveal with causal interventions how control-parameter information is processed through attention to shape the learned dynamics. These results suggest that a surprisingly narrow window into a system's local behavior may suffice for autoregressive transformers to generalize to its unseen global dynamical organization.

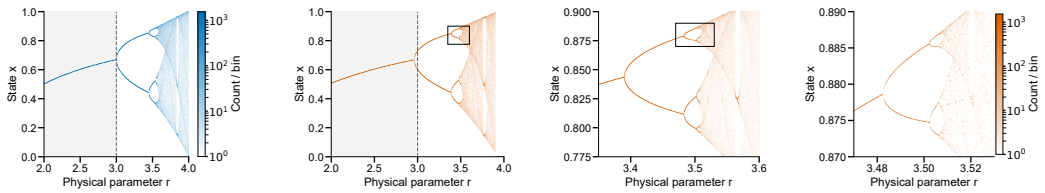
1 INTRODUCTION

Autoregressive models of dynamical systems learn locally from partial observations by construction. They are trained to predict the next observation from those available so far, often on trajectories that cover only a restricted range of system conditions. From this limited evidence, the model is expected to internalize an account of how the underlying system actually evolves. At inference, each prediction is fed back as the next input, turning into an autonomous closed-loop dynamical system whose behavior can unfold horizons and parameter regimes that were never represented during training. There is no guarantee that these unfolded dynamics remain faithful to the underlying system. Recursive generation shifts the input distribution from observed states to model-generated ones (Bengio et al., 2015; Ross et al., 2011), while extrapolation in the parameter distributions requires extending the learned transition beyond the conditions constraining it during training.

Can a model trained only on local observations give rise to parameter-dependent closed-loop operators that are sufficiently accurate and robust to preserve aspects of ground-truth global organization of the underlying dynamics when extrapolated beyond the regime it has seen? Prior work confirms that learned dynamical models can reproduce unseen behavior in some parameterized systems (Pathak et al., 2018; Vlachas et al., 2018; Köglmayr & Räth, 2024; Kim et al., 2021; Huh et al., 2025; van Tegelen et al., 2025), while recent results with transformers occupy an ambiguous position (Srivastava et al., 2023; Casert et al., 2024). Here, we ask whether a small autoregressive transformer can learn parameter-dependent local dynamics that, when recursively applied, give rise to the correct unseen qualitative structure.

Bifurcations provide a particularly informative test of this question. Unlike ordinary out-of-distribution (OOD) prediction, extrapolating across a bifurcation requires the learned dynamics to cross a qualitative change in long-term behavior, where the trivial, stable fixed-point trajectories give

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.



(a) Ground truth (b) Closed-loop dynamics of an autoregressive transformer

Figure 1: **Autoregressive Transformer trained only on pre-bifurcation trajectories of the logistic map recovers self-similar period-doubling structure beyond the domain it was trained on.** Closed-loop results are generated by transformer of seed 42 trained after 10,000 steps with one state token, where the boxes mark zoomed-in areas in the successive panels. The shaded areas mark the regime $[2, 3]$ used for training. The Transformer cascade is displaced relative to the reference, as quantified in Section 4.1; the sine counterpart is shown in Appendix G.

way to periodic orbits and eventually chaos. In the quadratic universality class, successive stable cycles of periods $2, 4, 8, \ldots$ emerge through nested sequences of bifurcations, whose parameter intervals asymptotically approach the Feigenbaum scaling ratio (Feigenbaum, 1978). Such properties give us increasingly stringent notions of extrapolation, whether a model trained only on stable-point trajectories simply mimics the superficial pattern, whether it captures a coherent sequence of period doublings, whether it places those transitions correctly, and ultimately whether their relative scaling follows the same universality structure.

We experiment with small decoder-only transformers trained autoregressively from scratch on trajectories sampled only from restricted parameter regimes of several nonlinear systems, including the logistic and sine maps with period-doubling structures, and the Lorenz systems and generalized Hopf systems, which provides multidimensional chaotic attractors and multistable behaviors. Control parameter scalars and the observed states are represented directly as independent continuous tokens, and training uses only local next-state prediction, without any other labels, stability annotations, governing equations, or long-time dynamical summaries. At test time, we recursively apply the learned transition at parameter values far outside the training range and examine the resulting closed-loop dynamics.

Across these systems, autoregressive transformers extrapolate qualitatively correct global dynamics far beyond the parameter regimes observed during training. For one-dimensional maps, models trained before the onset of period doubling generate almost perfect cascades and chaotic behavior; in the logistic family, seven successive flips up to period 128 are numerically verified, with scaling ratio closely matching the reference Feigenbaum constant. The cascade persists under narrow and substantially shifted training windows and disappears when the correspondence between parameters and trajectories is disrupted, linking the extrapolated behavior to learned parameter-conditioned dynamics. Beyond period doubling, models on the Lorenz and generalized Hopf system also successfully recover unseen attractor behavior. Causal interventions on attention further reveal how control-parameter information shapes the learned dynamics. Together, these results suggest that local next-state supervision within trivial dynamics of a system can give rise to autonomous dynamics that preserve nontrivial global organization well beyond the training distribution.

2 Background and Preliminaries

2.1 From system identification to learned scientific models

Learning scientific dynamics from data targets different objects. Classical neural system identification seeks learning nonlinear transition laws directly from observations (Narendra & Parthasarathy, 1990), while automated scientific-discovery methods seek explicit symbolic descriptions of the underlying system, including symbolic reconstruction of nonlinear dynamics (Bongard & Lipson, 2007), free-form natural-law discovery (Schmidt & Lipson, 2009), sparse identification of governing equations (Brunton et al., 2016), equation-learning networks (Sahoo et al., 2018), and physics-informed symbolic regression (Udrescu & Tegmark, 2020). More recent approaches use learned representations as intermediate scientific models, for example by extracting symbolic relations from neural networks (Cranmer et al., 2020) or by designing interpretable function approximators (Liu et al., 2025). A complementary family of methods learns predictive models without requiring an explicit symbolic law. Koopman-inspired models seek representations in which nonlinear evolu-

tion becomes approximately linear (Lusch et al., 2018); neural differential equations parameterize continuous-time dynamics directly (Chen et al., 2018); and recurrent or reservoir models can reproduce autonomous attractor-level behavior from observed trajectories (Lu et al., 2018; Pathak et al., 2018). Related ideas appear in scientific surrogate modeling, where reduced-order models and neural operators learn parameterized families of physical solutions (Hesthaven & Ubbiali, 2018; Li et al., 2021), and in *world models*, where learned predictive dynamics support internal simulation and planning (Ha & Schmidhuber, 2018). These approaches differ in what they aim to recover: an explicit governing law, a useful representation, a solution operator, or a predictive dynamical surrogate. Our setting concerns rather than whether the exact governing equation is identified, but whether a generic learned predictor can preserve structural consequences of that equation outside the conditions from which it was learned.

2.2 Parameterized dynamical systems and bifurcations

We consider a family of dynamical systems whose evolution depends on control parameter $r \in \mathcal{R}$. For a discrete-time system, we have

$$x_{t+1} = f_r(x_t), \tag{1}$$

where $x_t \in \mathcal{X}$ is the state. For a continuous-time system

$$\dot{x} = v_r(x) \tag{2}$$

we denote by $\Phi_r^{\Delta t}$ its flow map over an observation interval Δt, such that

$$x_{t+\Delta t} = \Phi_r^{\Delta t}(x_t). \tag{3}$$

This allows discrete maps and discretely observed flows be viewed through a common notion of repeated state transition, and we denote F_r for either form when the distinction is unimportant.

Given a learned predictor g_θ, recursive application likewise induces a parameterized autonomous dynamical system, which we denote by $F_{\theta,r}$ when the learned state is Markovian. Our analysis concerns selected properties of the dynamics $F_{\theta,r}$ induces under iteration. We examine *structural extrapolation* by asking whether the learned system $F_{\theta,r}$ preserves aspects of the organization of F_r outside the parameter support observed during training, including properties of the qualitative or quantitative organization of the induced dynamics, such as attractor type, periodicity, bifurcation structure, scaling relations, chaotic instability, etc.

Bifurcations are qualitative changes in the dynamics of a parameterized system as a control parameter varies (Kuznetsov, 1998). The relevant stability object depends on the system and invariant set under consideration. For a scalar discrete map F_r, let x belong to an orbit of minimal period-p $x, F_r(x), \ldots, F_r^{p-1}(x)$, so that the orbit and its cycle multiplier $\Lambda_p(r, x)$ satisfy

$$F_r^p(x) = x, \qquad \Lambda_p(r, x) = \left(F_r^p\right)'(x) = \prod_{j=0}^{p-1} F_r'\left(F_r^j(x)\right). \tag{4}$$

The orbit is locally asymptotically stable when $|\Lambda_p| < 1$. Under the standard transversality and nondegeneracy conditions, a flip, or period-doubling, bifurcation occurs when the multiplier passes through -1, creating a nearby period-$2p$ branch. We combine orbit closure, the multiplier condition, and two-sided stability continuation jointly to quantitatively verify such transitions numerically.

Successive flip bifurcations can organize into a period-doubling cascade. Let $r_1, r_2, \cdots$ denotes successive bifurcation parameters along a principal cascade, with attracting periods $1, 2, 4, \ldots$. For smooth unimodal maps in the quadratic universality class, the finite-order ratios of successive parameter intervals $\{\delta_n\}$ converge asymptotically to the Feigenbaum constant

$$\delta = \lim_{n\to\infty} \delta_n \approx 4.6692016091, \qquad \delta_n = \frac{r_n - r_{n-1}}{r_{n+1} - r_n}, \tag{5}$$

under the usual assumptions defining this universality class (Feigenbaum, 1978). This universality is a statement about the asymptotic organization of the cascade, with different maps in the same universality class having different bifurcation locations r_n while sharing the same limiting scaling ratio. We distinguish this throughout the paper between bifurcation placement, measured by the physical values of r_n, from cascade scaling, measured by the finite-order ratios δ_n.

The continuous-time experiments involve other forms of bifurcation mechanisms. For example, a generic Hopf bifurcation occurs when a complex-conjugate pair of Jacobian eigenvalues crosses the

imaginary axis under the corresponding transversality and nondegeneracy conditions (Kuznetsov, 1998). We elaborate system-specific stability and attractor diagnostics for the flow experiments in the repsective sections.

2.3 Learning dynamics across parameter-conditioned regimes

Data-driven models can reproduce autonomous behavior that extends beyond short-horizon trajectory prediction. Recurrent and reservoir models can reproduce long-run or attractor-level behavior of chaotic systems (Vlachas et al., 2018; Pathak et al., 2018; Lu et al., 2018; Gauthier et al., 2021), while parameterized scientific surrogates and neural operators generalize solution maps across physical conditions (Hesthaven & Ubbiali, 2018; Li et al., 2021). These settings motivate studying a learned predictor itself as a dynamical system, but generalization of a solution family does not necessarily imply recovery of qualitative regime transitions.

Closer to our setting are models that explicitly condition evolution on a control parameter. Recurrent networks can infer control-dependent transformations and bifurcation structure from local examples and extrapolate them beyond the observed control range (Kim et al., 2021). Parameter-aware reservoir methods have been used to extrapolate tipping transitions and post-transition dynamics (Köglmayr & Räth, 2024), and dynamical analysis of such a reservoir shows that the learned autonomous model can itself undergo a corresponding bifurcation near that of the target system (Sisodia & Jalan, 2024). Parameterized Neural ODEs similarly extrapolate bifurcation structure through learned parameter-dependent vector fields (van Tegelen et al., 2025).

Specifically, transformers (Vaswani et al., 2017) have also been studied under several related formulations. Autonomously generated trajectories can exhibit emergent behavior at physical conditions absent from training (Casert et al., 2024); other work predicts bifurcation diagrams directly from trajectory observations, making bifurcation structure itself the supervised object (Zhornyak et al., 2024); and recent experiments report failures to extrapolate unseen collapse transitions despite successful parameter-aware reservoir baselines (Zhai et al., 2026). These results suggest that regime extrapolation is not guaranteed by architecture or low local prediction error, but is an empirical property of the autonomous dynamics induced by the learned transition. Beyond behavioral evaluation, causal-intervention methods provide tools for testing whether internal representations or computational pathways mediate model outputs (Geiger et al., 2021; Meng et al., 2022). We adopt this perspective to examine how control-parameter information influences the learned transition to help better understand how local predictive learning over a restricted parameter region induces the appropriate dynamical organization outside that region.

3 Methodology and Experiments

3.1 Systems, Data, and Training Objective

We first study two one-parameter families of scalar maps on $x \in [0, 1]$,

$$F_r(x) = rx(1 - x) \qquad \text{and} \qquad F_r(x) = r \sin(\pi x), \tag{6}$$

corresponding to the logistic and sine families. For the logistic map, training parameters are sampled uniformly from $\mathcal{R}_{\text{train}} = [2, 3]$ and evaluation covers $2, 4$. For the sine map, training uses $0.1, 0.71$ and evaluation covers $0.1, 1$. The first period-doubling bifurcations of the reference maps occur at $r = 3$ and $r \approx 0.719961683$, respectively, so neither training distribution contains an interval beyond the first period doubling. For each trajectory, the control parameter and initial state are sampled independently and uniformly.

Given a trajectory $x_0, \ldots, x_T$ generated at parameter r, the models are trained with teacher-forced next-state autoregression,

$$\mathcal{L}(\theta) = \mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1} \left\| g_\theta(r, x_{0:t}) - x_{t+1} \right\|_2^2\right], \tag{7}$$

where every prediction is conditioned on the reference prefix $x_{0:t}$ during training, and model-generated states enter only during autonomous closed-loop evaluation. Sequence lengths are drawn from sampling per batch, and no burn-in samples is removed from the training dataset. Appendix A provides the complete sampling, optimization, and checkpoint protocols.

3.2 Parameterized Transformer and Induced Dynamics

The control parameter and states are represented as separate continuous tokens. In the scalar setting,

$$h_r^{(0)} = W_r r + b_r + v_r, \qquad h_t^{(0)} = W_x x_t + b_x + v_x, \tag{8}$$

where v_r and v_x are learned type embeddings distinguishing parameter from state tokens. These are linear embeddings of continuous numerical values rather than discrete one-hot vocabulary embeddings, and the parameter token(s) is appended preceding all state tokens in the causal sequence.

The model is a decoder-only Transformer with residual-connected layers. Each layer applies causal multi-head self-attention followed by a gated feed-forward block, with $\mathrm{SiLU}(z) = z\sigma(z)$,

$$u^{(\ell)} = h^{(\ell)} + \mathrm{Attn}_\ell(h^{(\ell)}), \qquad h^{(\ell+1)} = u^{(\ell)} + W_o[\mathrm{SiLU}(W_g u^{(\ell)}) \odot W_v u^{(\ell)}], \tag{9}$$

where $\mathrm{SiLU}(z) = z\sigma(z)$. A linear output head predicts the next state at every state-token position. The configuration for scalar map experiments has hidden width 16, two attention heads, feed-forward width 64, two untied residual layers, and 8,304 trainable parameters. It uses no positional embeddings, layer normalization, input normalization, nor dropout. Details in Appendix A.

Because feed-forward computation is token-wise, a separate parameter token exposes an explicit route within the attention mechanisms, through which parameter information can enter a state token before affecting its subsequent nonlinear computation. We exploit this route for intervention in Section 3.4. This does not imply that attention weights themselves implement the analytic parameter dependence of the physical maps, nor that successful trajectories identify a particular internal arithmetic algorithm; Appendix P addresses these representational possibilities separately.

At evaluation time the trained parameters θ are frozen and predictions are returned to the model as inputs. For a context containing at most C state observations,

$$\hat{x}_{t+1} = \Pi_{[0,1]} g_\theta\big(r, \hat{x}_{\max(0,t-C+1):t}\big), \tag{10}$$

where $\Pi_{[0,1]}$ clips the scalar predictions to the physically allowed state interval and the available prefix is used before the context reaches length C. For $C = 1$, recursive prediction defines a smooth, parameterized neural transition and the corresponding rollout map

$$F_{\theta,r}(x) = \Pi_{[0,1]} \widetilde{F}_{\theta,r}(x), \qquad \widetilde{F}_{\theta,r}(x) = g_\theta(r, x). \tag{11}$$

The local bifurcation calculations below concern periodic orbits in the interior of the state domain, where the clamp is inactive and the derivatives of $F_{\theta,r}$ coincide with those of the smooth neural transition $\widetilde{F}_{\theta,r}$. This separates the differentiable map used for local stability analysis from the range constraint used during autonomous rollout.

For $C > 1$ the generated process is not a one-dimensional map even though each observation is scalar. Once the context is full, define

$$X_t(\hat{x}_{t-C+1}, \ldots, \hat{x}_t) \in [0,1]^C. \tag{12}$$

The induced autonomous update is then the delay-state map

$$\mathcal{F}_{\theta,r}(X_t) = \big(\hat{x}_{t-C+2}, \ldots, \hat{x}_t, \Pi_{[0,1]} g_\theta(r, X_t)\big). \tag{13}$$

We consequently restrict scalar cycle-multiplier and root-continuation calculations to $C = 1$, while trajectory- and distribution-level comparisons are applied to all evaluated contexts.

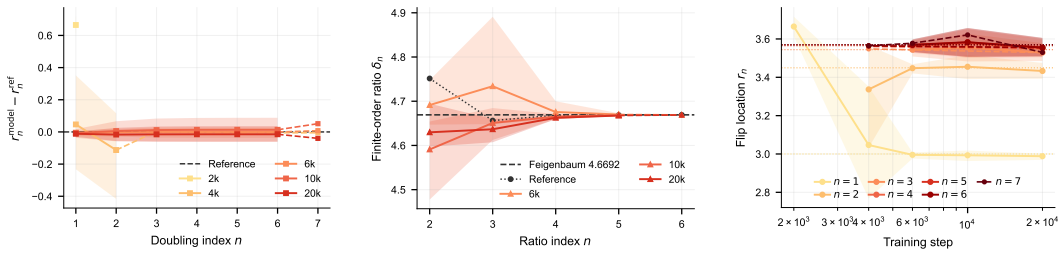
3.3 Extension to Continuous-Time Flows

For continuously evolving systems observed at a fixed interval Δt, we retain the same separation between parameter and state tokens but predict a finite-time state increment instead of the next state directly. Denoting the learned increment as Δ_θ, the resulting discrete transition is

$$\widehat{\Phi}^{\Delta t}_{\theta,r}(x) = x + \Delta_\theta(r, x), \tag{14}$$

which approximates the physical flow map $\Phi^{\Delta t}_r$ introduced in Section 2. The model is therefore trained on discretely observed flow transitions; the continuous vector field v_r itself is neither supplied nor explicitly reconstructed. All state, parameter, and increment normalization statistics are estimated from the corresponding training data only, and the flow rollouts are not clipped.

We apply this construction to Lorenz and generalized Hopf systems. The Lorenz experiments include both a one-parameter setting and a joint setting in which (ρ, σ) are embedded in a single parameter token with $\beta = 8/3$ fixed; the state token contains the Cartesian state (x, y, z). In the



(a) Flip parameter displacement (b) Finite-order scaling ratios (c) Flip parameters during training

Figure 2: **Emergence of period-doubling cascades beyond the observed regime can be numerically verified via the precise reconstruction of period-doubling parameters.** Results for the Logistic map, averaged across seeds with shaded min–max ranges; solid segments are evidence supported by all seeds, dashed segments only partially supported by those that had reached that level. r_n is verified as detailed in Section B.1.

joint experiment, training parameters are drawn only from trajectories that numerically satisfy a numerical fixed-destination criterion, with their finite-time transients retained. For the generalized Hopf system, the parameter token contains (μ_1, μ_2) and the state token contains Cartesian coordinates (x, y). Training is restricted to a region in which every sampled nonzero trajectory contracts radially, so no stable periodic attractor appears in the training data. The detailed parameters, sampling, normalization, and optimization settings are specified in Appendices H, J, and K.

3.4 Intervention on Parameter Conditioning

To test whether parameter information transmitted through attention causally affects the state prediction, we patch activations while holding the state history fixed. Let H denote a fixed state history and let r and r' be recipient and donor parameters, with $y_{\mathrm{rec}} = g_\theta(r, H)$ and $y_{\mathrm{don}} = g_\theta(r', H)$. Let $a_\ell(r, H)$ be the complete attention output at the final state position in layer ℓ, immediately before its residual addition. Replacing it by $a_\ell(r', H)$ in the recipient computation produces a patched prediction y_ℓ^{patch}, and the fraction of the donor effect transferred by the intervention is

$$R_\ell = 1 - \frac{\mathbb{E}[(y_\ell^{\mathrm{patch}} - y_{\mathrm{don}})^2]}{\mathbb{E}[(y_{\mathrm{rec}} - y_{\mathrm{don}})^2]}. \tag{15}$$

Thus $R_\ell = 1$ corresponds to recovery of the donor model output and $R_\ell = 0$ to no reduction in donor discrepancy relative to the unmodified recipient. We adopt sham replacement that reuses the recipient activation, and a random control that uses a per-example random direction of equal norm as the donor–recipient activation difference. We also complement this single-step intervention with closed-loop ones. In a selected layer, all attention logits from state queries to the parameter key are set to $-\infty$ at every generated step, and softmax is recomputed over the remaining admissible keys. This removes the selected parameter-to-state attention edge. Appendix L reports the resulting trajectories together with the native model, sham and equal-norm random controls.

3.5 Measuring Structural Extrapolation

We evaluate properties of the autonomous system induced by the frozen predictor. The scalar measurements separately examine long-run state distributions, periodic organization, bifurcation placement, and cascade scaling, along with longitudinal measurements comparing these quantities with local transition accuracy during training. The multidimensional flow systems use diagnostics appropriate to their higher-dimensional attractors and stability structure. More details in Appendix B.

4 Results and Analyses

Here we organize results concerning whether and how global structure emerges beyond the observed regime during training, how much observation it requires, what kind of structure is recovered, how parameter information supports it, and whether the same phenomenon extends beyond scalar maps.

4.1 Emergence of Global Structure Beyond the Observed Regime

The learned logistic maps develop a high-order period-doubling cascade entirely outside the training interval. At the final checkpoint, seven consecutive flips can be numerically verified, reaching period-128. The deepest finite-order ratio shared across all seeds is 4.668686, compared with

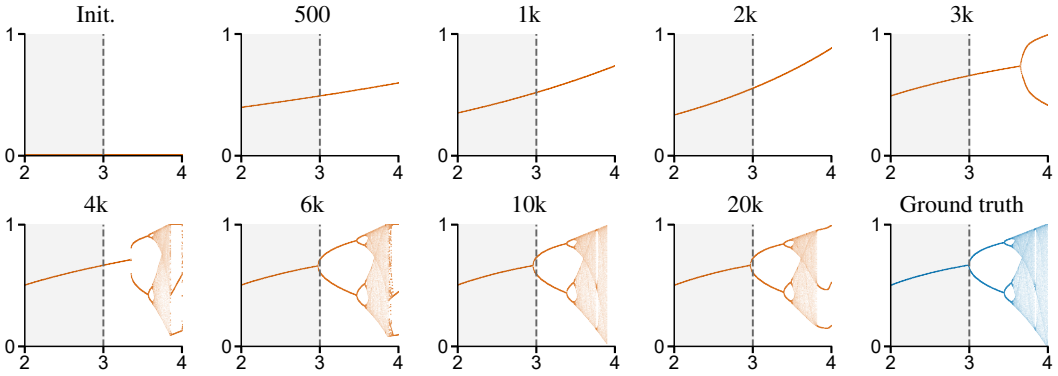


Figure 3: **Emergence of period-doubling cascades beyond the observed regime can be visually examined via the similar patterns reconstructed.** Results for the Logistic map, with seed 42, $C = 1$, on the same 513-point grid used in Figure 1. Gray shading marks the training interval.

