RIGHT STATES, WRONG PATHS: CONTROLLING TEMPORAL DEPENDENCE IN CHAOTIC FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

In chaotic systems, small forecast errors grow over time, setting a predictability horizon beyond which forecasts lose track of the reference trajectory. Past this horizon, the goal shifts from tracking a realization to reproducing the correct statistics. Yet matching the distribution of visited states (state-space fidelity) is not enough: those states must also be organized correctly in time (path-space fidelity). We show that the Energy Score (ES), a standard probabilistic objective whose expectation over whole trajectories is uniquely minimized by the true joint distribution, becomes increasingly insensitive to errors in temporal dependence as the effective number of spatial dimensions grows, while remaining sensitive to marginal-variance errors. Similar ES values can therefore hide very different temporal dynamics. We propose a post-hoc calibration method that makes temporal dependence an explicit, adjustable target. For generators with per-time stochastic inputs, such as the coarse-to-fine JUMP–COMPLETE forecaster we use, correlating these inputs across forecast times changes temporal dependence while keeping the learned weights fixed and the one-time latent marginals unchanged. The coupling is selected with a temporal Variogram Score; we show that its standard finite-ensemble estimator is biased and can move temporal correlations away from the reference, and use an unbiased version instead. Across three chaotic systems, calibration substantially improves temporal correlations and, for two of them, temporal spectra, while joint ES remains nearly unchanged.

1 INTRODUCTION

Trajectories of physical systems provide a natural source of supervision for neural surrogates (Takamoto et al., 2022; Koehler et al., 2024). In chaotic dynamics, however, small forecast errors grow until the forecast loses track of the reference trajectory (Boffetta & Musacchio, 2017; Lorenz, 1969). The corresponding time scale, beyond which forecasts lose synchronization, is the *predictability horizon*. Beyond this horizon, pointwise error becomes increasingly uninformative (Jiang et al., 2023). The relevant question is no longer whether a forecast reproduces the particular reference trajectory, but whether it captures the distribution of possible future behavior.

Stochastic surrogates are a natural response to this regime because they represent multiple plausible futures rather than a single trajectory (Lippe et al., 2023). Yet reproducing the distribution of individual states is not sufficient to reproduce the dynamics of the system. Two trajectory distributions can have similar one-time statistics while differing substantially in how states are organized in time. This suggests a distinction between *state-space fidelity*, which concerns whether the correct distribution of states is reproduced, and *path-space fidelity*, which concerns whether those states are organized correctly in time.

Examples of the importance of path-space fidelity arise across a range of physical systems. Velocity correlations determine turbulent dispersion (Taylor, 1922), space–time source correlations influence radiated jet noise (Tam & Auriault, 1999), and in drift-wave turbulence, changes in fluctuation frequencies can alter particle flux through the density–potential cross-phase (Kim & Terry, 2013). Hence, accurate one-time statistics can coexist with incorrect temporal dynamics.

Figure 1 makes this distinction concrete. Re-anchoring a rollout to the true state, an oracle version of inference-time corrections such as Thermalizer (Pedersen et al., 2025), drives pointwise error to nearly zero but introduces an unphysical jump in the trajectory. State accuracy thus need not imply

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.

path-space fidelity. This motivates our central question: *how strongly do probabilistic objectives used for training and model selection resolve temporal dependence, and how can it be corrected when they do not?*

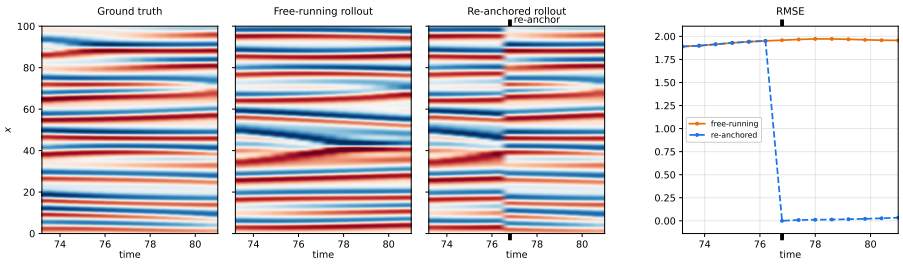


Figure 1: **Lower pointwise error can correspond to a worse path.** 1D Kuramoto–Sivashinsky. The free-running and re-anchored rollouts share the same history and noise up to the intervention. At $t^\star = 76.8$, the re-anchored rollout is reset to the ground-truth state. This reset reduces its RMSE to nearly zero, but introduces a visible discontinuity in the trajectory. In contrast, the free-running rollout has larger pointwise error while preserving a temporally continuous evolution.

As a representative objective for probabilistic forecasting, we study the Energy Score (ES) (Gneiting & Raftery, 2007), which has recently been used to train probabilistic neural operators (Bülte et al., 2025). When applied to whole trajectories, ES is minimized by the full joint trajectory distribution and, in principle, captures both one-time state statistics and temporal dependence. We show, however, that its sensitivity to temporal dependence weakens as spatial dimensionality grows, even while it remains sensitive to errors in marginal variance. We therefore propose to calibrate temporal dependence after training. For generators with per-time stochastic inputs, a single parameter ρ correlates these inputs across time. We instantiate this control with JUMP–COMPLETE, a coarse-to-fine forecaster, and select ρ with a temporal Variogram Score (Scheuerer & Hamill, 2015), a dependence-sensitive score based on increments between forecast times.

Contributions. Our contributions are:

- **We show that the Energy Score can miss errors in temporal dependence.** Although it is proper for whole trajectories, its sensitivity to temporal-correlation errors decreases as the effective spatial dimension grows, so a model can match the distribution of states while generating trajectories with incorrect temporal structure.
- **We propose a post-hoc calibration of temporal dependence.** Correlating per-time stochastic inputs changes temporal correlations and spectra with the learned weights fixed and one-time forecast statistics nearly unchanged; we show this on JUMP–COMPLETE and on an endpoint-free forecaster.
- **We make the calibration criterion reliable with finite ensembles.** The standard Variogram Score estimator is biased and can push temporal correlations in the wrong direction during training; we use its unbiased U-statistic version for both training and selection.

Relation to prior work. Our work connects two lines of research on probabilistic forecasting. First, multivariate proper scores are known to have limited discrimination for dependence misspecification (Pinson & Tastu, 2013; Scheuerer & Hamill, 2015; Pic et al., 2025); we quantify this effect for high-dimensional temporal-covariance errors and show how finite-ensemble variogram training can bias the learned temporal dependence. Second, some forecasting methods predict beyond a single next state, including temporal bundling (Brandstetter et al., 2022), endpoint-conditioned interpolation (Rühling Cachay et al., 2023), and coarse-to-fine approaches (Couairon et al., 2026). We connect these two perspectives through temporally correlated stochasticity. By exposing explicit stochastic inputs at different forecast times, we can vary their temporal dependence while keeping the learned model fixed, and use this controlled intervention to probe and calibrate path-space fidelity. In particular, we repurpose the progressive noise prior of Ge et al. (2023), originally introduced as a training prior for video generation, into a *post-hoc* mechanism for controlling temporal dependence in stochastic neural surrogates. Further related work is discussed in Appendix A.

2 PROBLEM SETUP: LONG-HORIZON PATH FIDELITY

We consider trajectories $x_{0:T}$ of a chaotic dynamical system, stored as sequences of spatial fields sampled at regular time steps. At forecast origin t, let c_t denote the available context, such as the current state x_t. A stochastic surrogate induces a distribution over predicted trajectories,

$$\hat{x}_{t+1:t+H} \sim p_{\Theta,\kappa}(\cdot \mid c_t), \tag{1}$$

where Θ denotes the learned model parameters and κ controls how stochasticity is coupled between forecast times. The separation matters throughout: Θ is fixed by training, whereas κ remains adjustable at inference.

We distinguish three notions of forecast fidelity: tracking fidelity, state-space fidelity, and path-space fidelity. Tracking fidelity captures the conventional pointwise notion of forecast accuracy, while the distinction between state-space and path-space fidelity, introduced below, is central to this work.

Tracking fidelity. Tracking fidelity measures agreement with the *reference trajectory* at corresponding lead times. It is informative while the forecast remains synchronized with the reference, but in chaotic systems errors grow until the two trajectories decorrelate. Matched-time error then approaches the random-pair scale and no longer distinguishes surrogates with different long-run state or temporal statistics.

State-space fidelity. State-space fidelity asks whether generated states reproduce the distributional structure of states visited by the reference dynamics, without requiring matched-time synchronization. It concerns one-time or order-insensitive properties such as moments, spatial correlations, and spatial spectra, and at long horizons includes fidelity to the invariant distribution. A surrogate may therefore have poor tracking fidelity while retaining accurate state-space statistics.

Path-space fidelity. Path-space fidelity asks whether generated trajectories reproduce the temporal dependence of the reference dynamics, rather than merely visiting states with the correct one-time statistics. Let C_ℓ^{ref} and C_ℓ^{gen} denote the lag-ℓ temporal correlations of the reference and generated processes, respectively. Let us define the normalized lag-ℓ correlation error

$$\widetilde{\Delta C}_\ell = \frac{C_\ell^{\mathrm{gen}} - C_\ell^{\mathrm{ref}}}{1 - C_\ell^{\mathrm{ref}}}. \tag{2}$$

Zero is ideal, with positive and negative values indicating over- and under-correlation, respectively. The normalization expresses the correlation error relative to the amount of decorrelation at that lag. We use $\widetilde{\Delta C}_1$ as our primary diagnostic of short-lag temporal dependence. Correlations are computed independently at each spatial cell and then averaged across cells. To probe temporal structure beyond a single lag, we additionally report complementary temporal diagnostics, including temporal power spectra (Appendix B).

State-space and path-space fidelity capture complementary properties of generated trajectories. In particular, state-space fidelity alone does not imply path-space fidelity. Independently sampling from the correct invariant distribution illustrates this distinction: it can reproduce the correct one-time state statistics while completely removing temporal dependence.

3 ENERGY SCORE RESOLUTION OF TEMPORAL DEPENDENCE

We measure probabilistic agreement with the Energy Score (ES) (Gneiting & Raftery, 2007). Let P denote the reference trajectory distribution and Q the predictive distribution, with target $y \sim P$ and ensemble members $\hat{y}^{(m)} \sim Q$. For an ensemble $\{\hat{y}^{(m)}\}_{m=1}^{M}$, the ES is

$$\mathcal{L}_{\mathrm{ES}} = \frac{1}{M}\sum_{m=1}^{M} \|\hat{y}^{(m)} - y\|_2 - \frac{1}{2M(M-1)} \sum_{m \neq n} \|\hat{y}^{(m)} - \hat{y}^{(n)}\|_2. \tag{3}$$

Applied to whole trajectory segments, ES is strictly proper for the joint law and hence, in principle, sensitive to both state-space and path-space fidelity.

Strict propriety, however, does not imply equal resolution of all forms of misspecification. Motivated by high-dimensional results for kernel two-sample tests (Yan & Zhang, 2023), we ask (i) how ES sensitivity to temporal-covariance and marginal-variance errors scales with effective spatial dimension, and (ii) how many forecast cases are needed to detect a temporal-dependence error.

For independent $X, X' \sim P$ and $Y, Y' \sim Q$, define the energy distance

$$D_E(P, Q) = 2\mathbb{E}\|X - Y\| - \mathbb{E}\|X - X'\| - \mathbb{E}\|Y - Y'\|.$$

Averaging over $Y \sim P$ gives $\mathbb{E}_{Y\sim P}[\mathrm{ES}(Q, Y) - \mathrm{ES}(P, Y)] = \frac{1}{2} D_E(P, Q)$.

Population resolution. We first quantify how the ES signal for a fixed temporal-covariance error scales with spatial dimensionality. Consider zero-mean Gaussian trajectories

$$P_d = \mathcal{N}(0, \Sigma_{x,d} \otimes \Sigma_t^P), \qquad Q_d = \mathcal{N}(0, \Sigma_{x,d} \otimes \Sigma_t^Q),$$

with $\operatorname{tr} \Sigma_t^P = \operatorname{tr} \Sigma_t^Q$. This fixes the total variance, so any ES separation reflects covariance structure rather than overall scale.

We define the effective spatial dimension $D_{\mathrm{eff},x}$, the relative temporal-covariance error η_t, and the variance scale t_d by

$$D_{\mathrm{eff},x} = \frac{(\operatorname{tr} \Sigma_{x,d})^2}{\|\Sigma_{x,d}\|_F^2}, \qquad \eta_t = \frac{\|\Sigma_t^Q - \Sigma_t^P\|_F}{\operatorname{tr} \Sigma_t^P}, \qquad t_d = \operatorname{tr} \Sigma_{x,d} \operatorname{tr} \Sigma_t^P.$$

$D_{\mathrm{eff},x}$ can be interpreted as the effective number of spatial features carrying distinct variability. It is large for many independent features and small for redundant ones

When many spatial features vary independently, distances between trajectories concentrate around a value set by the total variance, so a temporal-covariance error only affects their fluctuations, which vanish with dimension.

Proposition 1 (Temporal-covariance dilution)**.** *If Σ_t^P, Σ_t^Q are fixed and $D_{\mathrm{eff},x} \to \infty$, then*

$$\frac{D_E(P_d, Q_d)}{\sqrt{2t_d}} = \frac{\eta_t^2}{8 D_{\mathrm{eff},x}} + \mathcal{O}\Big(D_{\mathrm{eff},x}^{-3/2}\Big). \tag{4}$$

This implies that, for a fixed relative temporal-covariance error η_t, the ES gap relative to the overall trajectory scale decreases as $D_{\mathrm{eff},x}^{-1}$, so joint ES becomes weakly sensitive to temporal dependence as spatial dimensionality grows. The proof and a more general distribution-free decomposition are given in Appendix D.

Detection budget. We next quantify when this loss of sensitivity becomes practically relevant by asking how many forecast cases ES needs to distinguish the predictive distribution Q from the reference distribution P for a fixed temporal-dependence error. For N cases scored with independent M-member ensembles, we use a paired-test power calculation (Diebold & Mariano, 2002).

Let $D_{\mathrm{eff},t} = (\operatorname{tr} \Sigma_t^P)^2 / \|\Sigma_t^P\|_F^2$ be the temporal analogue of $D_{\mathrm{eff},x}$, i.e. the effective number of approximately independent time steps in the forecast window. For small η_t, the number of cases required to detect a temporal-covariance error at significance level α and power $1 - \beta$ scales as

$$N^\star \propto \frac{D_{\mathrm{eff},x}}{M D_{\mathrm{eff},t}\, \eta_t^4}, \tag{5}$$

whereas the number required to detect a small relative marginal-variance error ϵ scales as

$$N^\star_{\mathrm{var}} \propto \frac{1}{D_{\mathrm{eff},x} M D_{\mathrm{eff},t}\, \epsilon^4}. \tag{6}$$

Spatial dimension therefore has opposite effects on the two errors. It increases the budget in equation 5, since the temporal-covariance signal shrinks faster than the ensemble noise, and decreases it in equation 6, since the variance signal stays roughly constant while the noise shrinks. The full expressions for both budgets, including constants, are derived in Appendix E. Empirically, this scaling holds for Gaussian trajectories and in 1D Kuramoto–Sivashinsky, where the ES gap for temporal-dependence errors decays with $D_{\mathrm{eff},x}$ while that for variance errors stays flat (Appendix E.1).

In short, a low joint ES does not certify correct temporal dependence. At the effective dimensions of our systems, temporal-covariance errors fall well below what a realistic number of forecast cases can resolve. We therefore use joint ES as an overall probabilistic score, but assess path-space fidelity separately with dedicated temporal diagnostics.

4 PATH-SPACE CONTROL FOR STOCHASTIC CHAOTIC SURROGATES

Since joint ES cannot reliably enforce temporal dependence, we make it an explicit calibration target, adjusted after training and selected with a criterion sensitive to temporal dependence. In the surrogate of equation 1, the parameters Θ are fixed by training, whereas the coupling κ of stochastic inputs across forecast times remains free. We exploit this freedom in three steps. Section 4.1 introduces the generator we use, JUMP–COMPLETE, which receives one stochastic input per forecast time. Section 4.2 parameterizes κ by a single coefficient ρ that changes temporal dependence with the model fixed, providing both a calibration control and a probe of ES sensitivity on trained surrogates. Section 4.3 introduces the criterion used to select ρ, a temporal Variogram Score with an unbiased finite-ensemble estimator.

4.1 A GENERATOR WITH PER-TIME STOCHASTIC INPUTS

We use JUMP–COMPLETE (Figure 2) as a coarse-to-fine construction for long-horizon forecasting. The Jumper predicts a distant future state in a single evaluation, and the Completer then generates the intermediate states jointly in a second one. Each autoregressive transition therefore advances the forecast by a whole window rather than a single frame, which limits error accumulation over long rollouts and requires only two network evaluations per window, compared with several per frame for diffusion- or refinement-based surrogates (Appendix C.6).

The Jumper samples the distant endpoint, while the Completer models how the trajectory is organized in time conditional on it. Rather than generating every future state sequentially, JUMP–COMPLETE first samples the endpoint and then jointly generates the intermediate trajectory:

$$p_{\Theta,\kappa}(\hat{x}_{t+1:t+H} \mid c_t) = p_\theta(\hat{x}_{t+H} \mid c_t)\, p_{\phi,\kappa}(\hat{x}_{t+1:t+H-1} \mid c_t, \hat{x}_{t+H}), \tag{7}$$

with $\Theta = (\theta, \phi)$ and κ denoting the joint law of the Completer's stochastic inputs across forecast times.

The stochastic Jumper produces the endpoint, while the Completer jointly generates the interior states,

$$\hat{x}_{t+1:t+H-1} = C_\phi(c_t, \hat{x}_{t+H}, H, z_{1:H-1}), \qquad z_{1:H-1} \sim \kappa. \tag{8}$$

Crucially, the Completer receives one stochastic input per interior forecast time. This design allows us to choose how these inputs are correlated across time, from independent to fully shared draws. This correlation is governed by the coupling κ, which we parameterize in Section 4.2.