4.669132 for the reference under the same definition. The same pipeline verifies that the sine reference ladder reaches a corresponding finite-order ratio of 4.668805, with the learned sine model exhibiting the same qualitative cascade structure. Therefore, the learned closed-loop maps recover not only successive qualitative regime changes, but also the self-similar scaling structure organizing the cascade. Additional results are provided in Appendix C, G, and O.

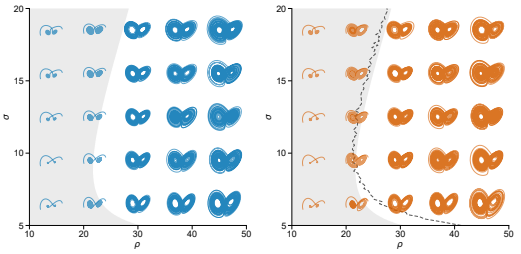
We can also observe from how the patterns emerge across training steps. In the logistic model, no doubling can be found by 2,000 updates, until the first one appears at 3,000, three by 4,000, etc. The sine model develops the same qualitative organization even earlier, with two doublings by 1,000 updates and three by 2,000. In Appendix G we provide additional results showing that how local prediction and closed-loop structures evolve differently, with between 6,000 and 10,000 logistic updates, OOD one-step MSE decreases from 1.10×10^{-3} to 2.41×10^{-4}, while closed-loop OOD W_1 increases from 0.0296 to 0.0867. This further confirms that structural organization cannot be reduced to monotonic improvement in one-step prediction accuracy. Appendix C and D report further details about the distributional comparisons, the per-seed displacement ranges and all per-seed values, and verified flip parameters.

We experimented varying the parameter interval available during training while keeping the remaining protocol fixed. Training on $[2, 3]$, $[2.5, 3]$, $[2.9, 3]$, or the substantially lower interval $[2, 2.5]$ still produces the same qualitative cascade organization. At least six consecutive flips can be numerically verified under each training window, and the coarse detector identifies a cascade in 11 of the 12 window–seed combinations. This confirms that emergence of the cascade does not require training observations to be even immediately adjacent to the transition. Details in Appendix E.

The closed-loop predictor can reproduce global structural consequences of the underlying dynamics without explicitly recovering the analytic governing equation. Here we use the recovered Feigenbaum scaling to study a particular structural property of the learned map. The logistic Transformer has a fitted critical exponent $z = 2.00$, and the sine Transformer $z \approx 2.10$, consistent with a quadratic critical point, which can also be visually examined in cobweb plot results provided in Appendix N and detailed in F. Their autonomous cascades therefore exhibit the organization expected of the quadratic universality class, confirming a recovery of ground truth dynamical organizations.

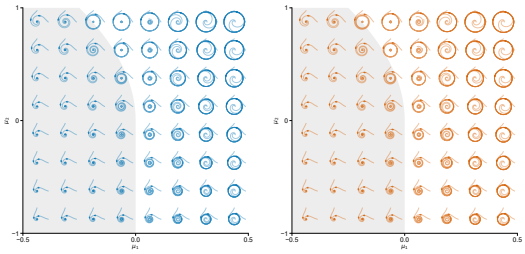
4.2 Parameter Conditioning Causally Shapes the Learned Dynamics

Counterfactual activation patching shows that parameter dependence is mediated through learned attention routes. Replacing the final-query attention output while holding the state history fixed identifies where parameter information enters the state computation. A second-layer patch transfers almost the full donor effect ($R_1 > 0.99999$) whereas a first-layer patch is negligible ($R_0 = 0.000151$). The equal-norm random control increases donor error and sham replacement leaves the prediction unchanged. The dominant mediating layer varies across seeds, confirming that the route is indeed learned during training rather than hard-coded by the architecture. Closed-loop blocking connects these local interventions to global dynamics. For logistic seed 42, blocking the relevant parameter-to-state route raises OOD W_1 from 0.03497 to 0.60605 and removes the period-doubling structure. The parameter–trajectory correspondence is also critically required during learning. When parame-



(a) Ground truth (b) Transformer

Figure 5: **Extrapolation of regime structure in the Lorenz system.** Training is confined to the gray fixed-destination region, where trajectories converge to fixed states. Beyond it, the model recovers sustained double-wing dynamics and the transition between the two regimes. The dashed curve marks the reconstructed boundary.



(a) Ground truth (b) Transformer

Figure 6: **Extrapolation of multistable dynamics in the generalized Hopf system.** Training is confined to the gray radial-contraction region, where no stable cycle is observed. Beyond it, the model recovers stable-point, stable-cycle, and coexisting-attractor behavior from matched initial conditions.

ter labels are permuted across training examples, all runs lose coarse-grid flips and periodic agreement, with OOD error rising sharply. The extrapolated organization therefore depends on learning how the control parameter changes the transition itself. Detailed results can be found in Appendix L.

4.3 Structural Extrapolation in Continuous Flows

The same parameter-conditioned formulation recovers nontrivial regime organization in a multidimensional chaotic flow. Across a joint (ρ, σ) evaluation region, the model trained only on short observations of fixed-destination trajectories agrees with the reference fixed/nonfixed classification in $593/625$ cells and reproduces repeated wing switching in $362/375$ reference cells exhibiting such motion. Figure 5 show the corresponding change of regime at matched parameter pairs: inside the gray region that the model has seen during training, trajectories spiral onto a fixed point, and outside they extrapolate into correct sustained double-wing motion whose amplitude grows with ρ. The model also gives an accurate boundary, tracking the reference to a median absolute displacement of 0.95 in ρ with a maximum of 10.7 only at the steep low-σ end. Appendix K.1 reports both evaluated models, the complete boundary displacements, the distributional comparisons and the rendering protocols. The recovered nonfixed dynamics are genuinely unstable rather than merely visually double-winged. At $(\rho, \sigma) = (28, 10)$, the learned system has largest Lyapunov exponent 1.0404, compared with 0.9046 for the reference under the same estimator, and the sign agrees at all ten tested parameter pairs. Appendix I gives further details.

The two-parameter generalized Hopf normal form (Kuznetsov, 1998)

$$\dot{r} = r(\mu_1 + \mu_2 r^2 - r^4), \qquad \dot{\theta} = 1, \tag{16}$$

tests a qualitatively different structure: coexistence of attracting states. The model observes only Cartesian states (x, y) together with a parameter token (μ_1, μ_2); neither the radial coordinate nor the governing normal form is provided. Training is restricted to a region in which every sampled nonzero trajectory contracts radially, so the model observes only trajectories approaching the origin and never a stable periodic attractor. Figure 6 displays the resulting transient geometry on both sides of this curved training boundary. The model correctly reproduces the reference point-versus-cycle organization across all nine tested parameter pairs. All 36 trajectories remain finite through $t = 300$ and match the reference long-time outcome, with 14 approaching fixed points and 22 approaching cycles. At coexistence parameters, different initial radii converge to different attractors; for example, at $(\mu_1, \mu_2) = (-0.1, 1)$ the predicted outer-cycle radius is 0.948, compared with 0.942 for the reference. Because each parameter pair is evaluated from multiple initial radii and phases, the coexistence is generated by the learned dynamics rather than inferred from a single trajectory. Appendix J reports additional long-horizon basin tests, parameter slices, and learned stability boundaries.

Taken together, the scalar, Lorenz, and generalized Hopf results point to the same phenomenon across distinct dynamical mechanisms, where local parameter-conditioned prediction can give rise to autonomous systems that recover nontrivial unseen chaotic, periodic, and multistable long-time organization beyond the regimes represented during training.

5 DISCUSSION

Our experiments reveal a recurring pattern across several nonlinear systems: local next-state learning can organize a learned predictor into an autonomous dynamical system with global structure that was never represented in its training regime. In the scalar maps, the learned systems develop an extended hierarchy of period doublings whose finite-order scaling approaches the Feigenbaum constant. In Lorenz and generalized Hopf systems, the same formulation gives rise to unseen chaotic and multistable long-time behavior. The verification recipe is system-agnostic, with a blind coarse scan proposes candidates, paired burn-in budgets with phase agreement confirm them, and a closure-plus-multiplier solve certifies each flip, yielding flip parameters and ratios without reference to known bifurcation values (Appendices A and C). The same three ingredients, a parameter token, short local observations and closed-loop evaluation, carry over from scalar maps to continuous flows (Appendices H, K and J).

One of the most striking observations is that structural organization appears early in training and follows a trajectory distinct from one-step prediction accuracy. In the logistic system, a recognizable cascade is already present while its transition locations are still evolving, and later improvements in one-step OOD prediction do not translate monotonically into improved closed-loop fidelity. This suggests that recursive application exposes an organizing structure of the learned transition that is not captured by local prediction error alone. The same principle appears beyond one-dimensional period doubling. In the joint Lorenz system, a model trained on fixed-destination trajectories recovers a structured transition to sustained double-wing motion across two jointly varying parameters and produces positive Lyapunov exponents in chaotic regimes. In the generalized Hopf system, training trajectories only contract radially, yet the learned dynamics later generate stable cycles and reproduce coexistence between point and cycle attractors from different initial conditions. The common thread is the emergence of global regime organization from restricted local predictive experience.

Transformer represents parameters and states separately and combines them through attention. Counterfactual patches and closed-loop blocking establish that these learned parameter routes shape the generated dynamics across scalar systems and seeds (Section 4.2), while the verified cascade characterizes the resulting map. The token architecture is what makes that control route accessible to intervention, and the representation baselines of Appendix D place the result against alternatives that do not expose it. Together, these results connect token-level parameter processing to the global dynamical organization produced under recursive feedback.

These findings suggest a complementary perspective on data-driven scientific modeling. A learned predictor can encode scientifically meaningful organization not only in its pointwise predictions or internal representations, but also in the dynamical system obtained by recursively applying it. Bifurcation structure, universality scaling, chaotic instability, and attractor selection therefore become objects for probing what a model has learned about an underlying physical family. This points toward learned autonomous dynamics as a potentially useful substrate for scientific discovery, especially when the goal is to facilitate understanding global consequences of local observations.

6 CONCLUSION

Autoregressive Transformers trained only on restricted local observations can generate unseen global dynamical organization. In scalar maps, the learned closed-loop systems develop numerically verified high-order period-doubling cascades with Feigenbaum scaling; in Lorenz and generalized Hopf systems, the same parameter-conditioned formulation produces unseen chaotic and multistable long-time behavior. Causal interventions further connect these global dynamics to learned pathways through which control parameters influence state computation. These results suggest that a surprisingly narrow window into a system's local behavior may suffice for autoregressive transformers to generalize to its unseen global dynamics.

AI USE STATEMENT

Generative AI tools were used to assist with language editing and polishing of the manuscript, and to support retrieval and discovery of potentially relevant literature. All retrieved references were independently checked by the authors for relevance and correctness. Generative AI was not involved in the design of methodology and experiments, the analysis and interpretation of results, generating

synthetic datasets, or proving mathematical claims. All scientific content, analyses, results, and final wording were reviewed and approved by the authors.

Reproducibility Statement

The appendix documents all detailed training, evaluations, and interventions results.

References

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. In *Advances in Neural Information Processing Systems*, volume 28, 2015. URL https://proceedings.neurips.cc/paper_files/paper/2015/hash/e995f98d56967d946471af29d7bf99f1-Abstract.html.

Josh Bongard and Hod Lipson. Automated reverse engineering of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 104(24):9943–9948, 2007. doi: 10.1073/pnas.0609476104. URL https://www.pnas.org/doi/abs/10.1073/pnas.0609476104.

Steven L. Brunton, Joshua L. Proctor, and J. Nathan Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 113(15):3932–3937, 2016. doi: 10.1073/pnas.1517384113. URL https://www.pnas.org/doi/abs/10.1073/pnas.1517384113.

Corneel Casert, Isaac Tamblyn, and Stephen Whitelam. Learning stochastic dynamics and predicting emergent behavior using transformers. *Nature Communications*, 15:1875, 2024. doi: 10.1038/s41467-024-45629-w. URL https://doi.org/10.1038/s41467-024-45629-w.

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.

Miles Cranmer, Alvaro Sanchez Gonzalez, Peter Battaglia, Rui Xu, Kyle Cranmer, David Spergel, and Shirley Ho. Discovering symbolic models from deep learning with inductive biases. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 17429–17442. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/c9f2f917078bd2db12f23c3b413d9cba-Paper.pdf.

Eusebius J. Doedel, Bernd Krauskopf, and Hinke M. Osinga. Global invariant manifolds in the transition to preturbulence in the lorenz system. *Indagationes Mathematicae*, 22(3):222–240, 2011. ISSN 0019-3577. doi: https://doi.org/10.1016/j.indag.2011.10.007. URL https://www.sciencedirect.com/science/article/pii/S001935771100067X. Devoted to: Floris Takens (1940–2010).

Mitchell J. Feigenbaum. Quantitative universality for a class of nonlinear transformations. *Journal of Statistical Physics*, 19(1):25–52, 1978. doi: 10.1007/BF01020332. URL https://doi.org/10.1007/BF01020332.

Daniel J. Gauthier, Erik Bollt, Aaron Griffith, and Wendson A. S. Barbosa. Next generation reservoir computing. *Nature Communications*, 12(1):5564, Sep 2021. ISSN 2041-1723. doi: 10.1038/s41467-021-25801-2. URL https://doi.org/10.1038/s41467-021-25801-2.

Atticus Geiger, Hanson Lu, Thomas Icard, and Christopher Potts. Causal abstractions of neural networks. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 9574–9586. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/4f5c422f4d49a5a807eda27434231040-Paper.pdf.

David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

J.S. Hesthaven and S. Ubbiali. Non-intrusive reduced order modeling of nonlinear problems using neural networks. *J. Comput. Phys.*, 363(C):55–78, June 2018. ISSN 0021-9991. doi: 10.1016/j.jcp.2018.02.037. URL https://doi.org/10.1016/j.jcp.2018.02.037.

In Huh, Changwook Jeong, and Muhammad Alam. Context-informed neural ODEs unexpectedly identify broken symmetries: Insights from the poincaré–hopf theorem. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 26311–26337. PMLR, 2025. URL https://proceedings.mlr.press/v267/huh25a.html.

Jason Z. Kim, Zhixin Lu, Erfan Nozari, George J. Pappas, and Danielle S. Bassett. Teaching recurrent neural networks to infer global temporal structure from local examples. *Nature Machine Intelligence*, 3(4):316–323, Apr 2021. ISSN 2522-5839. doi: 10.1038/s42256-021-00321-2. URL https://doi.org/10.1038/s42256-021-00321-2.

Daniel Köglmayr and Christoph Räth. Extrapolating tipping points and simulating non-stationary dynamics of complex systems using efficient machine learning. *Scientific Reports*, 14(1):507, Jan 2024. ISSN 2045-2322. doi: 10.1038/s41598-023-50726-9. URL https://doi.org/10.1038/s41598-023-50726-9.

Yuri A. Kuznetsov. *Elements of Applied Bifurcation Theory*. Springer New York, New York, NY, 1998. ISBN 978-0-387-22710-8. doi: 10.1007/978-0-387-22710-8_8. URL https://doi.org/10.1007/978-0-387-22710-8_8.

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.

Ziming Liu, Yixuan Wang, Sachin Vaidya, Fabian Ruehle, James Halverson, Marin Soljacic, Thomas Y. Hou, and Max Tegmark. KAN: Kolmogorov–arnold networks. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/afaed89642ea100935e39d39a4da602c-Abstract-Conference.html.

Zhixin Lu, Brian R. Hunt, and Edward Ott. Attractor reconstruction by machine learning. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 28(6):061104, 06 2018. ISSN 1054-1500. doi: 10.1063/1.5039508. URL https://doi.org/10.1063/1.5039508.

Bethany Lusch, J. Nathan Kutz, and Steven L. Brunton. Deep learning for universal linear embeddings of nonlinear dynamics. *Nature Communications*, 9(1):4950, Nov 2018. ISSN 2041-1723. doi: 10.1038/s41467-018-07210-0. URL https://doi.org/10.1038/s41467-018-07210-0.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 17359–17372. Curran Associates, Inc., 2022. doi: 10.52202/068431-1262. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/6f1d43d5a82a37e89b0665b33bf3a182-Paper-Conference.pdf.

K.S. Narendra and K. Parthasarathy. Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks*, 1(1):4–27, 1990. doi: 10.1109/72.80202.

Jaideep Pathak, Brian Hunt, Michelle Girvan, Zhixin Lu, and Edward Ott. Model-free prediction of large spatiotemporally chaotic systems from data: A reservoir computing approach. *Phys. Rev. Lett.*, 120:024102, Jan 2018. doi: 10.1103/PhysRevLett.120.024102. URL https://link.aps.org/doi/10.1103/PhysRevLett.120.024102.

Stephane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pp. 627–635. PMLR, 2011. URL https://proceedings.mlr.press/v15/ross11a.html.

Subham S. Sahoo, Christoph H. Lampert, and Georg Martius. Learning equations for extrapolation and control. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pp. 4442–4450. PMLR, 2018. URL https://proceedings.mlr.press/v80/sahoo18a.html.

Michael Schmidt and Hod Lipson. Distilling free-form natural laws from experimental data. *Science*, 324(5923):81–85, 2009. doi: 10.1126/science.1165893. URL https://www.science.org/doi/abs/10.1126/science.1165893.

Dishant Sisodia and Sarika Jalan. Dynamical analysis of a parameter-aware reservoir computer. *Phys. Rev. E*, 110:034211, Sep 2024. doi: 10.1103/PhysRevE.110.034211. URL https://link.aps.org/doi/10.1103/PhysRevE.110.034211.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R. Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, Agnieszka Kluska, Aitor Lewkowycz, Akshat Agarwal, Alethea Power, Alex Ray, Alex Warstadt, Alexander W. Kocurek, Ali Safaya, Ali Tazarv, Alice Xiang, Alicia Parrish, Allen Nie, Aman Hussain, Amanda Askell, Amanda Dsouza, Ambrose Slone, Ameet Rahane, Anantharaman S. Iyer, Anders Johan Andreassen, Andrea Madotto, Andrea Santilli, Andreas Stuhlmüller, Andrew M. Dai, Andrew La, Andrew Kyle Lampinen, Andy Zou, Angela Jiang, Angelica Chen, Anh Vuong, Animesh Gupta, Anna Gottardi, Antonio Norelli, Anu Venkatesh, Arash Gholamidavoodi, Arfa Tabassum, Arul Menezes, Arun Kirubarajan, Asher Mullokandov, Ashish Sabharwal, Austin Herrick, Avia Efrat, Aykut Erdem, Ayla Karakaş, B. Ryan Roberts, Bao Sheng Loe, Barret Zoph, Bartłomiej Bojanowski, Batuhan Özyurt, Behnam Hedayatnia, Behnam Neyshabur, Benjamin Inden, Benno Stein, Berk Ekmekci, Bill Yuchen Lin, Blake Howald, Bryan Orinion, Cameron Diao, Cameron Dour, Catherine Stinson, Cedrick Argueta, Cesar Ferri, Chandan Singh, Charles Rathkopf, Chenlin Meng, Chitta Baral, Chiyu Wu, Chris Callison-Burch, Christopher Waites, Christian Voigt, Christopher D Manning, Christopher Potts, Cindy Ramirez, Clara E. Rivera, Clemencia Siro, Colin Raffel, Courtney Ashcraft, Cristina Garbacea, Damien Sileo, Dan Garrette, Dan Hendrycks, Dan Kilman, Dan Roth, C. Daniel Freeman, Daniel Khashabi, Daniel Levy, Daniel Moseguí González, Danielle Perszyk, Danny Hernandez, Danqi Chen, Daphne Ippolito, Dar Gilboa, David Dohan, David Drakard, David Jurgens, Debajyoti Datta, Deep Ganguli, Denis Emelin, Denis Kleyko, Deniz Yuret, Derek Chen, Derek Tam, Dieuwke Hupkes, Diganta Misra, Dilyar Buzan, Dimitri Coelho Mollo, Diyi Yang, Dong-Ho Lee, Dylan Schrader, Ekaterina Shutova, Ekin Dogus Cubuk, Elad Segal, Eleanor Hagerman, Elizabeth Barnes, Elizabeth Donoway, Ellie Pavlick, Emanuele Rodolà, Emma Lam, Eric Chu, Eric Tang, Erkut Erdem, Ernie Chang, Ethan A Chi, Ethan Dyer, Ethan Jerzak, Ethan Kim, Eunice Engefu Manyasi, Evgenii Zheltonozhskii, Fanyue Xia, Fatemeh Siar, Fernando Martínez-Plumed, Francesca Happé, Francois Chollet, Frieda Rong, Gaurav Mishra, Genta Indra Winata, Gerard de Melo, Germàn Kruszewski, Giambattista Parascandolo, Giorgio Mariani, Gloria Xinyue Wang, Gonzalo Jaimovitch-Lopez, Gregor Betz, Guy Gur-Ari, Hana Galijasevic, Hannah Kim, Hannah Rashkin, Hannaneh Hajishirzi, Harsh Mehta, Hayden Bogar, Henry Francis Anthony Shevlin, Hinrich Schuetze, Hiromu Yakura, Hongming Zhang, Hugh Mee Wong, Ian Ng, Isaac Noble, Jaap Jumelet, Jack Geissinger, Jackson Kernion, Jacob Hilton, Jaehoon Lee, Jaime Fernández Fisac, James B Simon, James Koppel, James Zheng, James Zou, Jan Kocon, Jana Thompson, Janelle Wingfield, Jared Kaplan, Jarema Radom, Jascha Sohl-Dickstein, Jason Phang, Jason Wei, Jason Yosinski, Jekaterina Novikova, Jelle Bosscher, Jennifer Marsh, Jeremy Kim, Jeroen Taal, Jesse Engel, Jesujoba Alabi, Jiacheng Xu, Jiaming Song, Jillian Tang, Joan Waweru, John Burden, John Miller, John U. Balis, Jonathan Batchelder, Jonathan Berant, Jörg Frohberg, Jos Rozen, Jose Hernandez-Orallo, Joseph Boudeman, Joseph Guerr, Joseph Jones, Joshua B. Tenenbaum, Joshua S. Rule, Joyce Chua, Kamil Kanclerz, Karen Livescu, Karl Krauth, Karthik Gopalakrishnan, Katerina Ignatyeva, Katja Markert, Kaustubh Dhole, Kevin Gimpel, Kevin Omondi, Kory Wallace Mathewson, Kristen Chiafullo, Ksenia Shkaruta, Kumar Shridhar, Kyle McDonell, Kyle Richardson, Laria Reynolds, Leo Gao, Li Zhang, Liam Dugan, Lianhui Qin, Lidia Contreras-Ochando, Louis-Philippe Morency, Luca Moschella, Lucas Lam, Lucy Noble, Ludwig Schmidt, Luheng He, Luis Oliveros-Colón, Luke Metz, Lütfi Kerem Senel, Maarten Bosma, Maarten Sap, Maartje Ter Hoeve, Maheen Farooqi, Manaal Faruqui, Mantas Mazeika, Marco Baturan, Marco Marelli, Marco Maru, Maria Jose Ramirez-Quintana, Marie Tolkiehn, Mario Giulianelli, Martha Lewis, Martin Potthast, Matthew L Leavitt, Matthias Hagen, Mátyás Schubert, Medina Orduna Baitemirova, Melody Arnaud, Melvin McElrath, Michael Andrew Yee, Michael Cohen, Michael Gu, Michael