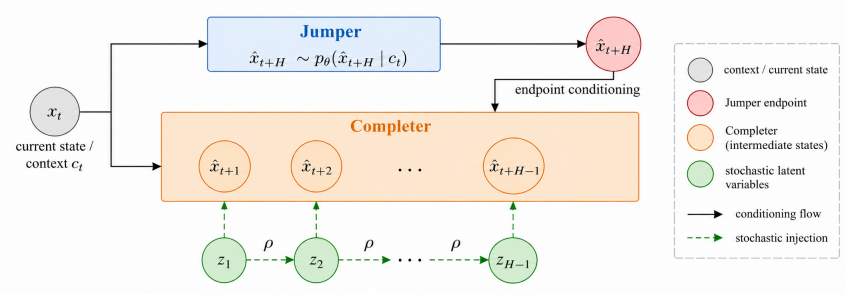


Figure 2: **JUMP–COMPLETE construction.** The Jumper samples a distant endpoint $\hat{x}_{t+H}$ from context c_t, while the Completer generates the intermediate states conditioned on this endpoint. The coupling coefficient ρ controls the temporal correlation of the stochastic inputs z_h.

For long free-running forecasts, we use an endpoint-conditioned rollout in which the Jumper endpoint conditions the current completion, while the final Completer state provides the context for the next window. This allows the generated path itself to determine the subsequent rollout. We additionally consider an endpoint-anchored variant, in which the Jumper endpoint is retained as the boundary state. This provides a controlled setting in which the coarse endpoint trajectory is fixed while the intermediate path can vary, echoing the distinction in Figure 1 between pointwise state agreement and temporal continuity.

4.2 POST-HOC TEMPORAL CONTROL VIA LATENT COUPLING

We now introduce the proposed control. Recall that in equation 1, κ denotes the joint law of the stochastic inputs across forecast times, which can be adjusted after training. We parameterize κ with a simple stationary coupling that preserves each one-time input marginal. For any generator with per-time stochastic inputs, including JUMP–COMPLETE, each input mixes the previous one with independent Gaussian noise.

Let $z_{1:H-1}$ denote the stochastic inputs associated with the interior forecast times. Starting from $z_1 \sim \mathcal{N}(0, I)$, we generate

$$z_h = \rho z_{h-1} + \sqrt{1-\rho^2}\,\epsilon_h, \qquad \epsilon_h \overset{\text{iid}}{\sim} \mathcal{N}(0, I), \qquad \rho \in [0, 1], \tag{9}$$

and denote the resulting joint law by κ_ρ.

For every ρ, each $z_h \sim \mathcal{N}(0, I)$, while $\mathrm{Cov}(z_h, z_{h+\ell}) = \rho^\ell I$. Thus, varying ρ changes only the cross-time dependence of the latent inputs, not their one-time marginals. At $\rho = 0$, the input is resampled independently at each time; at $\rho = 1$, a single draw is shared across the window.

The inputs z_h enter the model through the same learned projection at every forecast time, with a noise scale held fixed. Varying ρ therefore changes only how the inputs are correlated across time, not their amplitude or one-time marginal distribution. Because the network mixes information across forecast times, this does not guarantee that the one-time marginals of the generated states are preserved. We verify empirically that they vary only weakly with ρ (Appendix F.3).

In practice, we distinguish the training coupling ρ_{train} from the inference coupling ρ_{inf}. Once Θ is frozen, ρ_{inf} is the parameter we calibrate, although it changes the joint law of the inputs relative to training. It remains to choose its value, which requires a criterion sensitive to temporal dependence.

4.3 SELECTING THE COUPLING: A FAIR TEMPORAL VARIOGRAM SCORE

The criterion used to select ρ_{inf} must be sensitive to temporal dependence, which excludes joint ES (Section 3). We instead use the Variogram Score (VS) (Scheuerer & Hamill, 2015), a dependence-sensitive scoring rule based on pairwise increments. It serves both to select ρ_{inf} after training and, optionally, as a training loss alongside ES. To target temporal rather than spatial dependence, we restrict the pairwise comparisons to forecast times at fixed spatial–channel locations, yielding a temporal Variogram Score, VS_t.

A joint spatiotemporal VS would mix spatial and temporal discrepancies and require pairwise comparisons across the full spatiotemporal field, whose number grows quadratically with the number of grid points. Instead, we compute VS_t independently at each spatial–channel location and average over locations, which makes the cost linear in the number of locations. This isolates discrepancies in temporal increments while remaining agnostic to spatial ordering.

With $s \in \mathcal{S}$ indexing spatial–channel coordinates and $i, j \in \{1, \dots, K\}$ forecast times, we define

$$\mathcal{L}_{\mathrm{VS}_t} = \frac{1}{|\mathcal{S}|Z} \sum_s \sum_{i<j} w_{ij} \Big(\mathbb{E}_{\hat{X}} |\hat{X}_i(s) - \hat{X}_j(s)|^p - |x_i(s) - x_j(s)|^p \Big)^2, \tag{10}$$

where $Z = \sum_{i<j} w_{ij}$, $p = 1/2$, and $w_{ij} = |i-j|^{-1}$. We train with $\mathcal{L} = \mathcal{L}_{\text{ES}} + \lambda \mathcal{L}_{\mathrm{VS}_t}$.

The population objective in Equation 10 contains an expectation under the predictive distribution, which must be estimated from a finite ensemble during training. The natural plug-in estimator is generally biased. For a single temporal pair and spatial location, let

$$b^{(m)} = |\hat{x}_i^{(m)}(s) - \hat{x}_j^{(m)}(s)|^p - |x_i(s) - x_j(s)|^p.$$

Replacing the predictive expectation by the ensemble mean yields

$$\mathbb{E}[\bar{b}^2] = (\mathbb{E}b)^2 + \frac{\mathrm{Var}(b)}{M}. \tag{11}$$

Hence the expected finite-ensemble objective differs from the population objective by a forecast-dependent $O(M^{-1})$ term. This term also has a directional effect because, if predictive increments

vary mainly through a scale σ, then $\mathrm{Var}(b) = \mathrm{Var}(|\Delta\hat{X}|^p) \propto \sigma^{2p}$. The plug-in estimator therefore favors less dispersed temporal increments and, at comparable one-time variability, can favor more strongly correlated trajectories.

As an alternative, following the principle of fair finite-ensemble scoring (Ferro, 2014), we remove this term using the unbiased U-statistic

$$\hat{b}_U^2 = \frac{1}{M(M-1)} \sum_{m \neq n} b^{(m)} b^{(n)} = \bar{b}^2 - \frac{s_b^2}{M}, \qquad \mathbb{E}[\hat{b}_U^2] = (\mathbb{E}b)^2. \tag{12}$$

This unbiased estimator excludes the self-products $b^{(m)}b^{(m)}$ that cause the bias in equation 11, so it no longer rewards less dispersed temporal increments, at the same computational cost.

Put together, for generators with per-time stochastic inputs, we propose to calibrate the inference coupling ρ_{inf} by minimizing the unbiased estimate of VS_t on validation data. This allows us to correct, without retraining, errors in temporal dependence that ES cannot reliably detect.

5 Experiments

We evaluate the proposed post-hoc calibration of temporal dependence in stochastic surrogates of chaotic systems. We first compare the unbiased criterion with the standard plug-in estimator, showing that the latter can push temporal correlations away from the reference during training (Section 5.2). We then show that varying the inference coupling substantially changes temporal correlations with little effect on joint ES (Section 5.3). Finally, we select the coupling with the temporal Variogram Score (VS_t, equation 10) on short windows and assess whether this calibration carries over to long-horizon free-running forecasts (Section 5.4).

5.1 Experimental setup

We consider three chaotic systems: Kolmogorov flow (Kolmo), 2D Kuramoto–Sivashinsky (KS), and the Hasegawa–Wakatani system (HW2D). Unless stated otherwise, experiments use the Jump–Complete setting with 15 interior completion frames ($H = 16$) for Kolmo and KS and 11 ($H = 12$) for HW2D. The Completer is trained on truth-compatible Jumper-generated endpoints. Our main metrics are the joint Energy Score (ES, equation 3) and, for temporal fidelity, the normalized lag-one correlation error $\widetilde{\Delta C}_1$ (equation 2). Architectures, hyperparameters, and model-selection details are given in Appendix C. Reported standard deviations that appear as zero are smaller than the displayed numerical precision, rather than identically zero.

5.2 Finite-Ensemble Bias Changes Dependence-Sensitive Training

Since VS_t serves as our calibration criterion, we first verify that its estimator is reliable. Training with VS_t as an additional loss provides a direct test, as a biased estimator would steer the learned temporal dependence. On Kolmo, we train with $\mathcal{L} = \mathcal{L}_{\mathrm{ES}} + \lambda\mathcal{L}_{\mathrm{VS}_t}$, varying λ with ensemble size $M = 4$ and using either the plug-in or the unbiased estimator (Figure 3). As λ increases, both estimators reduce the evaluated VS_t, but temporal correlations move in opposite directions. The plug-in estimator drives $\widetilde{\Delta C}_1$ farther from zero and produces overly smooth trajectories, whereas the unbiased estimator moves it toward the reference. A biased estimator can thus *improve its own score while degrading temporal fidelity*. The unbiased estimator avoids this failure, making it a reliable calibration criterion. The same pattern appears on HW2D (Appendix F.1), and on Kolmo the gap between the two estimators shrinks as the ensemble size grows (Appendix F.2).

5.3 Temporal Dependence Remains Adjustable After Training

We now evaluate the proposed post-hoc calibration. With the learned weights frozen, we vary only the inference coupling ρ_{inf} of equation 9, from independent ($\rho = 0$) to fully shared inputs ($\rho = 1$). Figure 4 shows that this provides an effective post-training control of temporal dependence. Across checkpoints trained with different ρ_{train}, varying ρ_{inf} produces a large and consistent shift in $\widetilde{\Delta C}_1$ and VS_t, while joint ES changes only weakly. Importantly, one-time ES, state variance, and spatial

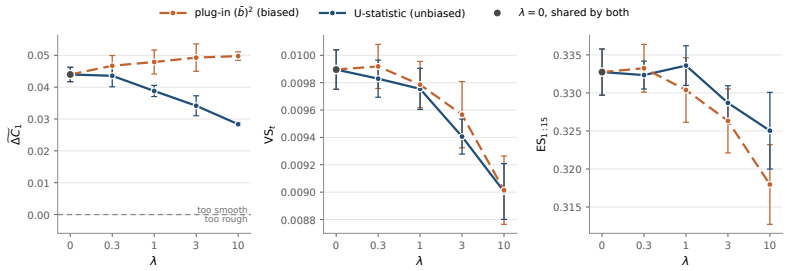


Figure 3: **Finite-ensemble bias changes dependence-sensitive training.** Kolmo experiment with ensemble size $M = 4$. We vary the VS_t weight $\lambda \in \{0, 0.3, 1, 3, 10\}$ using either the plug-in or unbiased estimator. **Left:** normalized lag-one correlation error. **Middle:** VS_t, evaluated with the unbiased estimator. **Right:** joint ES. Error bars show mean $\pm$ s.d. across three seeds.

spectra remain nearly unchanged across the coupling sweep (Appendix F.3). Thus, *changing* ρ_{inf} *can substantially alter temporal dynamics while largely preserving the one-time forecast statistics*. On Kolmo, this correction cannot be recovered by rescaling the noise amplitude alone, which only partially corrects the lag-one correlation error while degrading joint ES (Appendix F.4).

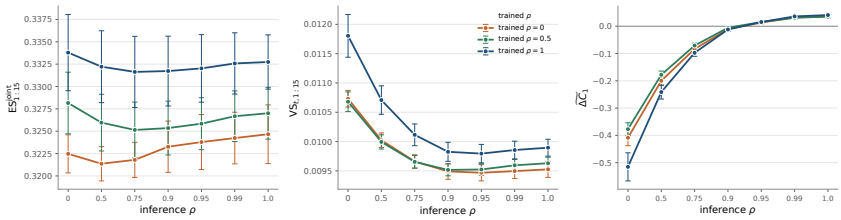


Figure 4: **Temporal dependence remains adjustable with fixed model weights.** Kolmo checkpoints trained at three coupling values. Within each checkpoint, only the temporal coupling of the stochastic inputs is varied. Left: joint ES for completion forecasts. Center: VS_t. Right: normalized lag-one correlation error across inference couplings.

This control is not specific to JUMP–COMPLETE: it persists in the endpoint-free JOINT-FORECAST model (Appendix F.7), at a coarser temporal resolution (Appendix F.8), and after training with VS_t (Appendix F.9). Taken together, the inference coupling controls temporal dependence after training while leaving one-time statistics and joint ES nearly unchanged.

5.4 CONSEQUENCES FOR LONG-HORIZON DYNAMICS

We finally test whether the proposed calibration carries over to long free-running forecasts. Following Section 4, we select the coupling $\rho^\star$ that minimizes VS_t on short validation windows. Since different temporal diagnostics can prefer different couplings (Appendix F.5), $\rho^\star$ is a calibration parameter for the targeted temporal property rather than a universally optimal setting. We compare against an endpoint-free JOINT-FORECAST model, DYffusion (Rühling Cachay et al., 2023), and PDE-Refiner (Lippe et al., 2023), to assess whether surrogates with similar long-horizon ES differ in temporal fidelity.

Table 1 reports 496-frame rollouts obtained by chaining completion windows in the endpoint-conditioned variant, where the final Completer state initializes the next window, so coupling acts on both the paths and the boundaries between them. On Kolmogorov flow and KS, *calibration brings the lag-one correlation error within windows substantially closer to zero with essentially unchanged joint ES*, with a smaller gain on HW2D; the error across window boundaries decreases on KS but increases on Kolmogorov flow and HW2D.

At this horizon, joint ES is already close to the climatological scale (Appendix F.12), since trajectory-scale variability dominates it, yet substantial differences in temporal fidelity remain.

Table 1: **Long-horizon free-running forecasts over 496 frames.** Normalized lag-one correlation error within windows (*int*) and across boundaries (*bnd*), together with joint ES. Zero is ideal for correlation errors. †PDE-Refiner has no intrinsic interior/boundary distinction; for comparability, we impose the same partition used for Jump–Complete. Mean ± s.d. results, including the endpoint-anchored variant, are in Appendix F.10; full spectra are in Appendix F.11.

	Kolmo			KS			HW2D		
Method	$\widetilde{\Delta C}_1^{\mathrm{int}}$	$\widetilde{\Delta C}_1^{\mathrm{bnd}}$	$\mathrm{ES}_{1:496}^{\mathrm{joint}}$	$\widetilde{\Delta C}_1^{\mathrm{int}}$	$\widetilde{\Delta C}_1^{\mathrm{bnd}}$	$\mathrm{ES}_{1:496}^{\mathrm{joint}}$	$\widetilde{\Delta C}_1^{\mathrm{int}}$	$\widetilde{\Delta C}_1^{\mathrm{bnd}}$	$\mathrm{ES}_{1:496}^{\mathrm{joint}}$
Jump–Complete	-0.3276	-0.0374	12.7661	-0.0258	-0.0182	14.1221	-0.0974	$+0.1333$	21.3312
+ coupling ($\rho^\star$)	$+0.0687$	$+0.0514$	12.7698	-0.0040	-0.0064	14.1415	$+0.0559$	$+0.2379$	21.3436
Joint-Forecast	$+0.1711$	-0.0224	12.7716	-0.0116	-0.0449	14.4281	$+0.2641$	$+0.2722$	21.3392
DYffusion	-0.1261	-0.6324	12.8207	-0.1464	-1.7452	14.8352	$+0.2539$	-0.6394	21.4055
PDE-Refiner†	$+0.0597$	$+0.0591$	12.8712	$+0.2485$	$+0.2490$	14.7776	$+0.0672$	$+0.0681$	21.2920

Across methods, joint ES varies by less than one percent on Kolmo and HW2D, while $\widetilde{\Delta C}_1$ ranges from strong under- to over-correlation. JUMP–COMPLETE attains the lowest joint ES on Kolmo and KS, with calibration changing it by less than 0.2%, and the smallest within-window error on KS, using substantially fewer network evaluations than DYffusion and PDE-Refiner (Appendix C.6). Boundary errors are smallest for calibrated JUMP–COMPLETE on KS, JOINT-FORECAST on Kolmo, and PDE-Refiner on HW2D.

Figure 5 further shows that short-horizon calibration also improves longer-range temporal structure, substantially reducing the temporal power-spectrum error (tPSD, Appendix B) on Kolmo and HW2D, with a smaller effect on KS.

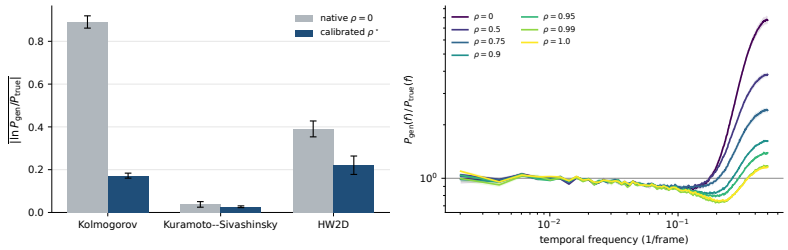


Figure 5: **Stochastic-input correlation impacts temporal power spectra. Left:** temporal power-spectrum error for Kolmo, KS, and HW2D under native $\rho = 0$ and $\rho^\star$ selected using short-horizon VS_t; lower is better. **Right:** generated-to-reference temporal power-spectrum ratio on Kolmo across inference couplings; 1 denotes exact spectral agreement.

Overall, post-hoc calibration selected on short windows improves temporal fidelity over long rollouts without affecting joint ES, and comparable long-horizon ES values can hide very different temporal dynamics across surrogates. This makes post-hoc calibration a cheap complement to training, requiring only a single-parameter sweep and no retraining.

6 CONCLUSION

Beyond the predictability horizon, chaotic forecasting requires generating not only the right states but also the right paths. We showed that the joint Energy Score becomes increasingly insensitive to temporal dependence as the effective spatial dimension grows, so similar ES values can hide very different temporal dynamics. We therefore proposed a post-hoc calibration for generators with per-time stochastic inputs, in which a single coupling parameter correlates these inputs across forecast times with the learned weights frozen, selected with an unbiased estimator of a temporal Variogram Score. Across three chaotic systems, a coupling selected on short windows improves temporal correlations and spectra over long rollouts while leaving joint ES nearly unchanged. This calibration requires per-time stochastic inputs, its single coupling represents one temporal scale, and the preferred value depends on the targeted temporal property. Future work can explore multi-scale couplings and other stochastic architectures.