Ivanitskiy, Michael Starritt, Michael Strube, Michał Swędrowski, Michele Bevilacqua, Michihiro Yasunaga, Mihir Kale, Mike Cain, Mimee Xu, Mirac Suzgun, Mitch Walker, Mo Tiwari, Mohit Bansal, Moin Aminnaseri, Mor Geva, Mozhdeh Gheini, Mukund Varma T, Nanyun Peng, Nathan Andrew Chi, Nayeon Lee, Neta Gur-Ari Krakover, Nicholas Cameron, Nicholas Roberts, Nick Doiron, Nicole Martinez, Nikita Nangia, Niklas Deckers, Niklas Muennighoff, Nitish Shirish Keskar, Niveditha S. Iyer, Noah Constant, Noah Fiedel, Nuan Wen, Oliver Zhang, Omar Agha, Omar Elbaghdadi, Omer Levy, Owain Evans, Pablo Antonio Moreno Casares, Parth Doshi, Pascale Fung, Paul Pu Liang, Paul Vicol, Pegah Alipoormolabashi, Peiyuan Liao, Percy Liang, Peter W Chang, Peter Eckersley, Phu Mon Htut, Pinyu Hwang, Piotr Miłkowski, Piyush Patil, Pouya Pezeshkpour, Priti Oli, Qiaozhu Mei, Qing Lyu, Qinlang Chen, Rabin Banjade, Rachel Etta Rudolph, Raefer Gabriel, Rahel Habacker, Ramon Risco, Raphaël Millière, Rhythm Garg, Richard Barnes, Rif A. Saurous, Riku Arakawa, Robbe Raymaekers, Robert Frank, Rohan Sikand, Roman Novak, Roman Sitelew, Ronan Le Bras, Rosanne Liu, Rowan Jacobs, Rui Zhang, Russ Salakhutdinov, Ryan Andrew Chi, Seungjae Ryan Lee, Ryan Stovall, Ryan Teehan, Rylan Yang, Sahib Singh, Saif M. Mohammad, Sajant Anand, Sam Dillavou, Sam Shleifer, Sam Wiseman, Samuel Gruetter, Samuel R. Bowman, Samuel Stern Schoenholz, Sanghyun Han, Sanjeev Kwatra, Sarah A. Rous, Sarik Ghazarian, Sayan Ghosh, Sean Casey, Sebastian Bischoff, Sebastian Gehrmann, Sebastian Schuster, Sepideh Sadeghi, Shadi Hamdan, Sharon Zhou, Shashank Srivastava, Sherry Shi, Shikhar Singh, Shima Asaadi, Shixiang Shane Gu, Shubh Pachchigar, Shubham Toshniwal, Shyam Upadhyay, Shyamolima Shammie Debnath, Siamak Shakeri, Simon Thormeyer, Simone Melzi, Siva Reddy, Sneha Priscilla Makini, Soo-Hwan Lee, Spencer Torene, Sriharsha Hatwar, Stanislas Dehaene, Stefan Divic, Stefano Ermon, Stella Biderman, Stephanie Lin, Stephen Prasad, Steven Piantadosi, Stuart Shieber, Summer Misherghi, Svetlana Kiritchenko, Swaroop Mishra, Tal Linzen, Tal Schuster, Tao Li, Tao Yu, Tariq Ali, Tatsunori Hashimoto, Te-Lin Wu, Théo Desbordes, Theodore Rothschild, Thomas Phan, Tianle Wang, Tiberius Nkinyili, Timo Schick, Timofei Kornev, Titus Tunduny, Tobias Gerstenberg, Trenton Chang, Trishala Neeraj, Tushar Khot, Tyler Shultz, Uri Shaham, Vedant Misra, Vera Demberg, Victoria Nyamai, Vikas Raunak, Vinay Venkatesh Ramasesh, vinay uday prabhu, Vishakh Padmakumar, Vivek Srikumar, William Fedus, William Saunders, William Zhang, Wout Vossen, Xiang Ren, Xiaoyu Tong, Xinran Zhao, Xinyi Wu, Xudong Shen, Yadollah Yaghoobzadeh, Yair Lakretz, Yangqiu Song, Yasaman Bahri, Yejin Choi, Yichi Yang, Sophie Hao, Yifu Chen, Yonatan Belinkov, Yu Hou, Yufang Hou, Yuntao Bai, Zachary Seid, Zhuoye Zhao, Zijian Wang, Zijie J. Wang, Zirui Wang, and Ziyi Wu. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=uyTL5Bvosj. Featured Certification.

Silviu-Marian Udrescu and Max Tegmark. Ai feynman: A physics-inspired method for symbolic regression. *Science Advances*, 6(16):eaay2631, 2020. doi: 10.1126/sciadv.aay2631. URL https://www.science.org/doi/abs/10.1126/sciadv.aay2631.

Eva van Tegelen, George van Voorn, Ioannis N. Athanasiadis, and Peter van Heijster. Neural ordinary differential equations for learning and extrapolating system dynamics across bifurcations. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 35(10):101103, 2025. doi: 10.1063/5.0288264. URL https://pubs.aip.org/cha/article/35/10/101103/3368889/Neural-ordinary-differential-equations-for.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Pantelis R. Vlachas, Wonmin Byeon, Zhong Y. Wan, Themistoklis P. Sapsis, and Petros Koumoutsakos. Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 474(2213):20170844, May 2018. doi: 10.1098/rspa.2017.0844.

Zheng-Meng Zhai, Celso Grebogi, and Ying-Cheng Lai. Can transformers predict system collapse in dynamical systems? *arXiv preprint arXiv:2605.04024*, 2026. URL https://arxiv.org/abs/2605.04024.

Lyra Zhornyak, M. Ani Hsieh, and Eric Forgoston. Inferring bifurcation diagrams with transformers. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 34(5):051102, 05 2024. ISSN 1054-1500. doi: 10.1063/5.0204714. URL https://doi.org/10.1063/5.0204714.

A Experimental Protocol

The scalar comparison uses a fixed 12-run neural experiment set and six polynomial baselines. High-order local verification and intervention experiments reuse the final trained checkpoints. The flow experiments have separate training supports and budgets, recorded in Appendices H, J, and K.

A.1 Data and Optimization

For each batch, we draw a shared sequence length T, then 512 independent control parameters uniformly in the training interval and 512 initial states uniformly in $[0.01, 0.99]$. The targets are generated by iterating the physical map in float32. No burn-in is discarded from training sequences. Let G have a geometric distribution with success probability $1 - \exp(-0.5)$ on $1, 2, \ldots$. We set $T = 4 + G - 1$; if this exceeds 64, we redraw T uniformly from the integers 4 through 64. Realized means for seeds 42–44 are 5.5405, 5.52695, and 5.54985, with maxima 22, 22, and 28. Both neural architectures use the same stream seed for each paired comparison.

The Transformer has width 16, two untied residual layers, two attention heads, feed-forward width 64, and maximum total context 256. Input projections have bias; attention projections, feed-forward projections, and the output head have no bias. Learned parameter/state type embeddings distinguish token roles. The MLP has three 64-unit SiLU hidden layers with biases and a scalar output. All models use teacher forcing at every state position. Training uses AdamW with its PyTorch default momentum coefficients $(0.9, 0.999)$ and $\epsilon = 10^{-8}$, 20,000 updates, learning rate 5×10^{-4}, weight decay 10^{-5}, 500 linear warm-up steps, and cosine decay to zero. The saved final checkpoint supplies the fixed-grid comparisons, high-order verification, and interventions. Training-stage figures additionally show intermediate checkpoints; the main overview and matched trajectory/cobweb examples use the explicitly labeled 10,000-update stage.

Ten neural runs were trained on an NVIDIA RTX 5060 Ti GPU. The two seed-44 MLP runs used CPU execution under the same sampling and optimizer budget. This scheduling choice was made without consulting OOD outcomes; per-run records identify the device. We do not use mixed-device wall-clock times as an efficiency comparison. The implementation used Python 3.13.9, PyTorch 2.9.1 with CUDA 12.8, NumPy 2.3.4, and SciPy 1.16.3.

A.2 Polynomial Baseline

The NVAR features are all monomials $\tilde{r}^i \tilde{x}^j$ with $i + j \leq d$, including a constant. We set $\tilde{x} = 2x - 1$ and map the training parameter interval affinely to $[-1, 1]$ without clipping test parameters. Candidate degrees are $d \in \{3, 5, 7\}$ and ridge strengths are $\lambda \in \{10^{-10}, 10^{-8}, 10^{-6}, 10^{-4}, 10^{-2}\}$. We fit

$$w = (\Phi^\top \Phi / N + \lambda I)^{-1} \Phi^\top y / N. \tag{17}$$

The fitting data use 64 batches from the same stream family as the neural models. Independent validation uses 16 batches with seed increased by 10,000. Selection minimizes ID validation MSE. Each logistic seed selects $(d, \lambda) = (3, 10^{-10})$ and each sine seed selects $(7, 10^{-10})$. Fitting and rollout use float64, with float32-generated training targets. This current-state polynomial baseline follows the NVAR representation principle; it is not a reproduction of the delayed features and parameter-channel construction of parameter-aware NG-RC.

B Measuring Structural Extrapolation

B.1 Distributional, Periodic, and Period-Doubling Structures

For each scalar family we evaluate the reference and learned systems on 513 uniformly spaced parameter values and three fixed initial conditions shared across all models. After $1{,}024$ autonomous iterations are discarded, the subsequent $N = 512$ states from each initial condition are retained. For

parameter r, we define the empirical distributional measures

$$\widehat{\mu}_r = \frac{1}{3N}\sum_{i=1}^{3}\sum_{t=1}^{N}\delta_{\hat{x}_t^{(i)}}, \qquad \mu_r = \frac{1}{3N}\sum_{i=1}^{3}\sum_{t=1}^{N}\delta_{x_t^{(i)}} \tag{18}$$

for the learned and reference systems, respectively. We compare them with one-dimensional Wasserstein-1 distance $W_1(\widehat{\mu}_r, \mu_r)$. However, this distributional comparison becomes inappropriate once nearby trajectories enter sensitive or chaotic regimes. Hence we evaluate periodicity separately. Candidate periods are tested in increasing order using temporal recurrence, within-phase consistency, and agreement of corresponding phase centers between earlier and later portions of the retained sequence. The smallest candidate satisfying all criteria is accepted. For $p > 1$, a tolerance is additionally constrained relative to the separation between distinct cycle phases, allowing small-amplitude cycles near a flip to be distinguished from a fixed point. Details specified in Appendix C.

A finite-resolution bifurcation diagram might be visually misleading near high-order transitions because of sparse parameter sampling, long transients, and unrelated periodic windows. We therefore separate candidate detection from numerical verification. A blind trajectory scan searches for consecutive $1 \to 2 \to 4 \to \cdots$ transitions without initializing the search at known reference bifurcation locations. Candidate transitions must be supported on both sides of the boundary and satisfy the prescribed consensus across initial conditions. Each accepted bracket, together with the next unresolved boundary, is then adaptively refined on a local parameter grid. At every refinement stage, trajectories are evaluated with two burn-in budgets B and $2B$; a candidate period must persist under the longer budget, and corresponding cycle phases from the two evaluations must agree up to cyclic alignment. For a candidate flip whose parent orbit has period p, we then independently solve

$$F_r^p(x) - x = 0, \qquad \Lambda_p(r, x) = -1, \tag{19}$$

with the cycle multiplier Λ_p of Equation 4. For the learned system this calculation uses the one-state map $F_{\theta,r}$ of Equation 11, with derivatives obtained by automatic differentiation, whereas analytic derivatives are used for the reference scalar maps. A transition is reported as numerically verified only if the joint solution lies inside the independently trajectory-derived bracket and continuation of the parent orbit to parameters on both sides confirms the corresponding stable-to-unstable crossing. For the verified flip parameters $r_1, \ldots, r_K$ we compute the finite-order interval ratios δ_n of Equation 5, and we report the physical locations r_n and the ratios δ_n separately. The former measure alignment of the learned and reference parameterizations, whereas the latter measure the relative geometry of the cascade.

To track the emergence of organizing patterns during training, we evaluate both quantities at fixed training checkpoints. Model snapshots are recorded every 250 updates. At each snapshot, one-step prediction is evaluated on fixed sets of independently sampled state–parameter pairs from both the training parameter range and the OOD range, and we also recompute the closed-loop distributional and periodic diagnostics above. This paired evaluation allows changes in the aforementioned properties to be compared on the same learned model without treating any one of them as a proxy for the others. More details can be found in Appendices G and D.)

B.2 Evaluation of Continuous Flows

For the continuous-time systems, the relevant long-run structures are not adequately described by scalar period labels, so we use system-specific diagnostics on the frozen discrete flow predictor of Equation 14.

For the joint Lorenz experiment, the primary coarse comparison classifies each parameter pair as fixed or nonfixed from the long-time trajectory, using the same numerical criterion that defines the fixed-destination training parameters. On nonfixed trajectories we additionally measure marginal state distributions and repeated motion between the two wings. Because repeated wing switching alone does not establish chaos, to test chaotic instability directly, we estimate the largest Lyapunov exponent of the frozen learned map by propagating a tangent vector through successive learned updates and repeatedly renormalizing it. A positive estimated largest exponent provides evidence of exponential separation along the generated attractor under the specified finite-time estimator. The wider two-parameter Lorenz analysis additionally identifies resolved periodic behavior through recurrence of crossings of a fixed Poincaré-section. Minimal recurrence lags are tested over a prescribed range with fixed crossing-count and recurrence-distance criteria.

Table 1: Final adaptive budgets. Each level compares two burn-in budgets; "maximum" is the longer final budget. Neural maps use logistic seed 42 and one current state.

Map	Levels	Max. burn	Tail
Logistic reference	4	65,536	2,048
Sine reference	4	65,536	2,048
MLP	5	131,072	2,048
Transformer	4	32,768	1,024

For the generalized Hopf system, the principal asymptotic distinction is between attraction to the origin and attraction to a periodic orbit. We therefore evaluate long-time radius and angular motion from multiple initial radii and phases. Point outcomes require sufficiently small mean radius, while cycle outcomes require sustained nonzero radius together with low radial variation and angular speed close to the reference rotation rate. Multiple initial conditions at the same parameter value allow coexistence of attracting states to be tested directly rather than inferred from a single trajectory. The operational thresholds and evaluation windows are fixed in advance and specified in Appendix J.

The reference and learned systems are evaluated using the same estimator. For more details, see Appendix I).

C Evaluation and Period Detection

Uniform-grid evaluations use 513 parameter values over $[2, 4]$ or $[0.1, 1]$. Three initial states are sampled once from $[0.05, 0.95]$ using NumPy generator seed 20260912 and shared by every model and reference. We discard 1,024 iterations and retain the next 512. Neural inference uses float32 and reference inference uses float64. Predicted states are clipped to $[0, 1]$. At each parameter, W_1 is computed over all 3×512 retained observations. OOD averages include parameters strictly above the training upper endpoint.

For a retained scalar sequence z_t, the base tolerance is $\epsilon = 10^{-5} + 10^{-4}(\max z - \min z)$. Candidate periods are tested in increasing order, up to $\min(64, \lfloor N/6 \rfloor)$, with at least 64 samples. The 0.99 quantile of $|z_{t+p} - z_t|/\epsilon$ must be at most one. Values are then arranged by cycle phase. Within-phase deviations from phase medians and the change between early and late phase medians must also satisfy the tolerance. For $p > 1$, the phase tolerance is capped at one percent of the minimum separation between distinct phase medians. This additional relative check resolves small-amplitude oscillations close to a flip. An unaccepted sequence is unresolved, which includes aperiodicity, long transients, and insufficient resolution; it is not classified as chaos.

Parameter-level cascade detection requires at least 80% initial-condition agreement, which requires all three initial states here. Both sides of a transition need three consecutive grid points. The maximum bracket width is 3% of the scanned parameter span. A primary cascade must proceed consecutively from period one without jumping a contradictory periodic region. Period agreement in the main comparisons is evaluated at individual parameter–initial-state pairs, over all evaluated parameters for which the reference period is accepted. It is distinct from the OOD-only W_1 average.

C.1 Local Refinement and Numerical Roots

The high-order analysis uses up to period 128, three fixed initial states $(0.123456789, 0.371239, 0.817321)$, and 33-point local grids. The initial uniform scans locate candidates without using theoretical bifurcation values. The reference scans use 2,049 parameters; neural scans reuse their main 513-point arrays. Each local stage tests burn-in budgets B and $2B$ and checks phase-center agreement up to a cyclic shift, in addition to period labels. Previously retained tails allow reanalysis of convergence criteria without changing learned weights.

For every parent period $p = 1, 2, 4, \ldots, 64$, the independent root calculation solves orbit closure and multiplier -1 jointly. It uses analytic reference derivatives or autograd derivatives of the float64 neural map. Continuation to either side checks the parent cycle's loss of stability. All reported solutions lie inside the independently detected trajectory brackets. Table 2 lists the resulting seven flip parameters, and under the same definition the sine reference ladder reaches a final finite-order ratio of 4.668805. The bracket widths represent spatial/recognition resolution, not statistical confidence intervals. Tests of the detector cover analytic first flips, known periods, unresolved tails, missing

Table 2: All numerically verified flip parameters. Learned maps use logistic seed 42. The Transformer receives one state token and a parameter token.

Index	Logistic reference	Sine reference	Logistic MLP	Logistic Transformer
1	3.0000000000	0.7199616830	2.9879977074	2.9692234640
2	3.4494897428	0.8332663537	3.4523234903	3.3927447143
3	3.5440903596	0.8586090599	3.5573408693	3.4848703460
4	3.5644072661	0.8640841737	3.5802157502	3.5048663661
5	3.5687594195	0.8652589607	3.5851355903	3.5091579915
6	3.5696916098	0.8655106638	3.5861901963	3.5100776679
7	3.5698912594	0.8655645755	3.5864161045	3.5102746561

Table 3: **OOD distributional fidelity.** Uniform-grid W_1 in units of 10^{-3}, mean $\pm$ sample SD over three training seeds. C counts state tokens, excluding the parameter token. NVAR is a structural baseline with a separate, smaller training budget.

Model	Logistic	Sine
TF, $C = 1$	27.86 ± 6.76	34.42 ± 6.20
TF, $C = 4$	34.02 ± 16.32	33.29 ± 16.39
TF, $C = 255$	129.77 ± 81.24	82.32 ± 36.12
MLP	22.97 ± 1.53	46.23 ± 6.66
Polynomial NVAR	2.15 ± 0.13	2.24 ± 0.16

boundaries, later periodic islands, and slowly decaying oscillations. The reported sequences require cross-budget phase agreement, not only matching period labels.

C.2 Precision of the Verified Cascade

The roots above are solved on float64 maps while the headline rollouts and comparisons use float32 inference. Re-evaluating the seven verified logistic flips of the seed-42 Transformer inside the float32 model gives closure residuals between 1.2×10^{-7} and 2.1×10^{-5} and multipliers within 2% of -1 for all seven flips. One-sided period probes placed at the midpoint of each inter-flip window reproduce the expected parent or daughter period in six of seven cases in float32 and in six of seven cases in float64; the unmatched cases are the period-64 and period-128 windows, whose width is about 2×10^{-4}. The verified structure is therefore not an artifact of float64 evaluation, but the two highest-order transitions are not resolvable at deployment precision.

D Baselines and Distributional Fidelity

The baselines are a parameter–state multilayer perceptron (MLP) and a polynomial nonlinear vector autoregression (NVAR) with an explicit state–parameter feature library. The MLP evaluates $m_\theta(r, x_t)$ directly from concatenated numerical inputs with three hidden layers of width 64 and SiLU activations, giving 8,577 parameters, so it learns the same transition target without a token communication stage. The NVAR uses all monomials of the current state and parameter up to degree 3, 5 or 7 with a ridge penalty, selects degree and penalty on independent in-distribution one-step validation, and its logistic fits use degree 3, a library that can express the logistic update exactly.

Table 3 reports the 30 fixed-grid comparisons as means and sample standard deviations over three training seeds, and Table 4 lists every model–context–seed combination. Error bars use the sample standard deviation of three independently trained models; trajectory samples are not treated as independent training replications. No significance test is used to claim a universal ordering of architectures.

Finite trajectory sampling contributes to these distances. Repeating the reference calculation with different initial states gives mean OOD distances of 0.00210 on logistic and 0.00224 on sine, and the polynomial errors are on this scale. A separate reference-versus-reference calculation uses 16 independent initial-state ensembles under the same sampling budget, yielding mean OOD W_1 of 0.002104 (SD 0.000092) for logistic and 0.002239 (SD 0.000188) for sine. This quantifies sampling variability under the evaluation budget; it is not a strict lower bound on model error.

Table 4: All fixed-grid evaluations. ID/OOD columns are mean W_1 over the corresponding parameter subset. Period agreement is over accepted reference pairs on the full grid; unknown model periods do not count as correct.

Family	Model / context	Seed	ID W_1	OOD W_1	Period agreement	Coarse flips
Logistic	TF, $C = 255$	42	0.087319	0.036792	73.6%	1
Logistic	TF, $C = 1$	42	0.001788	0.034595	70.2%	3
Logistic	TF, $C = 4$	42	0.002201	0.030639	66.4%	3
Logistic	MLP	42	0.000503	0.023623	90.0%	3
Logistic	Polynomial NVAR	42	0.000000	0.002137	99.8%	3
Sine	TF, $C = 255$	42	0.004700	0.054363	78.1%	3
Sine	TF, $C = 1$	42	0.007186	0.041563	74.2%	3
Sine	TF, $C = 4$	42	0.003129	0.049795	69.9%	3
Sine	MLP	42	0.002029	0.038802	73.1%	3
Sine	Polynomial NVAR	42	0.000005	0.002066	100.0%	3
Logistic	TF, $C = 255$	43	0.105557	0.165516	0.0%	1
Logistic	TF, $C = 1$	43	0.000497	0.027913	84.3%	3
Logistic	TF, $C = 4$	43	0.000487	0.019663	86.8%	3
Logistic	MLP	43	0.000440	0.024058	88.9%	3
Logistic	Polynomial NVAR	43	0.000000	0.002292	99.8%	3
Sine	TF, $C = 255$	43	0.027612	0.069479	62.4%	3
Sine	TF, $C = 1$	43	0.003296	0.031190	78.5%	3
Sine	TF, $C = 4$	43	0.001037	0.033045	79.6%	3
Sine	MLP	43	0.001938	0.048183	71.0%	3
Sine	Polynomial NVAR	43	0.000005	0.002380	100.0%	3
Logistic	TF, $C = 255$	44	0.069670	0.187007	0.0%	0
Logistic	TF, $C = 1$	44	0.000450	0.021069	81.1%	3
Logistic	TF, $C = 4$	44	0.000298	0.051771	67.9%	0
Logistic	MLP	44	0.000139	0.021218	87.9%	3
Logistic	Polynomial NVAR	44	0.000000	0.002023	99.8%	3
Sine	TF, $C = 255$	44	0.063694	0.123104	10.8%	3
Sine	TF, $C = 1$	44	0.003274	0.030495	73.1%	3
Sine	TF, $C = 4$	44	0.000800	0.017017	84.9%	3
Sine	MLP	44	0.002315	0.051689	71.0%	3
Sine	Polynomial NVAR	44	0.000005	0.002289	100.0%	3

The four-state Transformer retains substantially better fidelity than the 255-state rollout. Period agreement on accepted reference trajectories is 73.7% versus 24.5% on logistic and 78.1% versus 50.4% on sine. Since the latter context far exceeds realized training lengths, this contrast measures sensitivity to length extrapolation together with the learned computation. It cannot by itself establish an intrinsic disadvantage of long-context attention, and the 255-state rows show in-distribution W_1 above the one-state out-of-distribution values on every seed, so they are best read as a configuration that fails in distribution rather than as a clean context-length probe.