AI USE STATEMENT

In this work, we used generative AI tools to assist in writing proofs, provide critical ingredients for proving mathematical claims, propose or refine hypotheses, design or provide feedback on research methodology and experiments, where they were prompted to act as reviewers. Through code assistance for our simulation and training pipelines, we also used them to generate synthetic datasets, clean and reformat datasets, and implement methods, and we used them to assist with translation. We have not used generative AI tools to help develop theoretical models or conceptual frameworks, formulate mathematical claims, or interpret results, and supporting qualitative and thematic data analysis is not applicable to this work. Additionally, we used generative AI tools to create or modify scientific figures, create or edit software code, create artifacts, draft parts of the paper, search for information, edit the paper to improve readability, and identify relevant literature.

We have reviewed all AI-assisted work. All AI-generated code was reviewed and tested by one author. AI feedback on methodology was treated as reviewer-style critique, and the authors decided which suggestions to adopt. All literature suggested by AI tools was read and verified by the authors, and all derivations were checked by the authors. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Guido Boffetta and Stefano Musacchio. Chaos and predictability of homogeneous-isotropic turbulence. *Physical review letters*, 119(5):054102, 2017.

Johannes Brandstetter, Daniel Worrall, and Max Welling. Message passing neural pde solvers. *arXiv preprint arXiv:2202.03376*, 2022.

Christopher Bülte, Philipp Scholl, and Gitta Kutyniok. Probabilistic neural operators for functional uncertainty quantification. *arXiv preprint arXiv:2502.12902*, 2025.

Shubhadeep Chakraborty and Xianyang Zhang. A new framework for distance and kernel-based metrics in high dimensions. *Electronic Journal of Statistics*, 15(2):5455–5522, 2021.

Pascal Chang, Jingwei Tang, Markus Gross, and Vinicius Da Costa De Azevedo. How i warped your noise: a temporally-correlated noise prior for diffusion models. In *International Conference on Learning Representations*, volume 2024, pp. 30015–30046, 2024.

Guillaume Couairon, Alexis Jacq, Yu-Han Wu, Renu Singh, Yana Hasson, Quentin Berthet, and Romuald Elie. Kastor: An efficient fine-tuning strategy for generative emulation of pde simulations. *arXiv preprint arXiv:2608.06107*, 2026.

Ilias Diakonikolas, Prahladh Harsha, Adam Klivans, Raghu Meka, Prasad Raghavendra, Rocco A Servedio, and Li-Yang Tan. Bounding the average sensitivity and noise sensitivity of polynomial threshold functions. In *Proceedings of the forty-second ACM symposium on Theory of computing*, pp. 533–542, 2010.

Francis X Diebold and Roberto S Mariano. Comparing predictive accuracy. *Journal of Business & economic statistics*, 20(1):134–144, 2002.

Christopher AT Ferro. Fair scores for ensemble forecasts. *Quarterly Journal of the Royal Meteorological Society*, 140(683):1917–1923, 2014.

Songwei Ge, Seungjun Nah, Guilin Liu, Tyler Poon, Andrew Tao, Bryan Catanzaro, David Jacobs, Jia-Bin Huang, Ming-Yu Liu, and Yogesh Balaji. Preserve your own correlation: A noise prior for video diffusion models. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22873–22884. IEEE, 2023.

Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.

Robin Greif. Hw2d: A reference implementation of the hasegawa-wakatani model for plasma turbulence in fusion reactors. *Journal of Open Source Software*, 8(92):5959, 2023.

Akira Hasegawa and Masahiro Wakatani. Plasma edge turbulence. *Physical Review Letters*, 50(9): 682, 1983.

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

Juhyung Kim and PW Terry. Numerical investigation of frequency spectrum in the hasegawa-wakatani model. *Physics of Plasmas*, 20(10), 2013.

Felix Koehler, Simon Niedermayr, Nils Thuerey, et al. Apebench: A benchmark for autoregressive neural emulators of pdes. *Advances in Neural Information Processing Systems*, 37:120252–120310, 2024.

Mária Lakatos. A composite-loss graph neural network for the multivariate post-processing of ensemble weather forecasts. *Quarterly Journal of the Royal Meteorological Society*, 152(777): e70119, 2026.

Simon Lang, Mihai Alexe, Mariana CA Clare, Christopher Roberts, Rilwan Adewoyin, Zied Ben Bouallègue, Matthew Chantry, Jesper Dramsch, Peter D Dueben, Sara Hahner, et al. Aifs-crps: ensemble forecasting using a model trained with a loss function based on the continuous ranked probability score. *npj Artificial Intelligence*, 2(1):18, 2026.

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

Edward N Lorenz. Atmospheric predictability as revealed by naturally occurring analogues. *Journal of Atmospheric Sciences*, 26(4):636–646, 1969.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=Bkg6RiCqY7`.

Michael McCabe, Bruno Régaldo-Saint Blancard, Liam Parker, Ruben Ohana, Miles Cranmer, Alberto Bietti, Michael Eickenberg, Siavash Golkar, Geraud Krawezik, Francois Lanusse, et al. Multiple physics pretraining for spatiotemporal surrogate models. *Advances in Neural Information Processing Systems*, 37:119301–119335, 2024.

Juan Nathaniel and Pierre Gentine. Generative emulation of chaotic dynamics with coherent prior. *Computer Methods in Applied Mechanics and Engineering*, 448:118410, 2026.

Juan Nathaniel, Yongquan Qu, Tung Nguyen, Sungduk Yu, Julius Busecke, Aditya Grover, and Pierre Gentine. Chaosbench: A multi-channel, physics-based benchmark for subseasonal-to-seasonal climate prediction. *Advances in Neural Information Processing Systems*, 37:43715–43729, 2024.

Lorenzo Pacchiardi, Rilwan A Adewoyin, Peter Dueben, and Ritabrata Dutta. Probabilistic forecasting with generative networks via scoring rule minimization. *Journal of Machine Learning Research*, 25(45):1–64, 2024.

Chris Pedersen, Laure Zanna, and Joan Bruna. Thermalizer: Stable autoregressive neural emulation of spatiotemporal chaos. *arXiv preprint arXiv:2503.18731*, 2025.

Romain Pic, Clément Dombry, Philippe Naveau, and Maxime Taillardat. Proper scoring rules for multivariate probabilistic forecasts based on aggregation and transformation. *Advances in Statistical Climatology, Meteorology and Oceanography*, 11(1):23–58, 2025.

Pierre Pinson and Julija Tastu. Discrimination ability of the energy score. *Technical University of Denmark*, 2013.

Salva Rühling Cachay, Bo Zhao, Hailey Joren, and Rose Yu. Dyffusion: A dynamics-informed diffusion model for spatiotemporal forecasting. *Advances in Neural Information Processing Systems*, 36:45259–45287, 2023.

Salva Rühling Cachay, Miika Aittala, Karsten Kreis, Noah Brenowitz, Arash Vahdat, Morteza Mardani, and Rose Yu. Elucidated rolling diffusion models for probabilistic forecasting of complex dynamics. *Advances in Neural Information Processing Systems*, 38:1821–1860, 2026.

Michael Scheuerer and Thomas M Hamill. Variogram-based proper scoring rules for probabilistic forecasts of multivariate quantities. *Monthly Weather Review*, 143(4):1321–1334, 2015.

Yair Schiff, Zhong Yi Wan, Jeffrey B Parker, Stephan Hoyer, Volodymyr Kuleshov, Fei Sha, and Leonardo Zepeda-Núñez. Dyslim: Dynamics stable learning by invariant measure for chaotic systems. *arXiv preprint arXiv:2402.04467*, 2024.

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

Christopher KW Tam and Laurent Auriault. Jet mixing noise from fine-scale turbulence. *AIAA journal*, 37(2):145–153, 1999.

Geoffrey I Taylor. Diffusion by continuous movements. *Proceedings of the london mathematical society*, 2(1):196–212, 1922.

Jian Yan and Xianyang Zhang. Kernel two-sample tests in high dimensions: interplay between moment discrepancy and dimension-and-sample orders. *Biometrika*, 110(2):411–430, 2023.