The scalar W_1 ordering measures a different property from the cascade: the MLP leads on logistic and the Transformer on sine, and the degree-3 polynomial reaches the sampling floor on a family whose update it can express exactly. Reproducing the visited-state distribution and reproducing the cascade are therefore separate achievements, as the intervention results of Appendix L and the class diagnostics of Appendix F make concrete.

The same comparison at the level of verified flips is reported here in full. The MLP's cascade reaches the same order, with a final finite-order ratio of 4.668296 against 4.669132 for the reference map and 4.668686 for the Transformer, and its flip parameters are displaced in the opposite direction: the seventh flip lies at 3.586416 for the MLP against 3.510275 for the Transformer and 3.569891 for the reference. Both architectures therefore reproduce the quadratic-cascade organization with a seed-specific displacement. At the final checkpoint the three training seeds displace the same reference ladder in different directions, from -0.060 to -0.031 for seed 42, from $+0.001$ to $+0.036$ for seed 43 and from -0.020 to -0.005 for seed 44, an across-seed spread of 0.096 in r (Figure 2).

E Observation-Window Ablation

The ablation in Section 4.1 retrains the same scalar Transformer on four training parameter windows with everything else fixed: seed 42–44, 20,000 AdamW updates, batch size 512, the short-sequence length distribution of Appendix A, and the final checkpoint. Only the interval from which parameters are drawn changes; no evaluation signal enters training or checkpoint selection. Each run takes 209–274 s on a single CPU machine in our reproduction. The evaluation reuses the 513-point physical grid, three fixed initial states, 1,024 discarded steps and 512 retained states of Appendix C.

Table 6 summarizes the four windows, and Table 5 lists every seed.

Table 5: **Per-seed window ablation.** Same protocol for every row. W_1 is the uniform-grid OOD average; agreement is over reference-periodic parameter–initial-state pairs; coarse flips count consecutive detected doublings on the 513-point grid.

Training window	Seed	ID W_1	OOD W_1	Agreement	Coarse flips
$[2.0, 3.0]$	42	0.00179	0.03456	70.2%	3
	43	0.00050	0.02772	84.0%	3
	44	0.00045	0.02108	81.3%	3
$[2.5, 3.0]$	42	0.00077	0.03172	90.0%	3
	43	0.00073	0.03940	88.2%	3
	44	0.00389	0.06319	76.9%	0
$[2.9, 3.0]$	42	0.00415	0.03756	85.1%	3
	43	0.00138	0.04022	86.9%	3
	44	0.00149	0.03047	92.6%	3
$[2.0, 2.5]$	42	0.00100	0.06865	63.9%	3
	43	0.00022	0.16082	67.8%	3
	44	0.00068	0.06104	75.5%	3

Table 6: **Observation-window ablation.** All four windows use the locked protocol of Appendix A and are evaluated on the same 513-point grid. OOD W_1 is in units of 10^{-3}, mean $\pm$ sample SD over three training seeds. "Verified flips" counts consecutive flips passing the local root and multiplier checks for seed 42 (three refinement rounds for every window).

Training window	OOD W_1	Periodic agreement	First verified flip	Verified flips
$[2.0, 3.0]$	27.8 ± 6.7	78.5%	2.9692	7
$[2.5, 3.0]$	44.8 ± 16.4	85.0%	3.0170	6
$[2.9, 3.0]$	36.1 ± 5.0	88.2%	2.9884	6
$[2.0, 2.5]$	96.8 ± 55.5	69.1%	2.9304	6
Reference	—	—	3.0000	7

Adaptive verification of the ablation arms. Each window was refined with the blind candidate search, paired burn-in budgets and phase-center agreement of Appendix C, followed by the independent closure-plus-multiplier solve. Three refinement rounds were used for every arm, whereas the published $[2, 3]$ baseline used four levels, so the count of verified flips is budget-limited for the new arms rather than capability-limited.

Table 7: **Verified flips per training window** (seed 42). Parameters are the flip parameters verified by orbit closure and multiplier -1 with two-sided continuation; δ_n follows Equation (5). The last column shows the largest finite-order ratio reached in the available levels.

Training window	Verified flips	First three flip parameters	Largest δ_n
$[2.0, 3.0]$ (published)	7	2.9692235, 3.3927447, 3.4848703	4.6687
$[2.5, 3.0]$	6	3.0169678, 3.4462607, 3.5237060	4.6731
$[2.9, 3.0]$	6	2.9884133, 3.3770377, 3.4491587	4.6745
$[2.0, 2.5]$	6	2.9303844, 3.2847299, 3.3538837	4.6739
Logistic reference	7	3.0000000, 3.4494897, 3.5440904	4.6691

The early ratios differ between windows because the displaced first flip changes the interval lengths from which the first ratios are formed; all arms converge towards the same same-definition reference value as more levels are resolved. The window therefore controls where the cascade sits and how fast the finite-order sequence approaches the reference, but not whether the cascade exists.

Provenance. Training applies the locked protocol with the parameter interval overridden, and refinement reuses the candidate search, cross-budget period test, temporal confirmation and multiplier solve of Appendix C.

F Universality Class Diagnostics

A finite-order ratio close to 4.6692 shows membership of a universality class, not identification of a system: the ratio is fixed by the order of the map's maximum. This appendix separates the two, using two descriptors that the main text does not report.

Critical exponent and Schwarzian derivative. For each map we locate the maximum x_c numerically, fit $\log|F(x) - F(x_c)|$ against $\log|x - x_c|$ on both sides of x_c for offsets 10^{-4} to 10^{-2}, and evaluate the Schwarzian derivative $SF = F'''/F' - 1.5\,(F''/F')^2$ on the state interval where $|F'| > 0.05$. A negative Schwarzian together with a single critical point is the classical sufficient condition for a full period-doubling cascade.

Table 8: **Class diagnostics** at an out-of-distribution parameter. z is the fitted order of the maximum ($z = 2$ quadratic, $z = 4$ quartic). Derivatives are autograd first derivatives of the float64 map; second and third derivatives are finite differences of that exact first derivative.

Map	Fitted z	Reference z	Schwarzian negative
Logistic Transformer, $C = 1$	2.000	2.000	97.3% (reference 100%)
Logistic MLP	2.021	2.000	100% (reference 100%)
Sine Transformer, $C = 1$	2.100	2.000	96.6%
Unimodal $p = 4$ Transformer	**2.005**	3.932	100%

A family with a different universality class. We train the same architecture on the family $f(x) = 1 - \mu|x|^4$ (quartic maximum, pre-bifurcation interval $[0.1, 0.4785]$) and run the identical detection and verification pipeline on the reference and on the learned map, evaluating the cascade region $\mu \in [0.4, 1.7]$.

Table 9: **Quartic family.** Reference flips and finite-order ratios obtained with the manuscript's pipeline; the literature value for the quartic class is $\delta \approx 7.2847$, clearly separated from the quadratic 4.6692. The learned map is verified at one flip only, and its fitted critical exponent is quadratic.

Quantity	Reference	Transformer
Verified flips	6	1
First flip	0.4882812	0.4987569
Finite-order ratios ($n = 2..4$)	5.580, 6.963, 7.283	—
Fitted critical exponent z	3.932	2.005
Coarse flips in $\mu \in [0.4, 1.7]$	3	1
OOD W_1 (three seeds)	—	0.0982, 0.3668, 0.0962

The learned map has an approximately quadratic critical point despite being trained on a quartic family, and it fails to reproduce the quartic cascade: only one flip is verified. Thus no learned finite-order scaling ratio can be estimated in this experiment. Together with the quadratic-family results, this suggests that matching the Feigenbaum scaling in Section 4.1 reflects the universality class of the learned map rather than identification of the underlying analytic family.

Function-level errors. The same measurements give a direct reading of how well the learned transition matches the physical one, beyond the distributional distances of Appendix D. Averages are over the out-of-distribution parameter slice; the reference mean $|\partial_r F|$ is 0.173 for logistic and 0.660 for sine.

Table 10: **Map-level diagnostics** (out-of-distribution slice, unclipped float64 map except where noted). $\sup_x |F_\theta - F|$ is in physical state units; $\partial_r^2 F_\theta$ is spurious parameter curvature, identically zero for the reference.

Map	$\sup_x \lvert F_\theta - F \rvert$	mean $\lvert \Delta \partial_x F \rvert$	mean $\lvert \Delta \partial_r F \rvert$	mean $\lvert \partial_r^2 F_\theta \rvert$
Logistic Transformer, $C = 1$	0.041	0.166	0.036	0.073
Logistic MLP	—	0.048	0.024	0.041
Sine Transformer, $C = 1$	—	0.079	0.121	0.638
Window $[2.9, 3]$	0.102	—	0.143	0.138
Window $[2, 2.5]$	0.103	—	0.143	0.138
Parameter token permuted	0.255	—	0.164	0.012

Two features of this table matter for the interpretation of the main results. The parameter derivative of the learned map differs from the physical one by roughly a fifth of the physical scale, and the map carries a non-zero parameter curvature that the physical family does not have. Both are consistent with a learned map that reproduces the organization of the family, including its universality class, while remaining a smooth surrogate rather than a recovered equation.

G How Dynamics Develop During Training

We record training dynamics for seed 42 on each system using the original 20,000-update optimization protocol. Weight snapshots are saved every 250 updates, including initialization, giving 81 stages per system. Training minibatch loss is recorded at each saved update. After training, every snapshot is evaluated on the same fixed 2,048 ID and 2,048 OOD one-state examples: parameters are uniform within each region and states are uniform on $[0.01, 0.99]$. These pointwise diagnostics are distinct from autonomous trajectory fidelity.

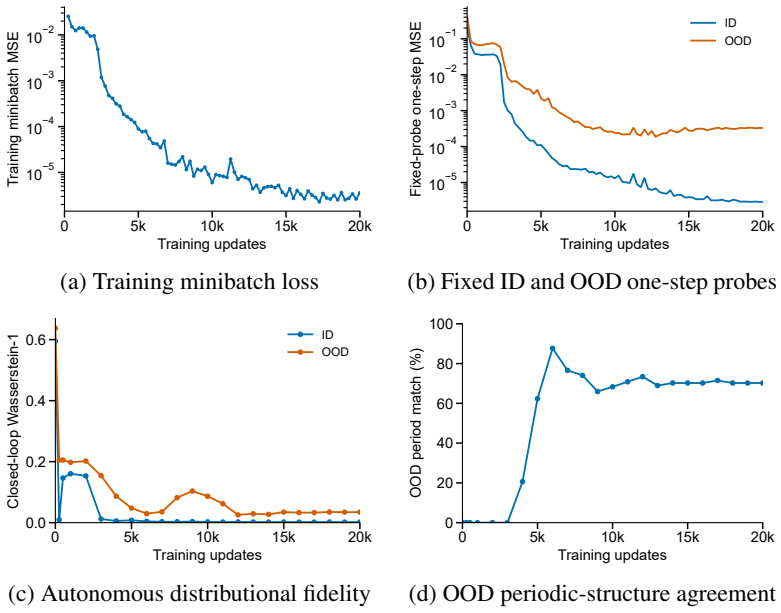
Closed-loop evaluations use all 23 predeclared stages: initialization, 250 and 500 updates, and every 1,000 updates through 20,000. Each uses the original 513-point physical grid, the same three initial states, 1,024 burn-in steps, 512 retained states, and $C = 1$. One-step predictions are unclipped; autonomous rollouts use the $[0, 1]$ clamp. The figures below display every evaluated stage. Error and structure curves connect these fixed stages; connecting lines do not resolve intermediate events. Period matching is evaluated only where reference periodicity is confirmed. The coarse flip count uses the trajectory detector on this uniform grid and is separate from adaptive high-order verification.

The final replayed weights match the original reported checkpoint exactly, tensor by tensor, on both systems. These longitudinal views therefore end at the same learned maps used in the main comparisons.

G.1 Structural Emergence and Physical Alignment

The training records separate the appearance of a cascade from its quantitative alignment with the physical system (Figures 9 and 10). In the logistic replay, the fixed-grid detector confirms no doubling through the 2,000-update recording, one at 3,000, and three at 4,000. At 4,000 updates, however, the first detected transition is near $r = 3.32$, compared with the physical value $r = 3$. A recognizable cascade therefore appears while its location is still moving. The sine replay already has two detected doublings at the 1,000-update recording and three at 2,000; subsequent coarse counts and locations continue to vary.

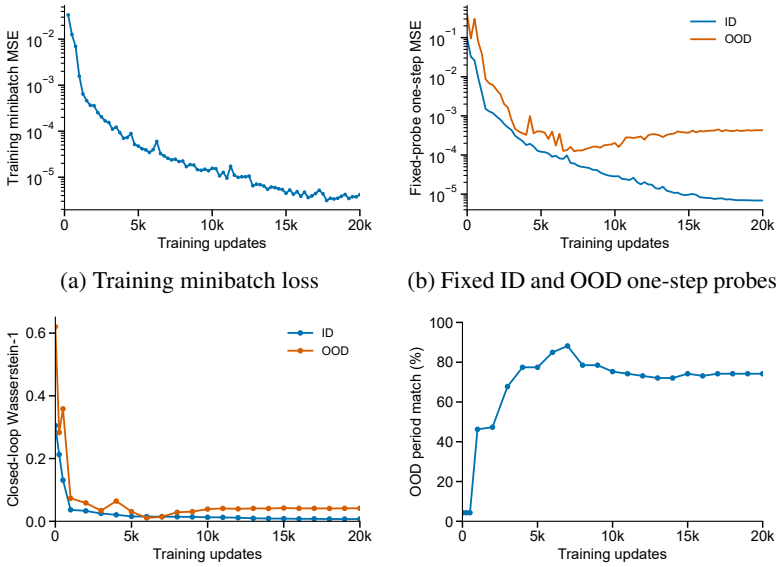
One-step fitting and autonomous fidelity follow different trajectories (Figure 7). For logistic, the fixed-probe OOD one-step MSE decreases from 1.10×10^{-3} at 6,000 updates to 2.41×10^{-4} at 10,000, while closed-loop OOD W_1 rises from 0.0296 to 0.0867. Across these stages, three doublings remain confirmed on the coarse grid. The learned update becomes more accurate on the sampled one-step task while its generated long-run distribution moves non-monotonically. Together, the stage diagrams, location curves, and fidelity curves distinguish learning a period-doubling structure from reproducing its precise physical realization.



(a) Training minibatch loss (b) Fixed ID and OOD one-step probes

(c) Autonomous distributional fidelity (d) OOD periodic-structure agreement

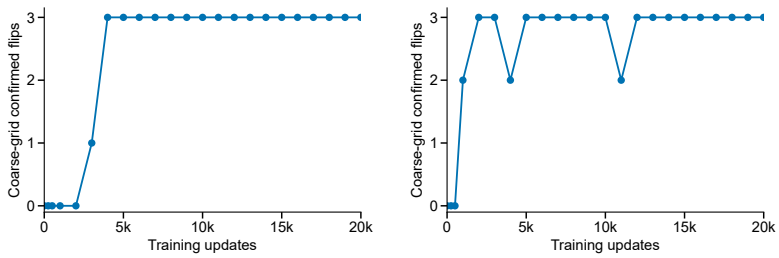
Figure 7: **Logistic learning curves.** Training and probe measurements are recorded every 250 updates; autonomous diagnostics use the 23 fixed stages. One-step probes use 2,048 samples per region. Period matching uses confirmed periodic reference trajectories, with unresolved model outputs counted as mismatches. All curves concern seed 42 and one state token.



(a) Training minibatch loss (b) Fixed ID and OOD one-step probes

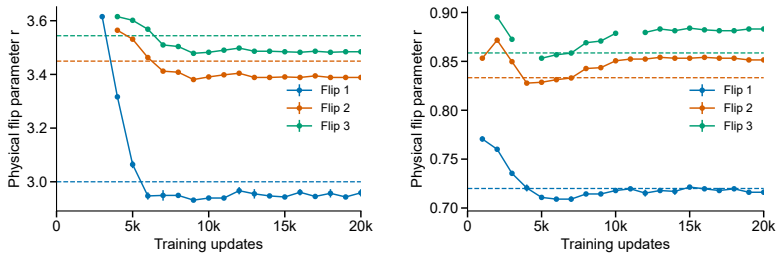
(c) Autonomous distributional fidelity (d) OOD periodic-structure agreement

Figure 8: **Sine learning curves.** Training and probe measurements are recorded every 250 updates; autonomous diagnostics use the 23 fixed stages. One-step probes use 2,048 samples per region. Period matching uses confirmed periodic reference trajectories, with unresolved model outputs counted as mismatches. All curves concern seed 42 and one state token.



(a) Logistic (b) Sine

Figure 9: **Coarse structural detection across training.** Each point counts consecutive confirmed period doublings on the fixed 513-point grid. Zero denotes no cascade confirmed at this budget. Adaptive high-order verification uses the separate local protocol.



(a) Logistic (b) Sine

Figure 10: **Flip parameters move during training.** Points show the first three detected flip parameters and vertical bars show their trajectory brackets. Dashed horizontal lines mark the reference flip parameters, solved independently of the trajectory detector. Unresolved detections appear as gaps. These curves track geometric displacement separately from the number of detected doublings.

G.2 Logistic Attractor Evolution

G.3 Sine Attractor Evolution

H Lorenz Protocol, Training Regions, and Parameter Atlas

System and parameter support. We use

$$\dot{x} = 10(y - x), \qquad \dot{y} = \rho x - y - xz, \qquad \dot{z} = xy - \tfrac{8}{3}z. \tag{20}$$

For these fixed coefficients, the origin loses stability at $\rho = 1$. A homoclinic explosion occurs near 13.9265, producing a chaotic saddle and potentially long chaotic transients while the two nonzero equilibria remain stable. Near 24.0579, a chaotic attractor appears and coexists with the stable equilibria until their Hopf bifurcation at $470/19 \approx 24.7368$ (Doedel et al., 2011). We use these numerical boundaries to define three sampling regions:

$$I_0 = (0, 1), \qquad I_1 = (1, 13.9265), \qquad I_2 = (13.9265, 24.0579). \tag{21}$$

The main mixed condition allocates 171, 171, and 170 training trajectories to these intervals, respectively, sampling uniformly within each interval. Independent validation uses 43, 43, and 42 trajectories. Initial coordinates are sampled uniformly from $[-10, 10] \times [-10, 10] \times [0, 30]$. Actual training parameters range from 0.0018139074 to 23.8602887306; 24.0579 is the upper sampling boundary, not the largest realized sample. No basin filtering or convergence-based rejection is applied. The short observations may contain rotations and cross-wing motion. An additional 100-unit oracle continuation is diagnostic only and is not used for learning; finite-time nonconvergence is not evidence of an asymptotic chaotic attractor.

Model, supervision, and fixed checkpoint. Each trajectory contributes 256 transitions at $\Delta t = 0.02$, giving 131,072 training pairs. The oracle uses float64 RK4 with four substeps of size 0.005. The Transformer has two layers, hidden width 64, four heads, and feed-forward width 256, with one state token plus a parameter token. It predicts a normalized increment added to the current state; the physical vector field is not supplied to the model. State means, state standard deviations, and increment standard deviations are estimated from the training set alone; the parameter input is

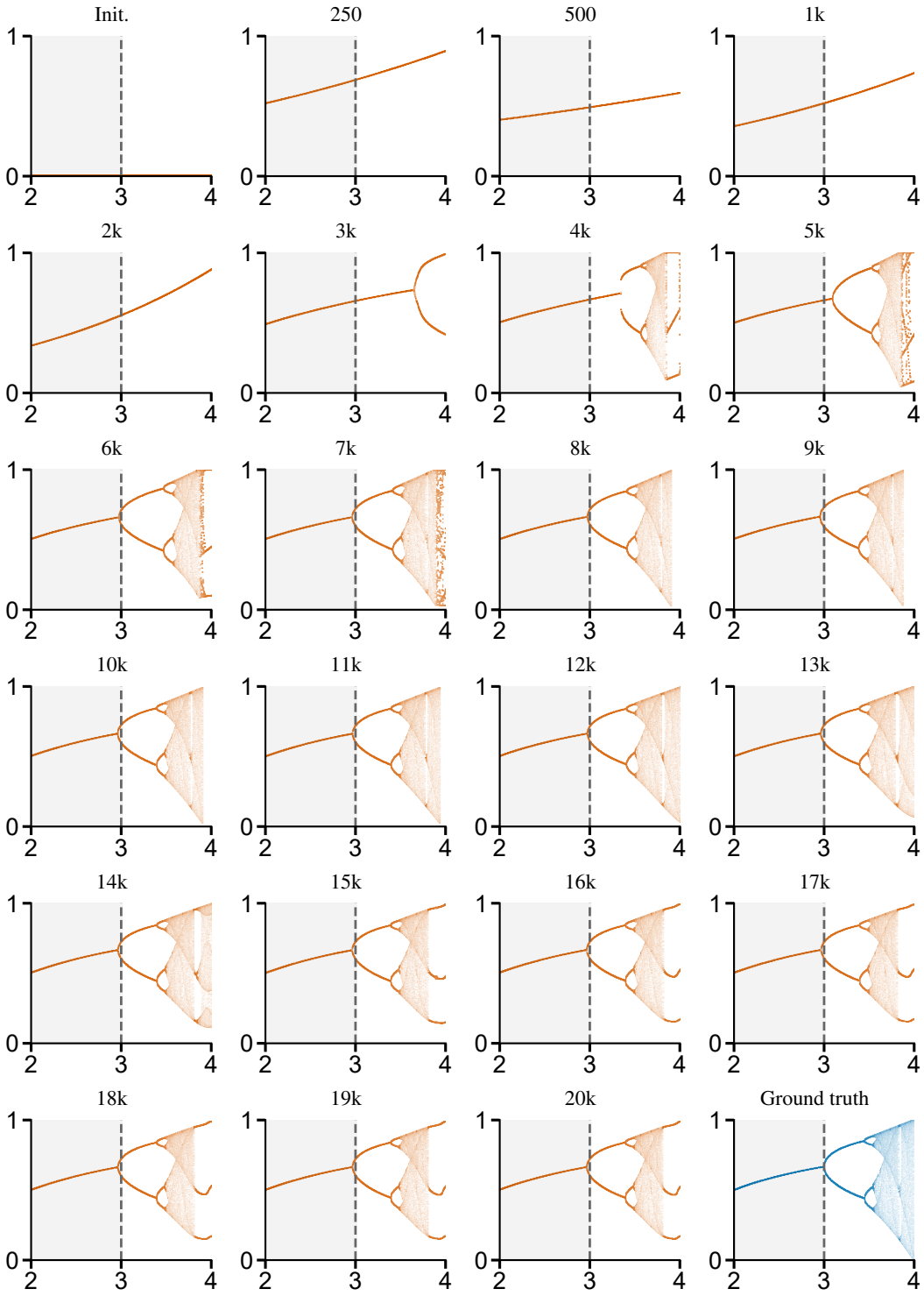


Figure 11: **Logistic attractor evolution.** All 23 closed-loop stages plus the ground-truth reference, arranged as four columns and six rows. Panels use the same compact style as Figure 3: no colorbars or axis titles, ticks only at the range endpoints and training boundary, and gray shading for the training interval. Orange denotes the Transformer and blue the reference.