Appendix Contents

A Related Work **14**

B Temporal diagnostics **14**

C Implementation details **15**

C.1 Data Generation 15

C.2 Architecture 16

C.3 Objectives and Conditioning 17

C.4 Optimization and Evaluation 17

C.5 Baselines 18

C.6 Model size and inference cost 19

D Energy-distance expansion **20**

E Derivation of the detection budget **22**

E.1 Empirical validation 24

F Additional experiments **25**

F.1 Finite-ensemble variogram training on HW2D 25

F.2 Effect of training ensemble size 25

F.3 State-space controls under inference-time coupling 26

F.4 Cross-time coupling versus latent amplitude 26

F.5 Metric-dependent temporal calibration 27

F.6 Jump–Complete endpoint-conditioning ablations 29

F.7 Cross-time coupling in the endpoint-free Forecast model 29

F.8 Post-training coupling across temporal sampling scales 30

F.9 Interaction between training- and inference-time controls 31

F.10 Full long-horizon forecasting results 32

F.11 Long-horizon temporal power spectra 33

F.12 Energy-Score skill relative to climatology 33

A Related Work

Proper scoring rules and dependence. Proper scoring rules provide principled objectives for evaluating probabilistic forecasts against observed outcomes (Gneiting & Raftery, 2007) and have subsequently been used as training objectives for generative models (Pacchiardi et al., 2024), including neural operator learning (Bülte et al., 2025). The Energy Score (ES) is a strictly proper scoring rule for multivariate probabilistic forecasts that reduces, in the univariate case, to the Continuous Ranked Probability Score (CRPS), a standard objective in probabilistic weather forecasting (Lang et al., 2026; Lakatos, 2026), making ES a natural objective for joint trajectory forecasts. Despite its strict propriety, ES can have limited sensitivity to dependence misspecification (Pinson & Tastu, 2013), motivating dependence-sensitive alternatives such as the Variogram Score (Scheuerer & Hamill, 2015). To the best of our knowledge, prior uses of the Variogram Score as an optimization objective for physics surrogates target spatial dependence and rely on a finite-ensemble plug-in estimator in both generative training and ensemble post-processing (Pacchiardi et al., 2024; Lakatos, 2026), with Pacchiardi et al. (2024) describing this estimator as unbiased. Fair-score corrections more generally account for finite-ensemble effects in probabilistic scoring (Ferro, 2014). We instead define a temporal Variogram Score by applying the Variogram Score to pairs of forecast times at fixed spatial locations, and use its fair finite-ensemble estimator both as a training objective and as a post-training calibration criterion for the temporal coupling of stochastic inputs.

Neural forecasting of chaotic dynamics. Benchmarks such as PDEBench (Takamoto et al., 2022), APEBench (Koehler et al., 2024), and ChaosBench (Nathaniel et al., 2024) evaluate neural PDE and climate emulators using a diverse set of metrics, including statewise rollout errors, physics-based diagnostics, and probabilistic scores, but do not directly assess temporal dependence in generated trajectories. A separate line of work targets stable long-horizon forecasting through iterative refinement in PDE-Refiner (Lippe et al., 2023), objectives that exploit properties of the stationary distribution (Schiff et al., 2024), and inference-time correction of forecast states toward the stationary distribution in Thermalizer (Pedersen et al., 2025). Closest to our setting, Cohesion (Nathaniel & Gentine, 2026) denoises entire trajectories in a single pass with a score network trained on subsequences rather than individual snapshots, which the authors argue enables it to learn temporal structure; however, temporal consistency is not assessed with dedicated temporal-dependence metrics. In contrast to these works, which modify the model or its training objective, our intervention isolates temporal dependence by varying only the joint law of the stochastic inputs while keeping the learned model, conditioning, and single-time latent marginals fixed.

Joint trajectory generation and temporal stochasticity. Generating multiple future states jointly can reduce autoregressive depth and better capture temporal structure, as in temporal bundling (Brandstetter et al., 2022) and boundary-conditioned forecasting methods such as DYffusion (Rühling Cachay et al., 2023), and coarse-to-fine PDE emulation in Kastor (Couairon et al., 2026). In video diffusion, temporally structured noise has been used to promote temporal coherence through correlated noise priors (Ge et al., 2023; Chang et al., 2024). Closer to our setting, ERDM (Rühling Cachay et al., 2026) uses the noise prior of Ge et al. (2023) in a diffusion forecaster, with a fixed temporal correlation during training and sampling selected by marginal CRPS. We instead treat temporal coupling as a post-hoc calibration parameter, varying it at inference with the learned model fixed and selecting it using a temporal-dependence-sensitive criterion, allowing us to directly probe temporal dependence while preserving the one-time latent marginals.

B Temporal diagnostics

Path-space fidelity cannot in general be characterized by a single low-order temporal statistic. We use lag-one correlation as our primary diagnostic because it provides a simple and interpretable measure of short-lag temporal dependence, with its normalized error indicating over- or under-correlation relative to the reference dynamics. We therefore complement it with diagnostics that probe temporal dependence across a broader range of timescales.

The temporal power spectral density characterizes how trajectory variance is distributed across temporal frequencies. For each spatial location (and channel, when applicable), we compute the temporal spectrum along the forecast trajectory and then average over spatial locations and trajectories.

We denote the resulting spectrum by

$$\mathrm{PSD}_t(f).$$

We report the ratio

$$R_{\mathrm{tPSD}}(f) = \frac{P_{\mathrm{gen}}(f)}{P_{\mathrm{ref}}(f)},$$

for which one denotes exact spectral agreement. Values above and below one indicate excess and deficient temporal power, respectively. The scalar tPSD error reported is

$$\overline{\left|\log \frac{P_{\mathrm{gen}}(f)}{P_{\mathrm{ref}}(f)}\right|},$$

averaged over the evaluated frequencies.

C Implementation details

C.1 Data Generation

Both Kolmogorov Flow (Kolmo) and 2D Kuramoto–Sivashinsky (KS) are generated with APEBench (Koehler et al., 2024) on a periodic square domain at 160×160, single channel, from Fourier initial conditions (`fourier;5;true;true`) after 500 warm-up steps discarded onto the attractor. We store 1000 training and 50 validation trajectories of 51 frames, and 50 test trajectories of 501 frames, with disjoint seeds.

The stored spacing is $\Delta t = 0.2$, twice the APEBench default. We obtain it by scaling *both* `dt` and `num_substeps` by two, so the inner solver step is unchanged and only the sampling is coarsened (Table 2). Kolmo is forced 2D Navier–Stokes on $[0, 2\pi]^2$ with injection at mode 4 (scale 1.0), diffusivity 10^{-2} and drag -0.1. KS is solved on $[0, 60]^2$ in the $a = (0, 0, -1, 0, -1)$, $b = (0, 0, -1)$ form.

Table 2: Generation parameters. `dt` and `num_substeps` are scaled together, leaving the inner solver step at the APEBench default.

	Kolmo	KS
Domain L	2π	60
Resolution	160×160	160×160
Stored Δt	0.2	0.2
`num_substeps`	40	2
Inner solver step	0.005	0.1

1D Kuramoto–Sivashinsky. The 1D Kuramoto–Sivashinsky (KS1D) systems are generated with APEBench (Koehler et al., 2024) in physical mode using $v_t + v\, v_x = -v_{xx} - v_{xxxx}$ (coefficients $a = (0, 0, -1, 0, -1)$ and convection -1) on periodic domains $[0, L]$ with $L \in \{50, 100, 200, 300, 400\}$ and 512 grid points. Trajectories start from Fourier initial conditions (`fourier;5;true;true`) after 500 warm-up steps on the attractor and are stored at $\Delta t = 0.6$ (`num_substeps`$= 6$, inner step 0.1).

We use the conservative variable $v = u_x$ rather than the height field u, whose variance is dominated by the lowest modes and whose D_{eff} saturates near 9. The 512-point grid remains resolved up to $L = 400$, where D_{eff} agrees with a 4096-point reference within 1%. Across the five domains, $D_{\mathrm{eff}} = 10.9,\ 24.2,\ 48.5,\ 72.5$, and 96.1, computed from the empirical spatial covariance.

We form 17-frame windows starting at frames 0, 17, and 34 from 1000 trajectories of 51 frames. Each of 32 random splits uses disjoint sets of 400 truth, 300 ensemble, and 300 donor trajectories.

2D Hasegawa–Wakatani. The Hasegawa–Wakatani model (Hasegawa & Wakatani, 1983) describes drift-wave turbulence at the edge of magnetized plasmas. It couples the density fluctuation n and electrostatic potential ϕ, with vorticity $\Omega = \nabla^2 \phi$. A background density gradient κ_n drives

the instability, while the adiabaticity parameter c_1 controls the coupling between n and ϕ. We use the reference implementation of Greif (2023):

$$\partial_t n = c_1(\phi - n) - [\phi, n] - \kappa_n \partial_y \phi - \nu(-\nabla^2)^N n,$$
$$\partial_t \Omega = c_1(\phi - n) - [\phi, \Omega] - \nu(-\nabla^2)^N \Omega,$$

with an Arakawa Poisson bracket, spectral Laplacian inversion, and RK4 time stepping. Parameters are the repository defaults ($c_1 = \kappa_n = 1$, $N = 3$, $\nu = 5 \times 10^{-8}$) on a periodic square of side $2\pi/0.15 \approx 41.9\,\rho_s$ at 512^2, with solver step 0.025.

The state has two channels, n and ϕ; Ω is recoverable and not stored. Each trajectory starts from independent white noise in n and Ω (amplitude 0.01), and 300 time units of spin-up are discarded. Frames are then stored every 10 solver steps ($\Delta t = 0.25$) and spectrally truncated to 160^2. As for the other systems, we store 1000 training, 50 validation (51 frames) and 50 test (501 frames) trajectories with disjoint seeds.