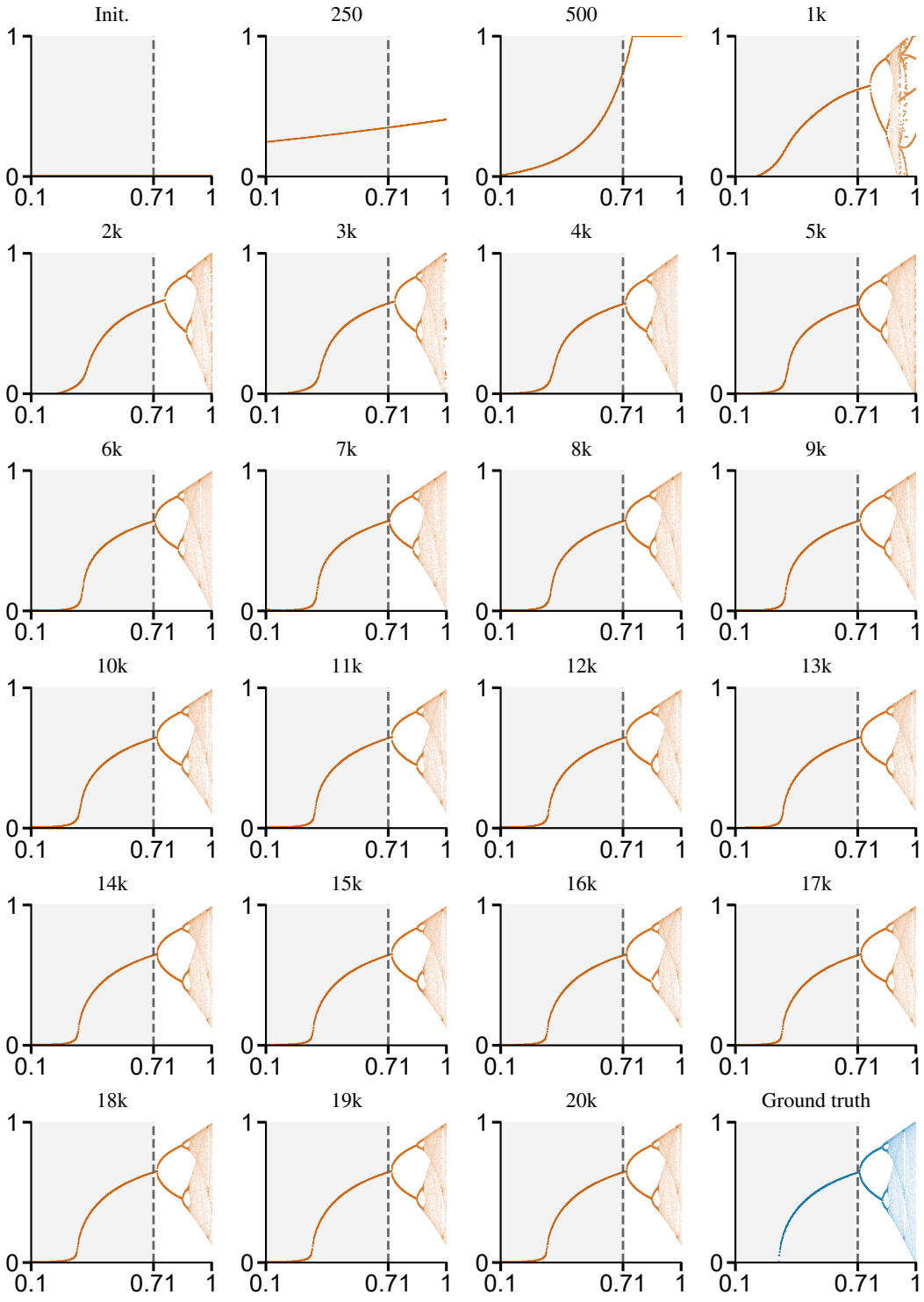


Figure 12: **Sine attractor evolution.** All 23 closed-loop stages plus the ground-truth reference, arranged as four columns and six rows. Panels use the same compact style as Figure 3: no colorbars or axis titles, ticks only at the range endpoints and training boundary, and gray shading for the training interval. Orange denotes the Transformer and blue the reference.

$(\rho - 12.5)/7.5$, a fixed affine encoding across conditions. Training uses seed 42, 10,000 AdamW updates, batch size 128, weight decay 10^{-5}, gradient clipping at 1, 500 warm-up steps, and cosine decay from 5×10^{-4} to 5×10^{-5}. The data seed is 20260917 and the mixed-region assignment seed is 20260919. The final checkpoint is used throughout, without selecting weights using OOD trajectories.

Quantitative tests and region controls. The primary OOD parameters are 28 and 32. At each parameter we evaluate four common positive initial states for 100 time units and score the tail $[50, 100]$. Table 11 holds the architecture, model seed, total trajectory count, initial-state draws, and optimization budget fixed while varying parameter support. Normalization statistics are estimated separately in each condition; scores are converted to shared physical units before comparison. Specifically, marginal Wasserstein distances are divided by the state standard deviations of the original broad $\rho \in [5, 20]$ training dataset, averaged over three state coordinates and then over four initial states. This reference dataset defines the metric scale only; it does not contribute training samples to the new conditions. Repeated switching denotes observed changes of the sign of x in the retained window and does not alone certify a positive Lyapunov exponent or identify a parameter-region boundary.

Table 11: **Training-region controls at a fixed final checkpoint.** One model seed per condition; 512 training trajectories each. Mixed uses balanced counts over I_0, I_1, I_2. Switching counts trajectories across the two OOD parameters and four initial states, not independent model seeds.

Training support	$W_1(28)$	$W_1(32)$	Repeated switching
I_0	2.3415	2.7005	0/8
I_1	1.7427	1.9778	0/8
I_2	0.1973	0.2857	8/8
Mixed	0.1022	0.2507	8/8

The mixed and I_2 conditions reproduce sustained double-wing motion in these tests. The I_0 and I_1 conditions do not do so at the fixed checkpoint, showing dependence on the observation region under this protocol rather than establishing an architectural impossibility.

Main-figure provenance. Figure 15 uses the 20 parameter values listed below. Every panel overlays the same four previously fixed test initial states, approximately $(4.981937, 4.941258, 9.182749)$, $(6.452753, 6.452225, 14.340579)$, $(3.399327, 3.401437, 4.343713)$, and $(7.300586, 7.071906, 19.377221)$. All panels retain $t \in [0, 5.12]$ without burn-in, so this figure compares equal-duration transients rather than final attractors. Arrows follow the displacement from $t = 0.3$ to 0.4. The x–z projection has shared limits $[-32.5, 32.5] \times [-5, 60]$, equal physical aspect ratio, and opacity 0.32. State axes are suppressed and only ρ is labeled. These are new parameter/initial-state combinations, not training observations. Independent validation remains separate.

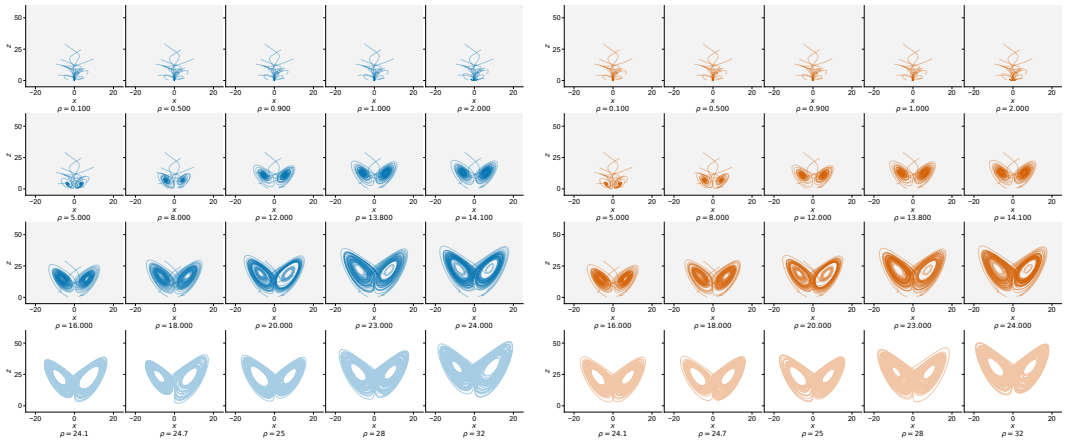
Twenty-parameter atlas. The fixed values, in row-major order, are

$$\begin{gathered} 0.1,\ 0.5,\ 0.9,\ 1,\ 2; \\ 5,\ 8,\ 12,\ 13.8,\ 14.1; \\ 16,\ 18,\ 20,\ 23,\ 24; \\ 24.1,\ 24.7,\ 25,\ 28,\ 32. \end{gathered} \tag{22}$$

This grid spans the origin regime, the pitchfork boundary, the vicinity of the homoclinic transition, the coexistence region, and the standard chaotic probes; it was fixed before computing the new atlas predictions. For mixed training, the first 15 panels are within the sampling support and the final five are OOD. The isolated boundary $\rho = 1$ is treated as an in-support interpolation probe, although a continuous sampler has zero probability of observing that exact value. For I_2 training, gray panels cover $14.1, 16, 18, 20, 23, 24$; all remaining displayed values lie outside its training interval. The finite atlas is a visualization of selected parameters, not a numerical determination of transition values.

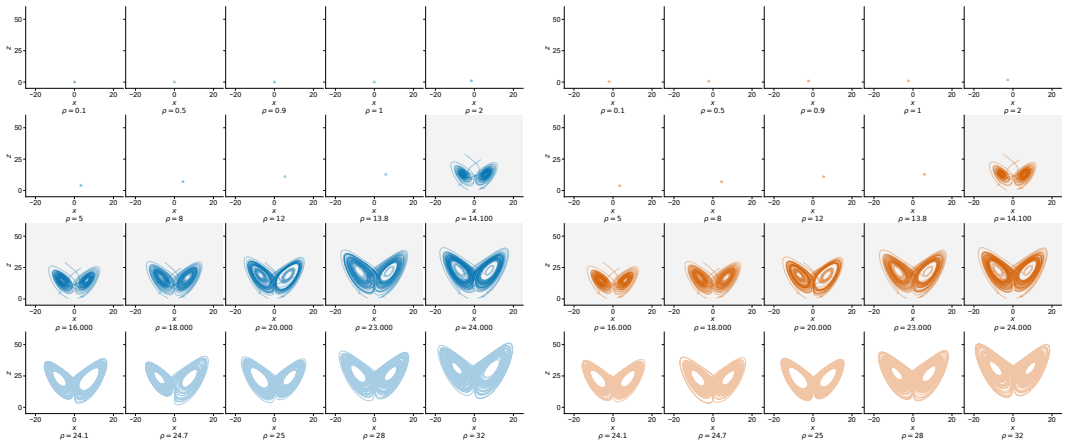
In the supplementary atlases below, gray panels overlay 16 trajectories over $[0, 5.12]$; white panels show one fixed initial state over $[50, 100]$. These supplementary windows deliberately distinguish short-transient fitting from autonomous long-time behavior, unlike the equal-window main figure.

These window types differ in horizon and are not comparable as convergence tests. All coordinates retain the same physical scale, with no alignment, centering at equilibria, or amplitude rescaling. Blue denotes GT, orange denotes predictions, and line opacity is 0.35.



(a) Ground truth (b) Transformer

Figure 13: **Lorenz atlas for mixed-region training.** Twenty parameters are shown in a 5×4 layout. Gray denotes the training support $0 < \rho < 24.0579$, while white denotes unseen parameters. Both panels use the same parameter values, initial states, time windows, and axes. The Transformer result is from the frozen seed-42 model after 10k updates.



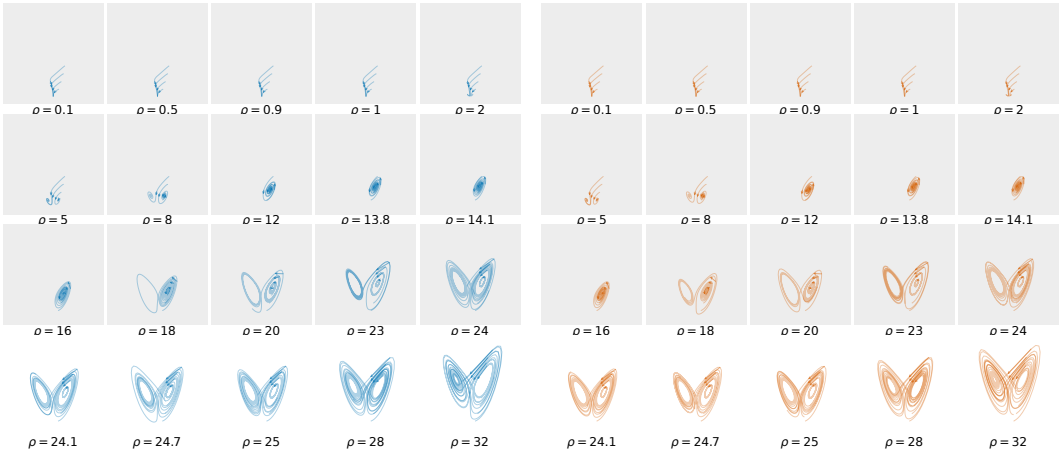
(a) Ground truth (b) Transformer

Figure 14: **Lorenz atlas for transient-region training.** Twenty parameters are shown in the same 5×4 layout. Gray denotes the training support $13.9265 < \rho < 24.0579$, while white denotes unseen parameters. Both panels use the same parameter values, initial states, time windows, and axes. The Transformer result is from the frozen final 10k model trained only on I_2.

I Lyapunov Exponents of the Learned Lorenz Flow

The flow comparisons of Section 4.3 report marginal distributions, fixed-state agreement and switching. None of those quantities certifies chaos, and the paper's own detector vocabulary treats an unresolved label as unresolved rather than as chaos. We therefore add a direct dynamical quantity: the largest Lyapunov exponent of the frozen model and of the reference, obtained by tangent-vector propagation of the one-step map with renormalization after every step ($\Delta t = 0.01$, 3,000 steps after 2,000 burn-in steps, float64).

The reference propagates the RK4 step map of the Lorenz system with a central-difference Jacobian; the model propagates the increment map $X \mapsto X + \text{scale} \cdot \text{factor} \cdot m_\theta(p_{\text{norm}}, (X - \text{center})/\text{scale})$ with the Jacobian taken by autograd, using the training-only normalization recorded with the



(a) Ground truth (b) Transformer

Figure 15: **Lorenz phase portraits across trained and unseen parameter regimes.** Each 5×4 atlas overlays four common initial states at each displayed ρ, retaining $t \in [0, 5.12]$ without burn-in. Curves show the x–z projection; arrows indicate motion. Blue: GT; orange: frozen parameter-token Transformer, seed 42, final 10k checkpoint. Gray denotes the training support $0 < \rho < 24.0579$; white denotes unseen parameters. All cells and both atlases share the same physical scale, with no trajectory normalization. These are test trajectories at new parameter/initial-state combinations. The equal-duration portraits show transients; the separate 100-unit tests quantify sustained double-wing motion.

checkpoint. As a self-check, the estimator returns $\lambda = 0.9046$ at the classical parameter set $(\rho, \sigma) = (28, 10)$, against the literature value 0.906.

Table 12: **Largest Lyapunov exponent** of the reference Lorenz flow and of the frozen 2,048-trajectory, width-128 model of Appendix K. Both columns use the same estimator. Positive values indicate exponential divergence on the attractor; negative values correspond to fixed-point or periodic destinations.

(ρ, σ)	Reference λ	Model λ
28, 10	0.9046	1.0404
32, 10	0.9431	1.0257
30, 10	0.9919	0.8687
26, 10	0.9001	0.8766
28, 9.375	0.8916	0.8496
28, 11.25	0.8111	0.9837
35, 12	0.9750	1.1131
45, 15	1.2560	1.2582
25, 6	−0.1062	−0.1683
20, 8	−0.1596	−0.0110
Mean	0.7408	0.7837

The model and the reference agree on the sign of the exponent at all ten parameter pairs and on its order of magnitude, with individual discrepancies up to a few tenths. The two parameters inside the fixed-state region are correctly predicted to be non-chaotic. This supports the use of "chaos" for the flow extension in the sense that the frozen model, not only the reference, has a positive exponential rate at the conventional chaotic parameters, while leaving open whether its chaotic attractor has the same geometry or dimension as the reference.

J Generalized Hopf: Training Support and Basin Tests

System and natural boundaries. In Cartesian coordinates, the system is

$$\dot{x} = (\mu_1 + \mu_2 r^2 - r^4)x - y, \quad \dot{y} = x + (\mu_1 + \mu_2 r^2 - r^4)y, \quad r^2 = x^2 + y^2. \tag{23}$$

The origin changes stability at $\mu_1 = 0$. Nonzero periodic orbits have squared radii $s_\pm = (\mu_2 \pm \sqrt{\mu_2^2 + 4\mu_1})/2$ when real and positive. For $\mu_2 > 0$ and $-\mu_2^2/4 < \mu_1 < 0$, the origin and outer cycle

are stable, separated by the unstable inner cycle; the cycles meet at $\mu_1 = -\mu_2^2/4$ (Kuznetsov, 1998). These boundaries define the training region before model evaluation. The rectangular sampling limits fix numerical scales and are not additional bifurcations.

Data and model. Parameters are sampled uniformly in area by rejection from $[-1, 0] \times [-1, 1]$, retaining $\mu_1 < -\max(\mu_2, 0)^2/4$. Initial states are uniform in a disk of radius $\sqrt{(1+\sqrt{5})/2}$, the largest stable-cycle radius in the declared evaluation box $[-1, 1]^2$. All 512 training trajectories have strictly decreasing radii; 128 additional trajectories provide independent validation. Each has 256 transitions at $\Delta t = 0.05$, spanning 12.8 time units. The oracle integrates the radial equation using ten RK4 substeps per observation and rotates the angle exactly. Against Cartesian DOP853 integration with tolerances $10^{-12}, 10^{-14}$, the largest one-step discrepancy in 24 independent checks was 2.60×10^{-9}.

The two-layer Transformer has width 64, four attention heads, and feed-forward width 256. A linear two-dimensional projection embeds the parameters into one token, separately from the Cartesian state token. Training-only state, parameter, and increment statistics define normalization. The incremental target, AdamW schedule, batch size 128, and 10k-update budget match the Lorenz setup. The model seed is 42 and data seed is 20260917. The final checkpoint is fixed without OOD selection; its held-out normalized one-step MSE is 2.2243×10^{-6}.

Long-time basin tests. The nine parameter pairs are $(-0.5, -0.5)$, $(-0.5, 1)$, $(-0.1, 1)$, $(-0.2, 1)$, $(-0.04, 0.5)$, $(0.1, -1)$, $(0.1, 0)$, $(0.1, 1)$, and $(0.5, 1)$. Each uses radii $0.1, 1.2$ and phases $0, \pi/2$. All 36 rollouts remain finite through $t = 300$. Over $t \in [150, 300]$, the predicted origin/cycle outcomes match the reference at all nine pairs: 14 trajectories approach points and 22 approach cycles. The operational point threshold is mean radius below 0.03; cycle labels require mean angular speed within 0.1 of one and radius standard deviation below 0.05. Continuous radius and frequency measurements are retained. These are tests of multiple initial conditions, not a quasistatic hysteresis experiment.

Parameter slices and boundary displacement. Two slices fix $\mu_2 = -1$ or 1 and evaluate 41 equally spaced μ_1 values from -0.5 to 0.5, each from radii $0.1, 1.2$ at phase zero. All 164 predictions remain finite. Mean absolute radius errors against matched reference windows $t \in [200, 400]$ are 0.00894 and 0.04975, respectively. At $\mu_2 = 1$, the large initial state incorrectly collapses at $\mu_1 = -0.225$, while small initial states incorrectly approach cycles at $\mu_1 = -0.05, -0.025$. Independent fixed-point solves and Jacobians of the learned discrete map locate complex-eigenvalue unit-circle crossings at $\mu_1 = 0.00720, -0.00652, -0.02872, -0.05615$ for $\mu_2 = -1, 0, 0.5, 1$; the reference boundary is zero throughout. Thus the model recovers coexistence but displaces its boundaries. Full unstable-cycle continuation and a dense two-dimensional asymptotic regime classification are not claimed.

Main-figure provenance. Figure 6 uses the 64 cell centers of a uniform 8×8 grid over $[-0.5, 0.5] \times [-1, 1]$. All four initial states are shown at each center for $t \in [0, 12.8]$, without burn-in. The glyph transformation is fixed globally, preserving relative amplitudes and the Cartesian aspect ratio. Arrows indicate actual displacements from $t = 1$ to 1.4 on the two outer-start trajectories. Slow convergence near a bifurcation can extend beyond this short display window. Gray marks the exact curved training-support intersection, not a classification inferred from trajectory appearance.

K Joint Lorenz Parameters: Fixed-Region Training and Regime Recovery

The joint experiment varies both ρ and σ at $\beta = 8/3$ to test periodic extrapolation and recovery of a parameter-dependent regime map. It complements the fixed-σ geometry tests in Appendix H, using a separately defined training population and evaluation protocol.

Numerically defined training support. We draw 131,072 scrambled Sobol proposals over $[1, 409] \times [0.5, 60]$ and integrate each from $(x, y, z) = (0.001, 0.001, \rho - 1)$ to $t = 100$, with RK4 step 0.00125. A proposal is retained when every coordinate has standard deviation below 10^{-3}

Table 13: Joint Lorenz evaluation on a 48×48 parameter grid after training on numerically certified fixed-point parameters. Entries count matching labels against the saved reference; denominators exclude the critical $\rho = 1$ line. Finite counts include all cells. All arms use seed 42 and 20,000 updates.

N	Width	ID MSE	Fixed/268	P2/358	P4/139	P8/22	Finite/2304
2048	64	1.56e-06	249	1	2	0	2155
2048	128	1.05e-06	266	9	3	0	2225
8192	64	1.74e-06	227	149	6	0	1945
8192	128	5.96e-07	259	2	5	0	1173

over the final two time units and its endpoint is within 0.01 of a nonzero analytic equilibrium. This selects 16,285 parameters by their observed fixed-point destination, without an analytic stability prefilter. The canonical initial condition is shared with the evaluation atlas; training transients are retained and can include cross-wing motion.