C.2 Architecture

Shared implementation and backbone. Both the Jumper J_θ and Completer C_ϕ are implemented in PyTorch with the same AViT-style spatial–temporal backbone inspired by MPP McCabe et al. (2024) and summarized in Table 3. A physical state is first projected channel-wise by a 1×1 convolution, embedded by a convolutional hMLP stem into D latent channels, processed by a stack of L space–time blocks, and decoded to the physical variables with an hMLP output head. Channel ordering is fixed for each physical system. Per-channel normalization statistics are computed only from the training split, cached, stored in the checkpoint, and reused unchanged for validation and test evaluation. The implementation supports distributed data-parallel training, automatic mixed precision, ensemble chunking, and gradient checkpointing of ensemble chunks.

Jumper architecture. Given context $c_t = x_{t-L_c+1:t}$, the Jumper represents $p(x_{t+H} \mid c_t)$. In the distributional model, the encoded context is augmented with one learned future-query token. For each ensemble member, a latent noise field z_m is projected to the model dimension and injected into this future query (additively in the experiments), after which the full space–time stack produces

$$\hat{x}^{(m)}_{t+H} = J_\theta(c_t, H, z_m).$$

Thus different latent draws produce different candidate endpoints while sharing the same observed context. The deterministic ablation removes the latent injection and uses the same backbone and forecasting interface with a single future query.

Completer architecture. The Completer represents $p(x_{t+1:t+H-1} \mid c_t, \tilde{x}_{t+H})$ and predicts the interior jointly rather than frame by frame. Tokens are ordered chronologically as

$$[c_t \mid q_1, \ldots, q_K \mid \tilde{x}_{t+H}],$$

where the experiments use the interior of the jump and each query corresponds to a fixed fractional lead $s_i = k_i/H$. Query i is initialized as

$$q_i = q_{\text{int}} + \phi(s_i) + (1 - s_i)h_t + s_i h_{t+H},$$

where q_{int} is a learned interior-token identity, ϕ is an MLP applied to a sinusoidal lead-time embedding, and h_t, h_{t+H} are the latent encodings of the last observed state and the conditioning endpoint. This latent interpolation provides a bridge between the boundary states, while the transformer learns the dynamical deviation from that bridge.

Latent injection and cross-time coupling. For each ensemble member m, the Completer associates a latent field $z_{m,i}$ with each interior query $i = 1, \ldots, K$. We generate these latents using the stationary AR(1) coupling

$$z_{m,1} = \epsilon_{m,1}, \qquad z_{m,i} = \rho z_{m,i-1} + \sqrt{1 - \rho^2}\,\epsilon_{m,i}, \qquad \epsilon_{m,i} \overset{\text{iid}}{\sim} \mathcal{N}(0, I).$$

Thus each $z_{m,i} \sim \mathcal{N}(0, I)$ for every ρ, while

$$\text{Cov}(z_{m,i}, z_{m,i+\ell}) = \rho^\ell I.$$

For additive injection, the same learned projection P is applied at every interior slot,

$$q_{m,i} \leftarrow q_{m,i} + \alpha P(z_{m,i}),$$

where α is the noise scale. Hence $\rho = 0$ gives independent per-query stochasticity, $\rho = 1$ gives a single latent shared across the entire completion, and intermediate values continuously interpolate between these cases while preserving all one-time latent marginals. The shared-latent setting $\rho = 1$ is implemented either by drawing a single latent field and reusing it across interior slots, or by the AR(1) recursion in equation 9 at $\rho = 1$; these coincide by construction, since $z_h = \rho z_{h-1} + \sqrt{1-\rho^2}\,\epsilon_h$ reduces to $z_h = z_{h-1}$ at $\rho = 1$. Some checkpoints were trained under the former implementation and evaluated under the latter, which we verified reproduces their sampling behaviour exactly.

In bridge mode the final token is the encoded, given endpoint plus a learned endpoint identity. For the forecast variant, an additional learned endpoint-query token allows the endpoint to be withheld and predicted jointly with the interior.

Table 3: Shared architecture for the Jumper and Completer.

Hyperparameter	Value
Patch size	8×8
Embedding dimension	384
Attention heads	12
Space–time blocks	8
Spatial / temporal attention	axial attention / attention
Relative positional bias	yes
Latent noise dimension	16

C.3 Objectives and Conditioning

Distributional objectives. The distributional Jumper is trained with the ES in the high-dimensional endpoint space, as in equation 3. The Completer uses the same proper scoring rule on the *joint path*: the K predicted interior states are treated as one structured target,

$$\mathcal{L}_{\text{comp}} = \text{ES}\Big(\{\hat{x}^{(m)}_{t+1:t+H-1}\}_{m=1}^{M}, x_{t+1:t+H-1}\Big).$$

Completer conditioning regimes. We implement the different regimes through the endpoint used on bridge steps, which is either the oracle future state x_{t+H} or a Jumper endpoint. In the latter case, 4 candidate endpoints are precomputed from a trained Jumper and the candidate nearest in ℓ_2 to the true x_{t+H} is selected, so that the conditioning endpoint remains compatible with the true interior segment used as supervision. This yields: (i) oracle bridge training, (ii) Jumper-conditioned bridge training, and (iii) endpoint-free forecast-only training. Validation uses the task appropriate to the trained model and otherwise retains the true endpoint for bridge evaluation, preventing a model from being selected merely because its validation conditioning matches a particular sampled-endpoint cache.

C.4 Optimization and Evaluation

Optimization. Both model families use AdamW (Loshchilov & Hutter, 2019); the training hyperparameters are summarized in Table 4. Weight decay is applied to matrix/tensor weights but not to one-dimensional parameters or learned token parameters. Training uses bf16 autocast, and the best checkpoint is selected by the configured validation metric. On HW2D, the Completer had not converged after 30k steps, so we train it for 50k steps.

Ensemble sizes. The stochastic Jumper objectives use $M = 8$ ensemble members during training and $M = 32$ during evaluation and the stochastic Completer uses $M = 4$ during training and $M = 16$ during evaluation.

Table 4: Training configuration. Batch sizes are per GPU.

Hyperparameter	Jumper	Completer
Optimizer	AdamW, $\beta = (0.9, 0.95)$	
Learning rate	3×10^{-4}	
Weight decay	0.05	
Schedule	1000-step warmup, cosine to 10^{-6}	
Gradient clipping	1.0	
Training steps	120k	30k (50k for HW2D)
Batch size	4	2
Training ensemble M	8	4
Interior states	–	15 (11 for HW2D)

C.5 BASELINES

To reduce architectural confounding, we implement the baselines using the same underlying backbone wherever compatible, while preserving their method-specific training and sampling mechanisms. This reduces differences arising from neural architecture and focuses the comparison on the forecasting formulation itself.

Joint-Forecast. JOINT-FORECASt is an endpoint-free control using the same joint forecasting backbone as the Completer. It generates multiple future states jointly from the observed context, analogous to temporal bundling (Brandstetter et al., 2022), but without using a predicted future boundary to constrain the interior. This isolates the effect of endpoint conditioning while retaining joint multi-time generation. We apply the same cross-time coupling family κ_ρ to its future-query latents, allowing us to test whether inference-time coupling remains consequential without JUMP–COMPLETE.

DYffusion. DYffusion (Rühling Cachay et al., 2023) also uses an interpolator–forecaster decomposition, but differs in how the intermediate path is represented and sampled. Our Completer directly models a joint conditional path distribution

$$p_{\phi,\kappa}(\hat{x}_{t+1:t+H-1} \mid c_t, \tilde{x}_{t+H}),$$

with an explicit coupling κ across the stochastic inputs associated with different forecast times. In contrast, DYffusion trains a stochastic time-conditioned interpolator for individual intermediate states,

$$p_\phi(\hat{x}_{t+i} \mid x_t, x_{t+H}, i),$$

using the true future endpoint x_{t+H} during interpolation training. Its stochasticity is provided by Monte Carlo dropout, and repeated interpolator evaluations do not introduce an explicit cross-time coupling variable. With fresh dropout realizations across interpolation calls, this is closest to the independently resampled $\rho = 0$ end of our coupling family at the interpolator level. DYffusion's full trajectory law is nevertheless not a product of independent interpolations, since its inference procedure iteratively alternates interpolation and forecasting.

We tune DYffusion's stage-1 Monte Carlo dropout rate on validation CRPS, since this parameter controls predictive ensemble spread at inference. We sweep candidate dropout rates, then train one stage-2 forecaster for the selected interpolator. The selected rates are 0.2 for Kolmogorov, and 0.1 for 2D Kuramoto–Sivashinsky and 2D Hasegawa–Wakatani; all reported DYffusion results use these validation-selected settings.

PDE-Refiner. PDE-Refiner (Lippe et al., 2023) is an autoregressive neural PDE solver that refines an initial one-step prediction through a sequence of Gaussian denoising steps, with geometrically decreasing noise $\sigma_k = \sigma_{\min}^{k/K}$ targeting progressively smaller prediction errors. We use $K = 3$ refinement steps and sweep

$$\sigma_{\min} \in \left\{10^{-4}, \sqrt{2 \times 10^{-7}} \approx 4.47 \times 10^{-4}, 2 \times 10^{-3}\right\},$$

selecting $\sigma_{\min}$ based on validation ES. The choice $K = 3$ follows the ablation of Lippe et al. (2023), which shows a tradeoff between refinement depth and computational cost. Performance improves

with additional refinement steps, but each step requires an additional model evaluation, and the gains begin to flatten after three to four steps.