A seeded random permutation supplies 8,192 training parameters and 1,024 disjoint validation parameters; the 2,048-trajectory condition uses a nested prefix. Sampling is uniform in parameter area conditional on the numerical fixed-point criterion. All 268 noncritical GT-fixed coarse cells contain training parameters in the larger condition. Each trajectory contributes 512 transitions at $\Delta t = 0.01$, spanning 5.12 time units. The smaller and larger datasets therefore contain 1,048,576 and 4,194,304 correlated transition pairs, respectively.

Matched updates and full-grid evaluation. The two-layer, four-head Transformer uses width 64 or 128 (131,840 or 525,824 parameters), a two-dimensional parameter token, a three-dimensional state token, and increment prediction with training-only normalization. All four conditions use seed 42, 20,000 AdamW updates, and batch size 512. This fixes transition draws at 10.24 million per condition, corresponding to approximately 9.77 and 2.44 passes through the two datasets. The final checkpoint is used throughout; this is a matched-update comparison, with different effective epoch counts.

We evaluate every cell of the common 48×48 grid for 100 time units, discarding the first 50 for section recurrence. The 48 critical cells at $\rho = 1$ remain in the finite-rollout denominator but are excluded from class scores. Reference outcomes define fixed-region membership; all nonfixed reference classes are OOD. Divergent or unresolved model outcomes remain in the scored population.

Section calibration. Upward crossings of $z = \rho - 1$ use degree-seven Lagrange interpolation through eight observations and 24 bisection steps. Signed (x, y) return points are tested for minimal recurrence lag from 1 to 32, requiring at least 40 crossings and four cycles. The 95th-percentile recurrence-distance threshold of 0.1 is the smallest tested value recovering all 358 P2 and 139 P4 reference cells at model observation resolution. Reference P8 recovery is 19/22; all 22 remain in the primary denominator. Calibration uses a finer reference observer at $\Delta t = 0.0025$. An aperiodic label records absence of accepted recurrence; chaos is certified separately by the Lyapunov measurement of Appendix I.

Data, capacity, and autonomous structure. Table 13 separates fixed-region agreement, periodic-regime agreement, and finite-rollout coverage. Increasing data raises P2 agreement from 1 to 149 cells at width 64, but none of the four conditions recovers the full P2/P4/P8 structure. Width 128 attains lower within-condition validation MSE while providing less reliable full-grid stability in the larger-data condition. These outcomes distinguish fitting the training transitions from recovering the joint regime map; they do not identify a unique cause of the extrapolation errors. The single model seed per condition supports these controlled comparisons, without estimating variability over training seeds.

The archived numerical audit also compares 80 saved local transitions against independent DOP853 solves, with maximum absolute discrepancy 2.46×10^{-5}. A sensitive long transient can separate pointwise under different integrators despite a matching eventual fixed-point destination. The local-target check and the finite-time destination criterion are therefore recorded separately.

K.1 Conventional-Parameter Boundary and Geometry

The main flow illustration uses the final 20k checkpoint of the 2,048-trajectory, width-128 condition. After inspecting the original domain, we evaluate the complete 25×25 grid $\rho \in [10, 50]$, $\sigma \in [5, 20]$, including the conventional vicinity of $(28, 10)$. The parameters vary jointly; the training data, normalization, and model are unchanged, and the initial state is $(0.001, 0.001, \rho - 1)$ throughout. Both this model and the 8,192-trajectory, width-128 condition are evaluated on the same grid, and the former is selected for the main display because its long-time geometry is closer to the reference. The display box and the model were chosen after inspecting the wider-domain outcomes, so the figure is an existence illustration rather than a held-out region benchmark.

Coarse evaluation. Reference integration uses float64 RK4 with step $0.0025/3$, observed every 0.0025; the model advances at 0.01 in float32. We retain the windows $[50, 100]$ and $[150, 200]$ and sample both tails every 0.02 for distributional comparisons. A cell is labeled fixed when all coordinate standard deviations over the final two time units fall below 10^{-3}. All 625 cells are retained; no cell is removed, smoothed, or interpolated. Short observations in the original training data can contain oscillations before convergence, and no long-time nonfixed training attractors are supplied.

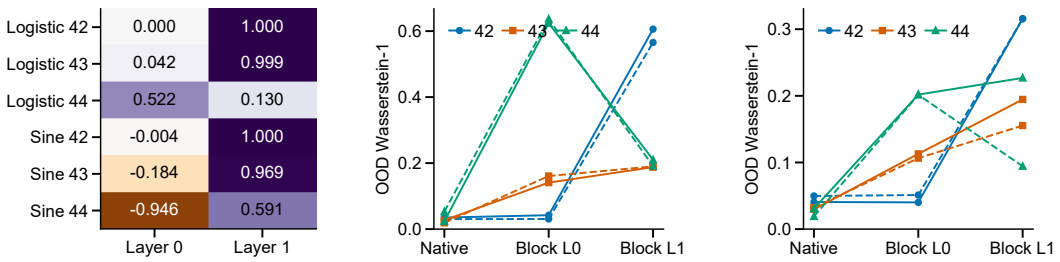
Adaptive boundary refinement. Every σ row of the coarse grid contains exactly one fixed-to-nonfixed transition, so the transition is bracketed and then refined locally instead of resampling a global fine grid. Each round places nine interior samples inside the current bracket and keeps the adjacent fixed/nonfixed pair, contracting the bracket tenfold. Three rounds reduce the maximum bracket width to 3.7×10^{-3} in ρ; the reported boundary is the bracket midpoint. The two refined boundaries differ by a median absolute displacement of 0.95 in ρ and a mean of 1.55, with a maximum of 10.7 at $\sigma = 5$, the steepest part of the curve. These are finite-time outcomes from the specified initial state rather than analytically certified bifurcation points.

Boundary and distribution comparisons. Both models remain finite at all 625 cells through $t = 200$. The displayed model agrees with the reference fixed/nonfixed classification in 593/625 cells and reproduces repeated wing switching, at least three x-sign changes per window, in 362/375 reference cells that switch. Its label is unchanged between the two windows in 600/625 cells. Marginal W_1 is computed from the equally spaced physical tails, normalized coordinatewise by the common 8,192-trajectory, width-64 training standard deviations, then averaged across coordinates. Among the reference nonfixed cells the displayed model reaches mean/median W_1 of $0.08947/0.06861$ and mean normalized standard-deviation error 0.02270. The 8,192-trajectory, width-128 model has better fixed/nonfixed agreement (600/625) and switching coverage (369/375) but larger distributional error ($0.14725/0.14473$) and amplitude error (0.04183); both complete comparisons are retained.

Figure rendering. Figure 5 uses a regular 5×5 grid of interior cell centers, $\rho \in \{14, 22, 30, 38, 46\}$ and $\sigma \in \{6.5, 9.5, 12.5, 15.5, 18.5\}$, chosen independently of the outcomes. Two initial states are released at every pair, $(18, 18, \rho - 1 + 6)$ and $(-18, -18, \rho - 1 - 6)$, identical in both panels; the second is deliberately far from the equilibrium so that fixed-state transients remain visible. Trajectories are drawn from $t = 0$ for 20 time units, which covers the visual settling of the fixed-state cells. The view is an axonometric rotation of the (x, y, z) state, azimuth 35° and elevation 22°, so that the convergence spirals and the two wings are legible in the same projection. Every glyph is translated so that its own bounding-box center sits on the parameter pair, which makes the two panels directly comparable cell by cell but removes absolute state offsets within a cell. All glyphs share one physical scale, ρ displacement 0.0799 per state unit, with no per-cell normalization, so amplitudes remain comparable across the panel.

L Interventions and Controls

Single-step experiments use 384 ID and 384 OOD parameters per trained Transformer. For each, a 64-state history is generated under the recipient parameter from a random initial condition. The donor parameter is $r' = \min(r + 0.25, 4)$ for logistic or $r' = \min(r + 0.08, 1)$ for sine. We keep the recipient history unchanged for the donor pass, so the donor is an explicit counterfactual input rather



(a) Donor-effect transfer (b) Logistic rollouts (c) Sine rollouts

Figure 16: **Token-level parameter interventions alter closed-loop dynamics.** (a) OOD donor-error reduction after patching final-query attention outputs, for all six Transformers; negative values mean increased donor error. (b,c) OOD W_1 after blocking parameter-to-state attention at every step. Lines connect conditions for the same seed; solid/dashed lines indicate one/four state tokens. All 36 conditions share the 129-point grid and trajectory budget.

Table 14: OOD donor-error reduction for complete attention-output patches and equal-norm random controls. Sham reduction is zero throughout. Each entry uses 384 OOD histories from one trained model.

Family	Seed	L0 donor	L1 donor	L0 random	L1 random
Logistic	42	0.000151	1.000000	0.000022	-40.688593
Logistic	43	0.042383	0.999443	-0.005526	-0.444243
Logistic	44	0.522322	0.129718	-2.708184	-0.225740
Sine	42	-0.003656	0.999995	-0.000583	-0.853852
Sine	43	-0.184315	0.969431	-0.161228	-0.314544
Sine	44	-0.945527	0.591043	-0.155559	-0.433707

than a newly generated trajectory. The same physical parameter/history samples are used across seeds within a family.

At each layer, we save the entire final-query attention output before its residual addition. Donor replacement, sham replacement, and a random direction matched to the donor–recipient vector norm are run separately. Table 14 lists the OOD donor-error reductions. Sham replacement has zero effect for every layer and seed. Negative random-control values indicate increased discrepancy from the donor output, rather than evidence of accurate physical prediction.

The layer at which parameter mediation occurs varies across seeds. In logistic seed 44, the first-layer patch yields $R_0 = 0.522$ and the second yields $R_1 = 0.130$. Sine seed 44 yields $R_1 = 0.591$, whereas seeds 42 and 43 yield approximately 1.000 and 0.969. These results support learned, seed-dependent routes of parameter influence. A full attention-output replacement tests mediation at the layer-output level; it does not identify a unique head or feature circuit.

For blocking, all state-query edges to the parameter key are masked in one layer. A single-step rescue restores the recipient attention output at the final query. At the final layer this exactly restores the downstream computation by construction, and is an implementation check. At the first layer, other history positions remain altered, so a final-query rescue need not restore the full later computation. These controls prevent interpreting an algebraic restoration as independent evidence of a unique circuit.

Closed-loop blocking uses all six Transformers, $C \in \{1, 4\}$, and native/layer-0-blocked/layer-1-blocked conditions, giving 36 runs on a shared 129-point grid with three initial states, 1,024 discarded steps, and 512 retained steps. Table 15 reports all conditions grouped by model and context. The cascade counts are conservative coarse-grid counts and are not comparable to the adaptive seven-flip result as if they had the same detection power.

Blocking also separates finite-grid periodic structure from fidelity: some blocked four-state models retain two detected flips while their distributions change markedly, so a count of detected transitions carries less information than their parameter values and associated trajectories. The 129-point intervention scan is used for matched causal comparisons; the seven-flip result comes from the separate refined scalar analysis.

Table 15: All closed-loop intervention conditions. Each row contains the three W_1 conditions for one model/context. Counts give native/L0-blocked/L1-blocked consecutive flips on the shared 129-point grid.

Family	Seed	C	Native W_1	Block L0	Block L1	Flip counts
Logistic	42	1	0.034967	0.042183	0.606053	2/2/0
Logistic	42	4	0.030323	0.030537	0.565538	2/0/0
Logistic	43	1	0.027007	0.140915	0.187707	2/0/1
Logistic	43	4	0.018944	0.160885	0.190042	2/0/1
Logistic	44	1	0.023330	0.623765	0.210141	2/1/1
Logistic	44	4	0.052811	0.636572	0.190833	2/0/2
Sine	42	1	0.040701	0.040120	0.315712	2/2/0
Sine	42	4	0.049765	0.051119	0.315712	2/2/0
Sine	43	1	0.029687	0.112884	0.194603	2/2/0
Sine	43	4	0.032185	0.106443	0.155418	2/2/0
Sine	44	1	0.030832	0.201860	0.226855	2/1/1
Sine	44	4	0.019002	0.201491	0.094361	2/0/2

The parameter route is necessary during training. The interventions above perturb a trained model. To test whether the parameter route is needed to build the structure at all, we train the same architecture under the locked protocol with the parameter label permuted across each batch: every trajectory keeps its dynamics but is labeled by a parameter drawn independently of it, so the marginal distribution of labels is unchanged while their pairing with the observed states is destroyed.

Table 16: **Trained with permuted parameter labels.** Three seeds, identical protocol and evaluation grid as the main comparison. Periodic agreement is over reference-periodic parameter–initial-state pairs; the parameter derivative is measured on the out-of-distribution slice, where the reference mean $|\partial_r F|$ is 0.173.

Seed	ID W_1	OOD W_1	Periodic agreement	Coarse flips
42	0.0417	0.1914	0.0%	0
43	0.0394	0.1912	0.0%	0
44	0.0421	0.1921	0.0%	0
Aligned baseline (mean of 3)	0.0009	0.0278	78.5%	3

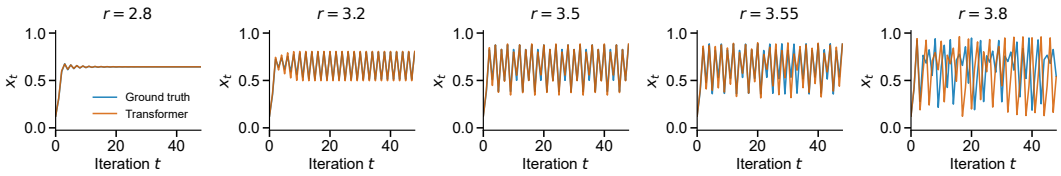
The cascade disappears completely, and the learned map recovers no systematic parameter dependence: its mean parameter-derivative mismatch is 0.164 against a reference scale of 0.173, which is as large as the variation the reference family itself carries, and its state critical exponent remains quadratic ($z = 2.03$). Parameter information therefore has to reach the state token for the extrapolated structure to exist, whereas blocking it after training only degrades a structure that training had already built.

M Artifact Integrity and Availability

The experiment archive contains raw rollout arrays, per-parameter diagnostics, checkpoints, configuration records, adaptive-stage tails, and root-verification results. Fixed-grid inputs and reference arrays are identical across compared models. The scalar integrity record covers 12 checkpoints, 30 main evaluations, and 36 closed-loop intervention conditions. Additional flow checks recompute the four joint Lorenz class-count tables, validate the saved training/validation membership, recover all 36 generalized Hopf outcomes, and recompute the mixed-Lorenz switching and marginal distributional distances from saved arrays. The selected 25×25 Lorenz confirmation retains all cells and checks representative labels across windows, initial states, model precision, and independent reference integration. Unit tests cover numerical routines and detect missing or changed figure provenance records.

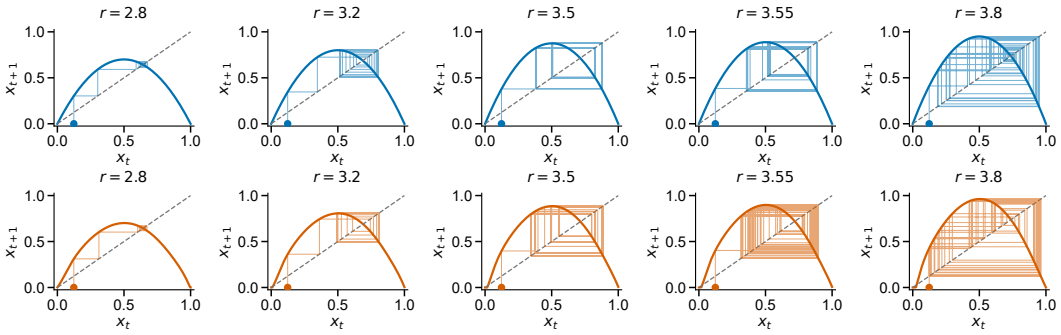
N Matched Trajectories and Cobwebs Across Training

Illustrative chaos and a shifted periodic window. Figure 17 and Figure 18 use logistic $r = 3.8$ to illustrate chaotic behavior in both maps at the 10,000-update checkpoint. After 2,048 burn-in steps, 2,048 further iterates give Lyapunov estimates of 0.422 for ground truth and 0.431 for the Transformer. The model derivative uses a centered finite difference through the clamped scalar map.



(a) Stable (b) Period 2 (c) Period 4 (d) Period 8 (e) Chaotic

Figure 17: **Matched trajectories across regimes.** Logistic ground truth and the one-state Transformer after 10,000 updates share seed 42, identical parameters, and the same initial state. Columns progress from a fixed point through period-2, period-4, and period-8 reference regimes to a chaotic reference. Each panel shows 48 autoregressive updates including transients. Blue marks ground truth and orange the Transformer. The chaotic illustration uses $r = 3.8$, where both maps have positive finite-orbit Lyapunov estimates; the appendix also shows the final 20k model's period-3 window at $r = 3.9$. Appendix N repeats these views across both systems and training stages.



(a) Stable (b) Period 2 (c) Period 4 (d) Period 8 (e) Chaotic

Figure 18: **Reference and learned cobwebs for the same regimes.** Each column stacks the ground-truth cobweb above the Transformer cobweb at the matching parameter after 10,000 updates. Solid curves show the corresponding scalar map and dashed lines mark the identity. Blue marks ground truth and orange the Transformer. Repeated iteration turns the learned transition into stable, periodic, or irregular motion. The chaotic illustration uses $r = 3.8$, where both maps have positive finite-orbit Lyapunov estimates; the appendix also shows the final 20k model's period-3 window at $r = 3.9$.

The atlas keeps $r = 3.9$ throughout training: there, the final model converges to period 3 (maximum lag-3 residual below 10^{-9} over the last 64 states) with a Lyapunov estimate of -0.099, although the reference is chaotic. This contrast exposes a displaced periodic window rather than presenting every chaotic-reference parameter as successful chaotic reproduction. These illustrative parameters do not select a checkpoint or alter the full-grid metrics. These views connect the transition function to its repeated application. All panels use seed 42, one state token ($C = 1$), and the same initial state $x_0 = 0.123456789$. Blue denotes ground truth and orange the Transformer. A cobweb alternates a vertical move to the map and a horizontal move to the identity line; repeated moves show convergence or recurrent motion. Solid curves show the reference or learned map and dashed gray lines show the identity. The learned map includes the same $[0, 1]$ output clamp as autonomous evaluation. We compute it directly from each checkpoint on 1,001 state values. No map is fitted to the plotted trajectories. For each system we fix five physical parameters: logistic $r = 2.8, 3.2, 3.5, 3.55, 3.9$ and sine $r = 0.6, 0.8, 0.845, 0.861, 0.99$. Column titles name the *reference* regime. Training covers the stable-reference parameters, including transient trajectories; the periodic and chaotic reference parameters are outside the training interval. We retain all 128 iterates from initialization in the source data. Plots display the first 48 updates without burn-in. Float64 evaluation resolves the scalar geometry; primary aggregate fidelity remains the original float32 evaluation. We show nine stages for both systems: initialization, 500, 1,000, 2,000, 3,000, 4,000, 6,000, 10,000, and 20,000 updates. Trajectory panels are grouped by system, then cobweb panels. Within each group a single caption closes the figure; row labels mark the stage and column titles mark the regime.

O VISUAL ATLAS

The atlas pairs ground truth with every reported model, seed, and Transformer context. Each displayed panel is a separate PDF with an editable SVG and 600-dpi PNG counterpart; the document assembles panels using LaTeX subcaptions. The full-range plots use the original evaluation arrays and share physical axes, 512 state bins, and logarithmic counts from 1 to 1,536. Every retained observation contributes. Occupied bins use fixed 0.6-point square marks for final-size visibility, with color encoding the original count; no bins are smoothed or omitted. Training parameter intervals are shaded gray. State context C excludes the parameter token; $C = 255$ is a length-extrapolation stress test.

O.1 DETAILED CASCADE GEOMETRY

The local cascade views use a separate 1,025-point uniform parameter grid, three initial states $(0.123456789, 0.371239, 0.817321)$, 8,192 burn-in steps, and 512 retained steps. Reference and seed-42 one-state Transformer rollouts use float64; trained weights are unchanged. The logistic window is $[3.35, 3.62]$ and the sine window is $[0.80, 0.92]$. These magnifications reveal local shape and displacement and do not replace the fixed-grid accuracy measurements or the independent flip-verification procedure.

O.2 PROGRESSIVE MAGNIFICATION OF VERIFIED FLIPS

The following views use the saved blind adaptive trajectories and numerically verified roots. At duplicate parameter values, the longest available burn-in is retained. Each view spans $r_n \pm 0.16(r_n - r_{n-1})$ and magnifies the orbit branch nearest $x = 1/2$. The parent phase nearest $x = 1/2$ is isolated using 45% of the distance to its nearest neighboring phase just below the root. The vertical crop includes the observed branch and its daughters across sampled parameters, with a 10% margin on each side; all observations within the crop are plotted. Dots occupy only evaluated parameter values: gaps are not interpolated. Dashed lines mark verified roots. Physical coordinates are retained, so these panels expose both repeated local splitting and displacement. The source manifest records exact crops, sample counts, and burn-in budgets. Sine panels here calibrate the reference map; the verified neural high-order panels concern logistic seed 42.

O.3 ALL TRANSFORMER SEEDS AND CONTEXTS

Each system is shown once. Columns are ground truth and the three training seeds; rows are Transformer contexts $C = 1, 4, 255$. Panels use the compact density style of Figure 3.

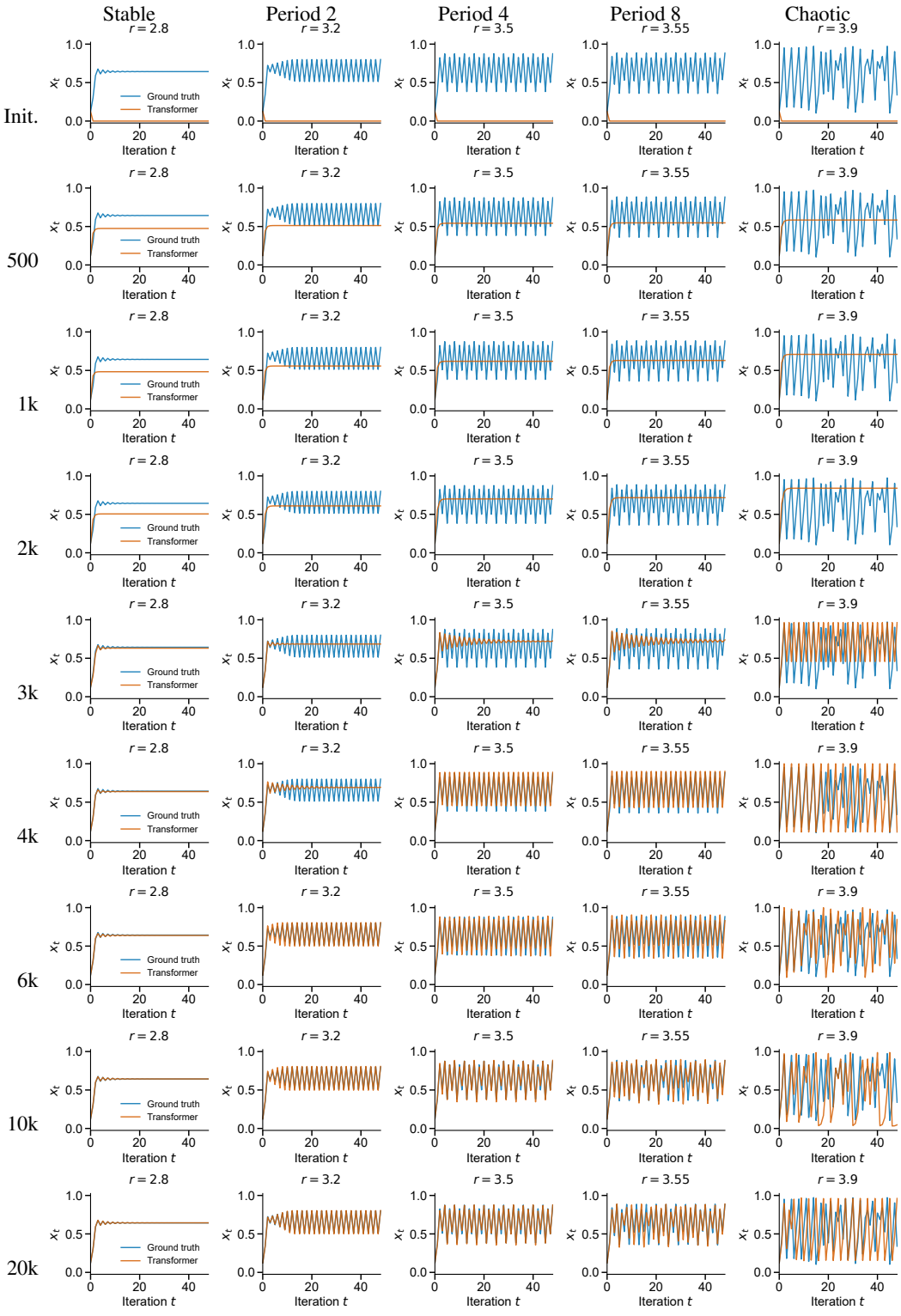


Figure 19: **Logistic trajectories across training.** Each row is one checkpoint; columns are the same five reference regimes. Blue denotes ground truth and orange the Transformer, with the legend only in the top-left panel. Regime names describe the reference, not a classification inferred for the checkpoint.

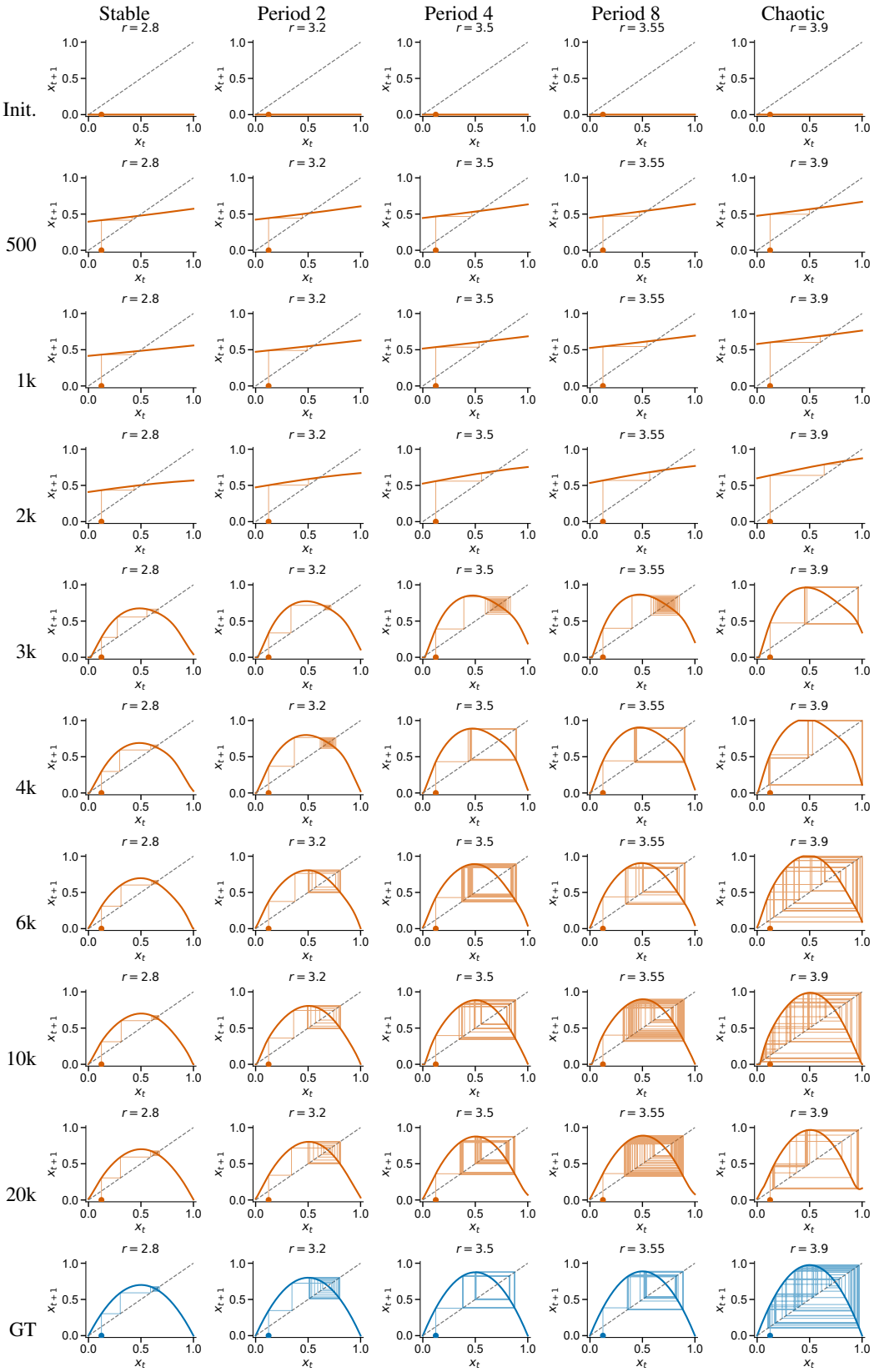


Figure 20: **Logistic cobwebs across training.** Each row shows the Transformer at the labeled stage; the final row is ground truth (GT). Columns keep the same reference regimes. Blue marks ground truth and orange the Transformer.

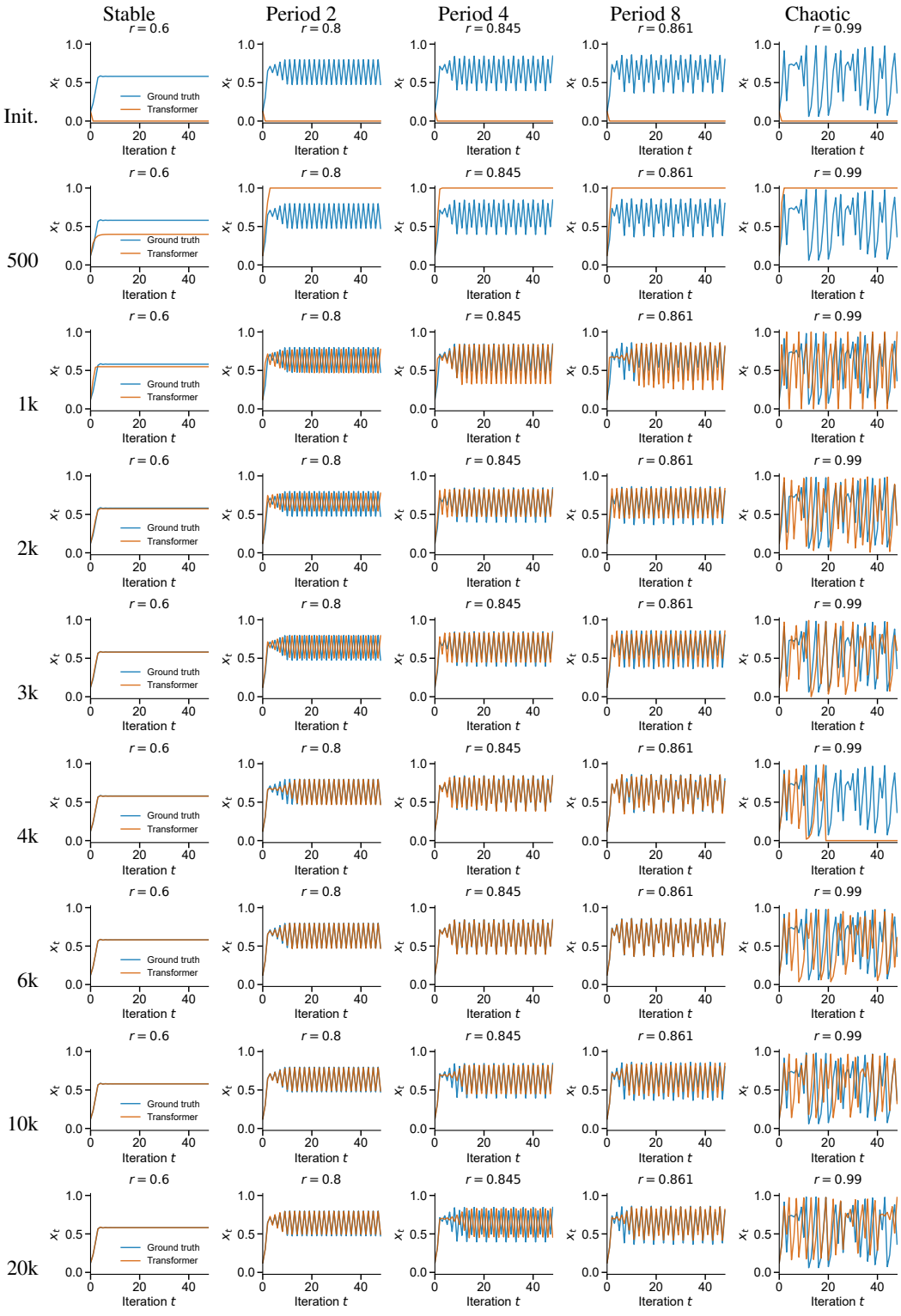


Figure 21: **Sine trajectories across training.** Each row is one checkpoint; columns are the same five reference regimes. Blue denotes ground truth and orange the Transformer, with the legend only in the top-left panel. Regime names describe the reference, not a classification inferred for the checkpoint.

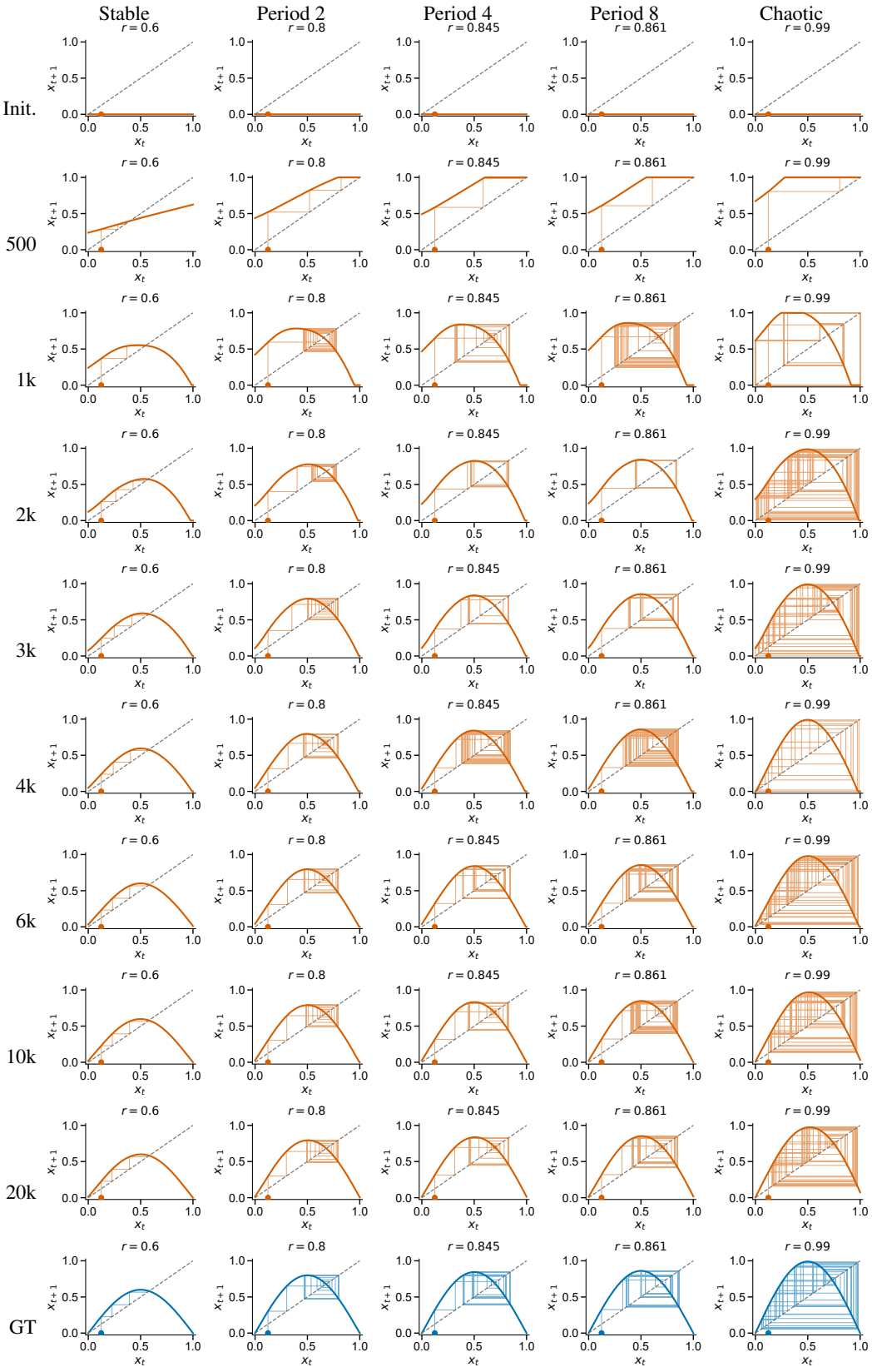
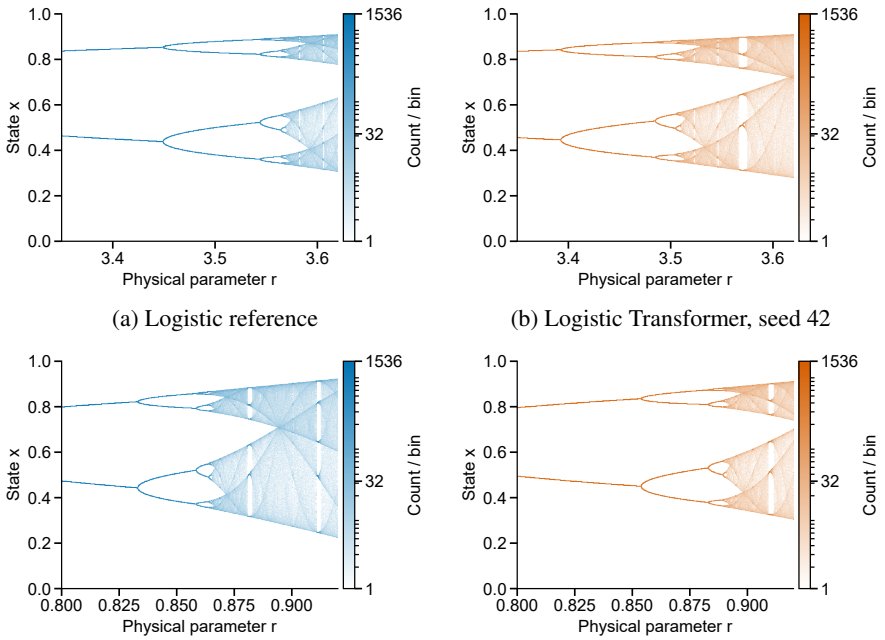


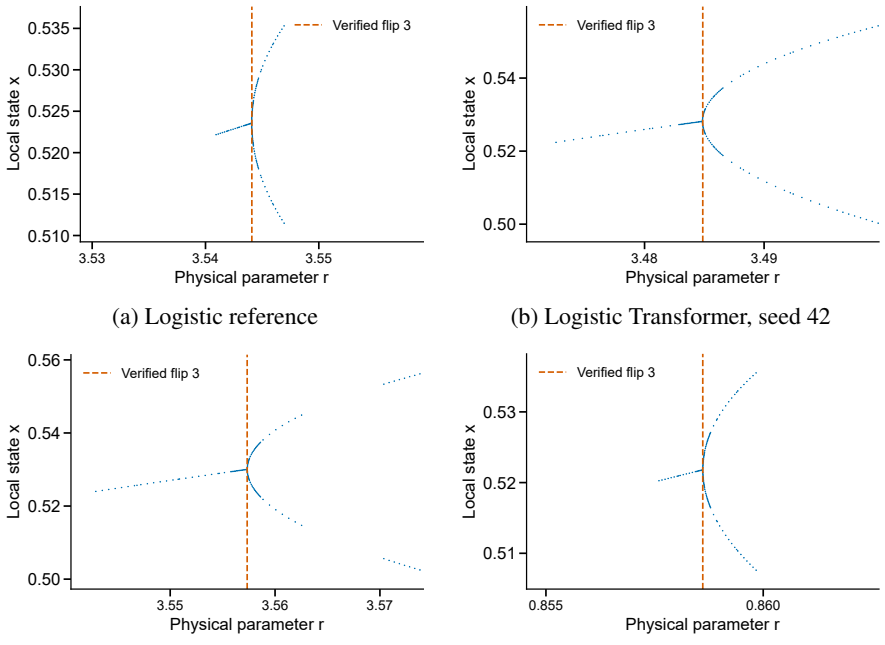
Figure 22: **Sine cobwebs across training.** Each row shows the Transformer at the labeled stage; the final row is ground truth. Columns keep the same reference regimes. Blue marks ground truth and orange the Transformer.



(a) Logistic reference (b) Logistic Transformer, seed 42

(c) Sine reference (d) Sine Transformer, seed 42

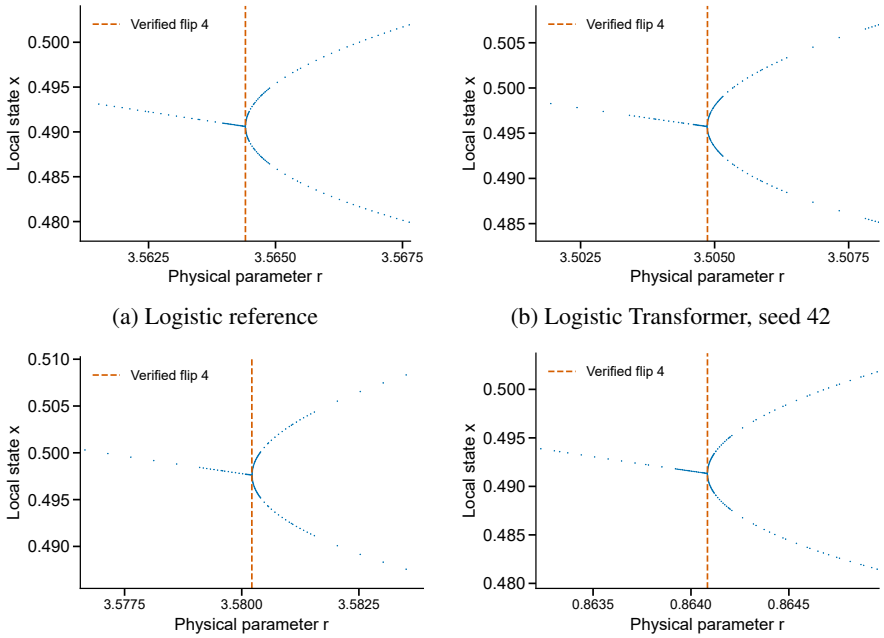
Figure 23: **Dense local views resolve the cascade shape.** Reference and Transformer panels share the same physical parameter window within each system. All $1{,}025 \times 3 \times 512$ retained samples enter the histograms. The parameter positions are not aligned or rescaled between reference and learned maps.



(a) Logistic reference (b) Logistic Transformer, seed 42

(c) Logistic MLP, seed 42 (d) Sine reference

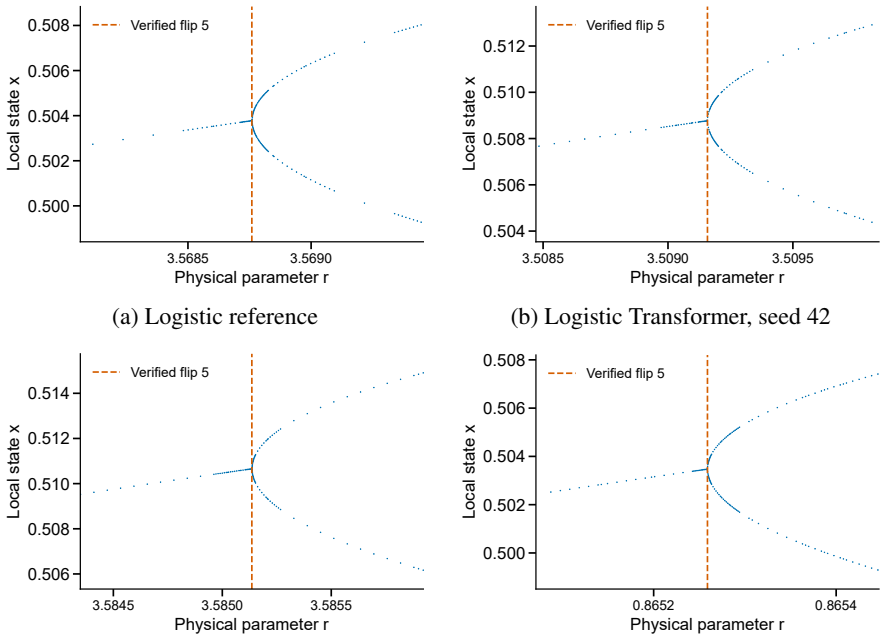
Figure 24: **Local geometry at doubling 3.** The verified parent period is 4 and the emerging doubled period is 8. Each panel uses its own physical root and explicitly labeled local state range. Dense clouds near a root can include slow transients; root verification additionally uses cycle and multiplier checks.



(a) Logistic reference (b) Logistic Transformer, seed 42

(c) Logistic MLP, seed 42 (d) Sine reference

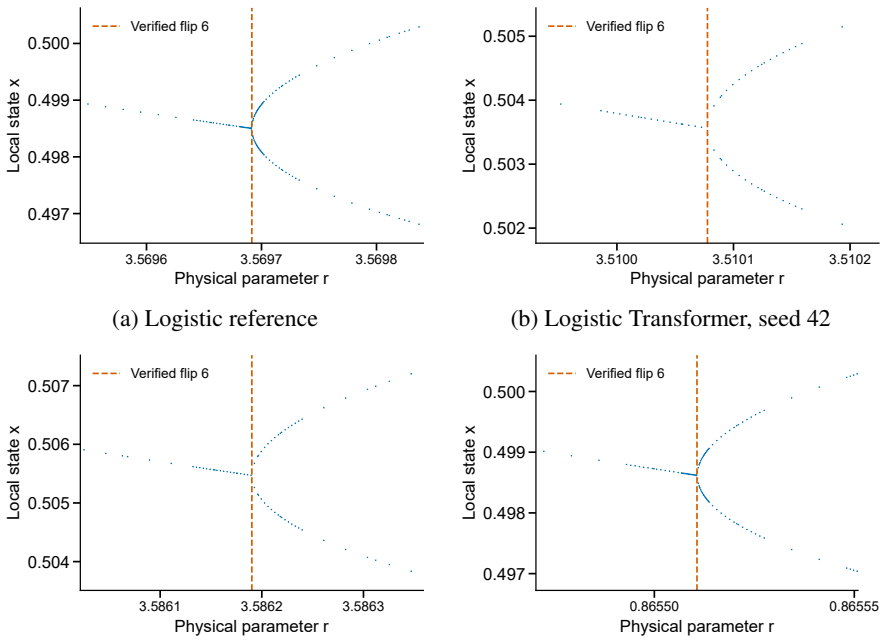
Figure 25: **Local geometry at doubling 4.** The verified parent period is 8 and the emerging doubled period is 16. Each panel uses its own physical root and explicitly labeled local state range. Dense clouds near a root can include slow transients; root verification additionally uses cycle and multiplier checks.



(a) Logistic reference (b) Logistic Transformer, seed 42

(c) Logistic MLP, seed 42 (d) Sine reference

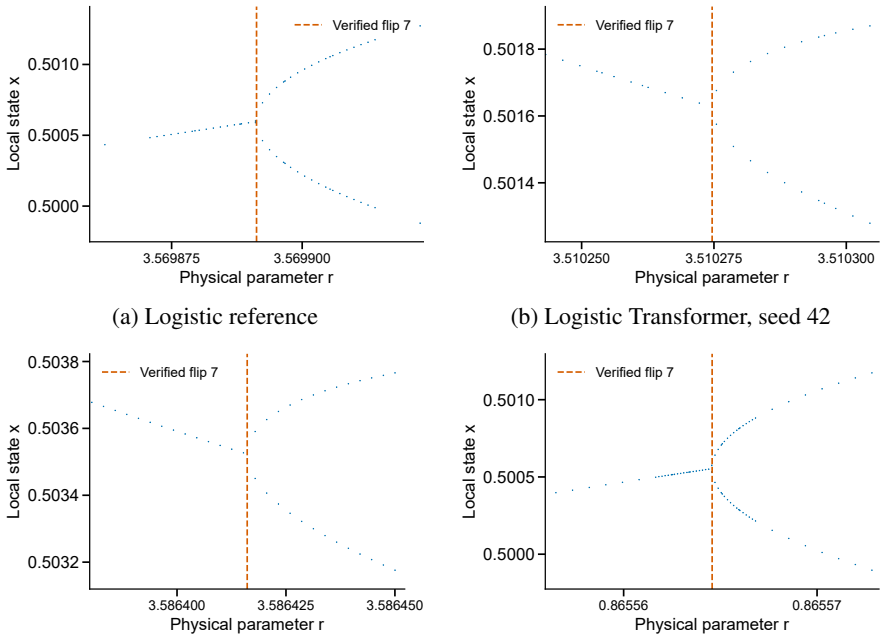
Figure 26: **Local geometry at doubling 5.** The verified parent period is 16 and the emerging doubled period is 32. Each panel uses its own physical root and explicitly labeled local state range. Dense clouds near a root can include slow transients; root verification additionally uses cycle and multiplier checks.



(a) Logistic reference (b) Logistic Transformer, seed 42

(c) Logistic MLP, seed 42 (d) Sine reference

Figure 27: **Local geometry at doubling 6.** The verified parent period is 32 and the emerging doubled period is 64. Each panel uses its own physical root and explicitly labeled local state range. Dense clouds near a root can include slow transients; root verification additionally uses cycle and multiplier checks.



(a) Logistic reference (b) Logistic Transformer, seed 42

(c) Logistic MLP, seed 42 (d) Sine reference

Figure 28: **Local geometry at doubling 7.** The verified parent period is 64 and the emerging doubled period is 128. Each panel uses its own physical root and explicitly labeled local state range. Dense clouds near a root can include slow transients; root verification additionally uses cycle and multiplier checks.

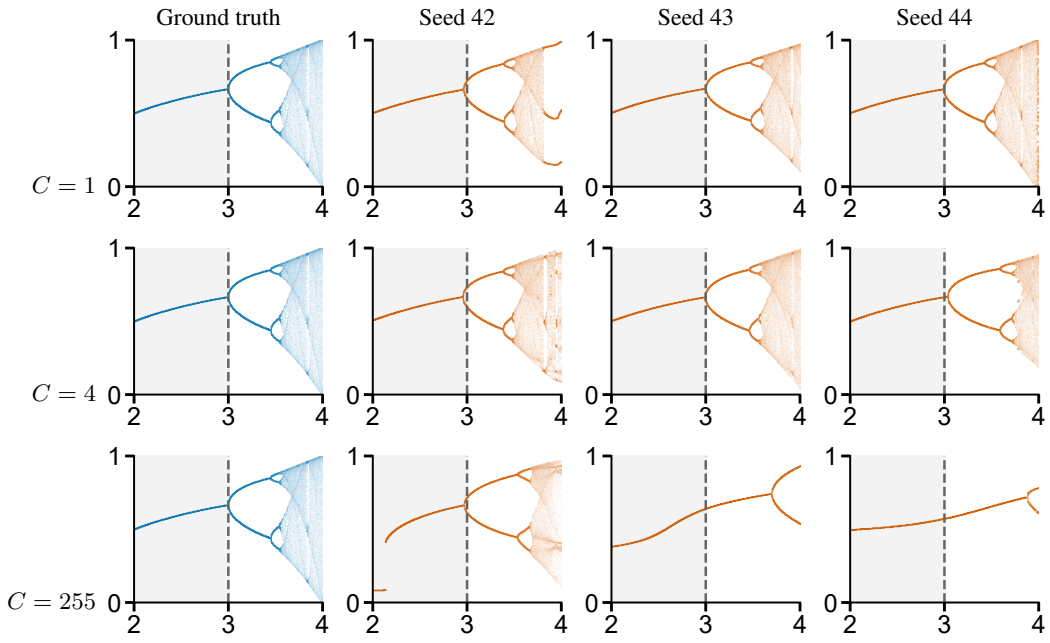


Figure 29: **Logistic Transformer seeds and contexts.** Columns share ground truth with seeds 42–44. Rows mark context length C. Gray shading marks the training interval.

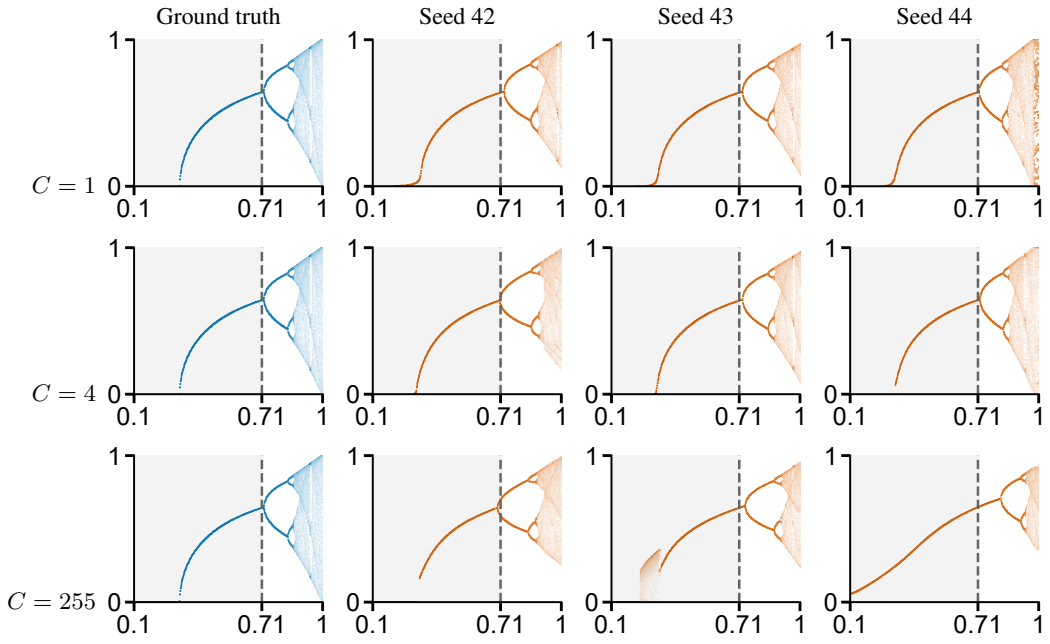


Figure 30: **Sine Transformer seeds and contexts.** Columns share ground truth with seeds 42–44. Rows mark context length C. Gray shading marks the training interval.

O.4 Direct-Map and Polynomial Comparisons

MLP and NVAR comparisons use the same four-column seed layout. Logistic and sine each occupy one figure per baseline.

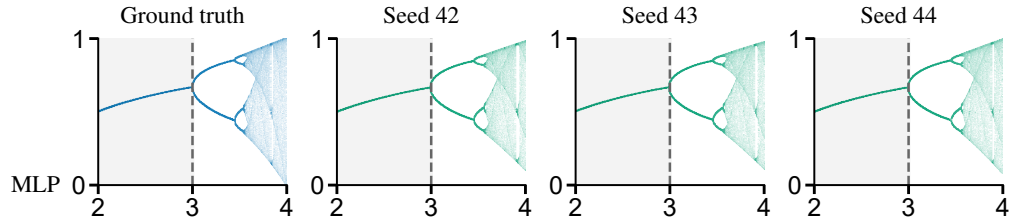


Figure 31: **Logistic MLP comparisons.** Columns are ground truth and seeds 42–44. The MLP directly represents the scalar next-state map.

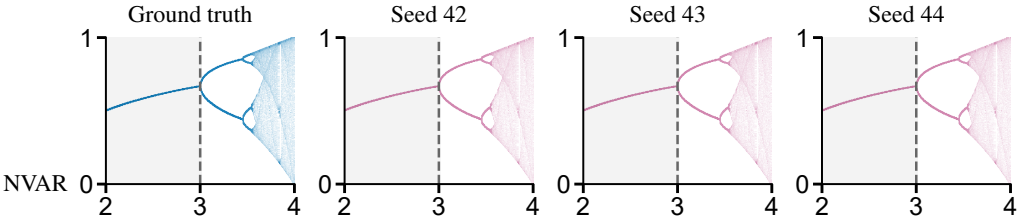


Figure 32: **Logistic NVAR comparisons.** Columns are ground truth and seeds 42–44. NVAR is a parameter-aware polynomial baseline with its separate training protocol.

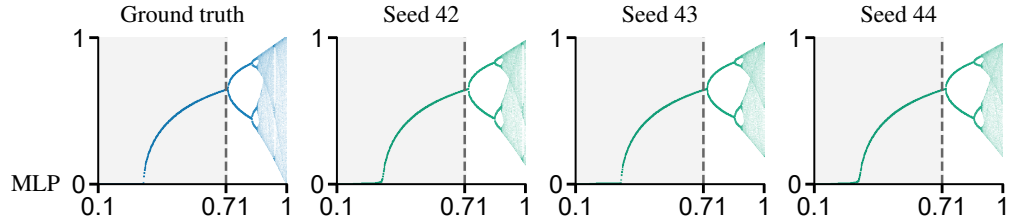


Figure 33: **Sine MLP comparisons.** Columns are ground truth and seeds 42–44. The MLP directly represents the scalar next-state map.

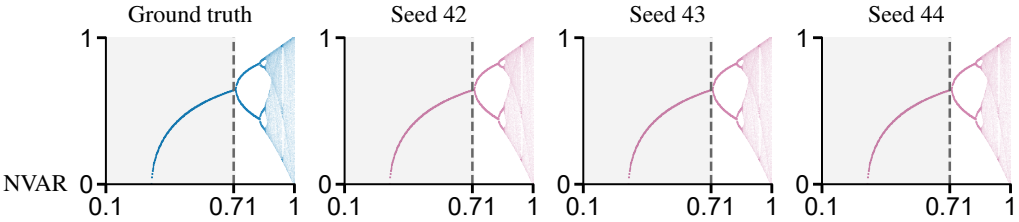
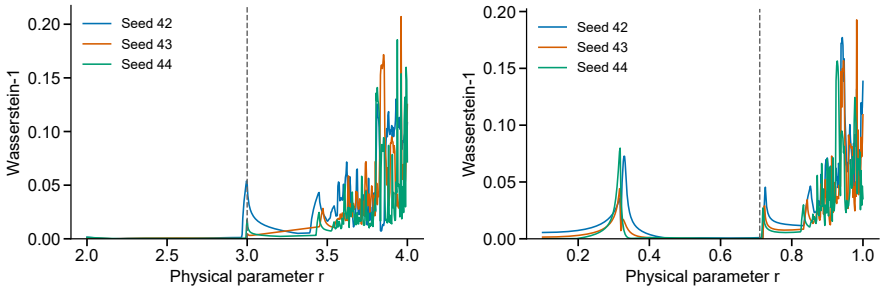


Figure 34: **Sine NVAR comparisons.** Columns are ground truth and seeds 42–44. NVAR is a parameter-aware polynomial baseline with its separate training protocol.

O.5 Where Distributional Errors Occur

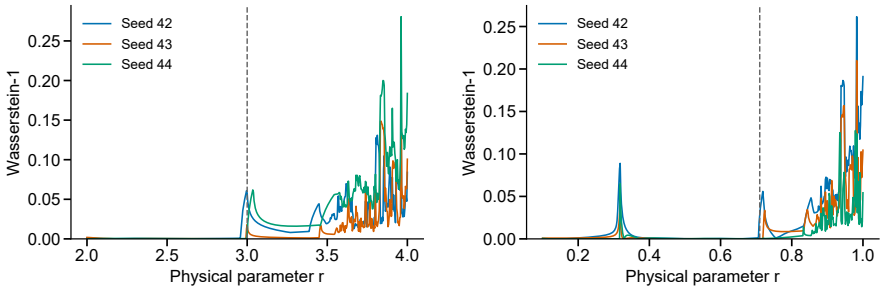
P Parameter Tokens, Channels, and Numerical Computation

The computational question behind our experiments is whether a learned token-space update can approximate the numerical operations of a parameterized dynamical system. We distinguish three questions: whether an architecture can represent such an update, whether optimization discovers it, and whether the learned update preserves dynamics beyond the training region. The analysis below concerns the first question and supplies diagnostics for the other two; it does not assume that successful trajectories establish an exact internal implementation of the governing equation.



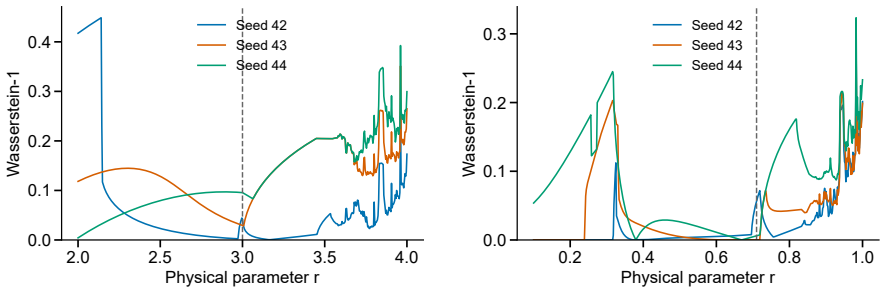
(a) Logistic (b) Sine

Figure 35: **Parameter-resolved fidelity for Transformer $C = 1$.** Lines show each seed separately, using every original uniform-grid Wasserstein-1 measurement. The dashed vertical line marks the training boundary. Vertical ranges adapt to the observed error and are labeled; these curves localize differences summarized by the main OOD table.



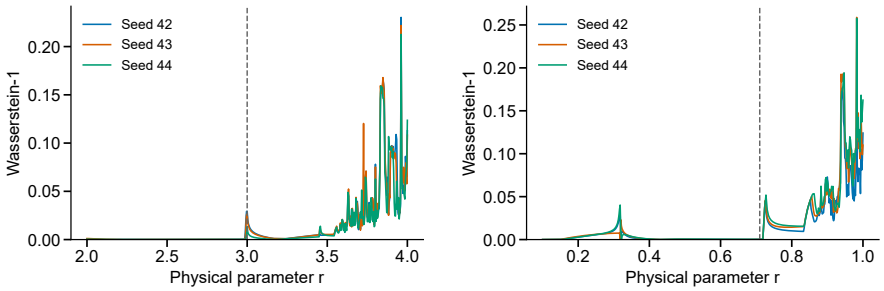
(a) Logistic (b) Sine

Figure 36: **Parameter-resolved fidelity for Transformer $C = 4$.** Lines show each seed separately, using every original uniform-grid Wasserstein-1 measurement. The dashed vertical line marks the training boundary. Vertical ranges adapt to the observed error and are labeled; these curves localize differences summarized by the main OOD table.



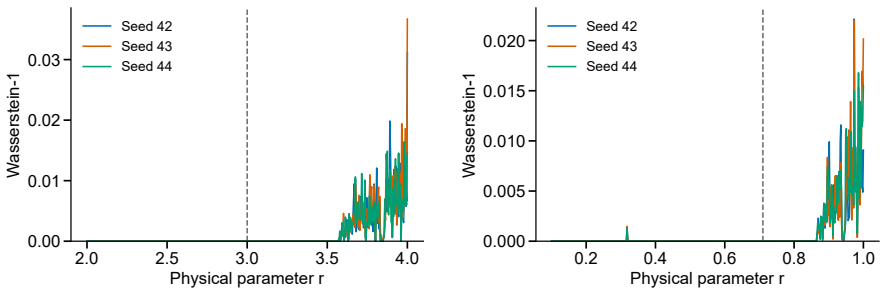
(a) Logistic (b) Sine

Figure 37: **Parameter-resolved fidelity for Transformer $C = 255$.** Lines show each seed separately, using every original uniform-grid Wasserstein-1 measurement. The dashed vertical line marks the training boundary. Vertical ranges adapt to the observed error and are labeled; these curves localize differences summarized by the main OOD table.



(a) Logistic (b) Sine

Figure 38: **Parameter-resolved fidelity for MLP.** Lines show each seed separately, using every original uniform-grid Wasserstein-1 measurement. The dashed vertical line marks the training boundary. Vertical ranges adapt to the observed error and are labeled; these curves localize differences summarized by the main OOD table.



(a) Logistic (b) Sine

Figure 39: **Parameter-resolved fidelity for NVAR.** Lines show each seed separately, using every original uniform-grid Wasserstein-1 measurement. The dashed vertical line marks the training boundary. Vertical ranges adapt to the observed error and are labeled; these curves localize differences summarized by the main OOD table.

P.1 Low-complexity parameter dependence in the scalar benchmarks

Both maps in Equation 6 have the separable form

$$F_r(x) = rg(x), \qquad g(x) = x(1-x) \text{ or } \sin(\pi x). \tag{24}$$

Thus a possible computational decomposition is to encode the scalar state, approximate a single nonlinear feature g, and multiply it by the parameter. The two benchmarks share this parameter dependence even though their state nonlinearities differ. Their success is consequently consistent with an accessible numerical computation, rather than requiring a distinct representation of every attractor. This is a representational hypothesis: the model is not given g, and neither training on trajectories nor reproducing a bifurcation diagram alone identifies this decomposition.

The gated feed-forward block has a useful arithmetic capability. For $s(z) = \mathrm{SiLU}(z) = z\sigma(z)$,

$$s(z) - s(-z) = z. \tag{25}$$

Two gated units can therefore form $s(a)b - s(-a)b = ab$ exactly when a and b are available as linear features at the same token. This identity establishes a multiplication primitive in the tested block; it does not show that the trained weights use this construction. Logistic evaluation additionally requires construction of the state polynomial and its product with r; sine evaluation requires approximating the trigonometric feature on the state domain. Width, depth, feature preservation through residual updates, and access to the parameter all affect the required implementation.

P.2 Routing a parameter into state computation

For a one-state context, our architecture receives two tokens. In the first layer, a state query and parameter key are affine in their respective inputs, so their dot product can contain terms proportional to rx, as well as terms linear in r and x. The attention weight, however, is a normalized exponential of these logits. A bilinear logit is not itself an exact multiplication output.

More directly, attention can transport parameter features without input-dependent routing. Set the query and key projections to zero, making the state token attend equally to the parameter and state tokens. If the embeddings occupy separate feature subspaces and the value projection selects the parameter subspace, the state update receives a fixed multiple of the parameter feature. The residual path retains the state feature, and an output projection can compensate for the factor of two. This construction requires sufficient feature dimensions and available projection choices, but shows why softmax normalization does not prohibit parameter transport at $C = 1$. Subsequent gated computation can combine the two inputs. It also separates *parameter transport through attention* from the stronger claim that *input-dependent attention weights implement the arithmetic*.

The channel alternative embeds the concatenated input through

$$\widetilde{h}_t = W_x x_t + W_r r + b. \tag{26}$$

With sufficient embedding rank, both inputs are already available in the same residual stream. This removes the need to transmit the parameter across a token boundary. With exactly one state token, attention has a single permitted key, its softmax weight is identically one, and its output is an affine value transformation independent of the query/key logits. Nonlinear computation remains available in the feed-forward blocks. With multiple state tokens, channel-based attention is generally input-dependent; the one-token observation does not apply to the longer-context models studied by Zhai et al. (2026).

These representations provide different routes to the same numerical inputs. A separate token exposes an intervenable communication path, whereas a parameter channel provides direct access at every state position. Neither construction implies a general expressivity advantage. A matched comparison should hold the backbone, parameter budget, training samples, optimizer, and context fixed. Adding a parameter-independent dummy token to the channel model further distinguishes parameter placement from the change in token count. Positional encoding, masking, and the full network must also be considered before making claims about permutation symmetry.

P.3 Diagnostics for learned computation and training time

Equation 24 predicts $\partial_r F = g(x)$, $\partial_r^2 F = 0$, and $\partial_x F = rg'(x)$. These yield function-level tests beyond pointwise prediction: parameter-Jacobian error, spurious parameter curvature, and state-Jacobian error on fixed ID and OOD probes. Derivatives must be measured in physical coordinates,

with normalization accounted for; the interior unclipped map is the relevant object when rollout clipping is present. Derivative agreement still does not identify an internal circuit. Routing interventions and matched channel controls address complementary causal questions.

For a scalar period-k orbit, stability depends on the multiplier

$$\Lambda = \prod_{j=0}^{k-1} \partial_x F_r(x_j). \tag{27}$$

Consequently, continued improvement in average one-step loss need not monotonically improve orbit stability, flip parameters, or long-run distributions. An early checkpoint can better preserve these quantities even when a later checkpoint fits observed transitions more closely. We assess this possibility with fixed evaluation probes and matched rollout protocols across training stages. Retrospectively favorable OOD checkpoints characterize learning dynamics; a deployable stopping rule must instead be selected using training-region validation and tested on independent runs. The empirical questions remain whether capacity improves local and multi-step fit, whether derivative structure develops with it, and whether those improvements transfer to autonomous dynamics.