C.6 Model size and inference cost

We compare the computational cost of the methods in the Kolmogorov flow experiments. Table 5 reports the number of trainable parameters, while Table 6 reports inference cost in network function evaluations (NFE). The individual networks have similar parameter counts, so methods composed of two networks, Jump–Complete and DYffusion, have roughly twice as many parameters as Joint–Forecast and PDE-Refiner. In contrast, inference cost is largely determined by the generation procedure: methods that generate blocks of frames without iterative refinement require fewer evaluations per frame. Jump–Complete combines a relatively large parameter count with low inference NFE, whereas DYffusion has both a similar parameter count and substantially higher inference cost.

Table 5: Model parameter counts.

Method	Network parameter counts	Total
Jump–Complete	19,508,678 + 19,804,772	**39,313,450**
Joint–Forecast	19,805,156	19,805,156
DYffusion	19,649,990 + 19,650,404	39,300,394
PDE-Refiner	19,649,994	19,649,994

Table 6: Inference network evaluations for Kolmogorov Flow. Relative cost is normalized to Jump–Complete.

Method	NFE per rollout step	Frames generated	NFE per frame	Relative cost
Joint–Forecast	1	16	0.0625	$0.5\times$
Jump–Complete	$1 + 1 = 2$	16	0.125	$1\times$
DYffusion	$16 + 29 + 1 = 46$	16	2.875	$23\times$
PDE-Refiner	$3 + 1 = 4$	1	4.000	$32\times$

D ENERGY-DISTANCE EXPANSION

The leading coefficient in Lemma 1 coincides with the second-order covariance term of Yan & Zhang (2023, Eq. (9)), obtained there asymptotically under a bounded-spectrum factor model; Chakraborty & Zhang (2021, Lem. 4.1, Thm. 4.1) only show that the first-order term vanishes under equal means and traces. Lemma 1 removes these assumptions and gives an explicit non-asymptotic remainder, which is what allows Proposition 1 to be stated in terms of $D_{\mathrm{eff},x}$.

Lemma 1 (General energy-distance decomposition)**.** *Let P and Q be zero-mean distributions with finite sixth moments, covariances Σ_P, Σ_Q, and equal total variance*

$$t = \operatorname{tr}\Sigma_P = \operatorname{tr}\Sigma_Q.$$

Then

$$D_E(P,Q) = \frac{\|\Sigma_P - \Sigma_Q\|_F^2}{2(2t)^{3/2}} + R, \tag{13}$$

where, for

$$\zeta_{PQ} = \|X - Y\|^2, \qquad \zeta_{PP} = \|X - X'\|^2, \qquad \zeta_{QQ} = \|Y - Y'\|^2,$$

the remainder satisfies

$$|R| \le \frac{3}{8(2t)^{5/2}} \left(2\mathbb{E}|\zeta_{PQ} - 2t|^3 + \mathbb{E}|\zeta_{PP} - 2t|^3 + \mathbb{E}|\zeta_{QQ} - 2t|^3\right). \tag{14}$$

Proof. We first derive a generic expansion. Let $\zeta \ge 0$ be any random variable with mean

$$m := \mathbb{E}\zeta > 0,$$

and define

$$x = \frac{\zeta - m}{m}.$$

Then $\mathbb{E}x = 0$, $x \ge -1$, and

$$\sqrt{\zeta} = \sqrt{m}\sqrt{1+x}.$$

Write

$$\sqrt{1+x} = 1 + \frac{x}{2} - \frac{x^2}{8} + h(x), \tag{15}$$

where

$$h(x) = \sqrt{1+x} - 1 - \frac{x}{2} + \frac{x^2}{8}.$$

To bound the remainder, set $u = \sqrt{1+x} \ge 0$, so that $x = u^2 - 1$. A direct calculation gives

$$h(x) = \frac{(u-1)^3(u+3)}{8}, \qquad x^3 = (u-1)^3(u+1)^3.$$

Hence

$$|h(x)| \le \frac{3}{8}|x|^3, \qquad x \ge -1.$$

Since $\mathbb{E}x = 0$ and

$$\mathbb{E}x^2 = \frac{\mathbb{E}(\zeta - m)^2}{m^2} = \frac{\operatorname{Var}(\zeta)}{m^2},$$

multiplying equation 15 by $\sqrt{m}$ and taking expectations yields

$$\mathbb{E}\sqrt{\zeta} = \sqrt{m} - \frac{\operatorname{Var}(\zeta)}{8m^{3/2}} + r_\zeta, \tag{16}$$

with

$$r_\zeta = \sqrt{m}\,\mathbb{E}h\left(\frac{\zeta - m}{m}\right), \qquad |r_\zeta| \le \frac{3}{8}\frac{\mathbb{E}|\zeta - m|^3}{m^{5/2}}. \tag{17}$$

We now apply this generic expansion to the three squared distances ζ_{PQ}, ζ_{PP}, and ζ_{QQ}. For example,

$$\mathbb{E}\zeta_{PQ} = \mathbb{E}\|X - Y\|^2 = \mathbb{E}\|X\|^2 + \mathbb{E}\|Y\|^2 - 2\mathbb{E}[X^\top Y].$$

Independence and centering give $\mathbb{E}[X^\top Y] = 0$, while

$$\mathbb{E}\|X\|^2 = \operatorname{tr}\Sigma_P, \qquad \mathbb{E}\|Y\|^2 = \operatorname{tr}\Sigma_Q.$$

Therefore

$$\mathbb{E}\zeta_{PQ} = \operatorname{tr}\Sigma_P + \operatorname{tr}\Sigma_Q = 2t.$$

The same argument gives

$$\mathbb{E}\zeta_{PP} = 2\operatorname{tr}\Sigma_P = 2t, \qquad \mathbb{E}\zeta_{QQ} = 2\operatorname{tr}\Sigma_Q = 2t.$$

Thus all three squared distances have the common mean $m = 2t$ in equation 16.

Let

$$v_{AB} := \operatorname{Var}(\zeta_{AB}).$$

Substitution into the definition of the energy distance gives

$$D_E(P, Q) = \frac{v_{PP} + v_{QQ} - 2v_{PQ}}{8(2t)^{3/2}} + R, \tag{18}$$

where

$$R = 2r_{PQ} - r_{PP} - r_{QQ}.$$

The bound equation 14 follows immediately from equation 17.

It remains to simplify the variance combination. Define

$$\kappa_P = \operatorname{Var}(\|X\|^2), \qquad \kappa_Q = \operatorname{Var}(\|Y\|^2).$$

Using independence and centering,

$$v_{PQ} = \kappa_P + \kappa_Q + 4\operatorname{tr}(\Sigma_P\Sigma_Q).$$

Similarly,

$$v_{PP} = 2\kappa_P + 4\operatorname{tr}(\Sigma_P^2), \qquad v_{QQ} = 2\kappa_Q + 4\operatorname{tr}(\Sigma_Q^2).$$

Therefore the fourth-moment terms cancel:

$$\begin{aligned} v_{PP} + v_{QQ} - 2v_{PQ} &= 4\left[\operatorname{tr}(\Sigma_P^2) + \operatorname{tr}(\Sigma_Q^2) - 2\operatorname{tr}(\Sigma_P\Sigma_Q)\right] \\ &= 4\|\Sigma_P - \Sigma_Q\|_F^2. \end{aligned}$$

Substituting into equation 18 proves equation 13. □

The cancellation above is distribution-free: neither Gaussianity nor separability is needed for the covariance term. The remainder contains the contribution not captured by this second-order covariance term.

Proof of Proposition 1. For the Gaussian trajectories in Section 3,

$$\Sigma_P = \Sigma_{x,d} \otimes \Sigma_t^P, \qquad \Sigma_Q = \Sigma_{x,d} \otimes \Sigma_t^Q.$$

The equal-trace assumption gives

$$\operatorname{tr}\Sigma_P = \operatorname{tr}\Sigma_Q = t_d = \operatorname{tr}\Sigma_{x,d}\operatorname{tr}\Sigma_t^P.$$

Using $\|A \otimes B\|_F = \|A\|_F\|B\|_F$,

$$\|\Sigma_P - \Sigma_Q\|_F^2 = \|\Sigma_{x,d}\|_F^2\|\Sigma_t^P - \Sigma_t^Q\|_F^2 \tag{19}$$

$$= \frac{(\operatorname{tr}\Sigma_{x,d})^2}{D_{\mathrm{eff},x}}\,\eta_t^2(\operatorname{tr}\Sigma_t^P)^2 \tag{20}$$

$$= \frac{\eta_t^2 t_d^2}{D_{\mathrm{eff},x}}. \tag{21}$$

Hence Lemma 1 gives

$$\frac{1}{\sqrt{2t_d}}\frac{\|\Sigma_P-\Sigma_Q\|_F^2}{2(2t_d)^{3/2}}=\frac{\eta_t^2}{8D_{\mathrm{eff},x}}. \tag{22}$$

It remains to bound the remainder. If $Z\sim N(0,A)$, write $Z=A^{1/2}g$ with $g\sim N(0,I)$; then $\|Z\|^2-\operatorname{tr}A=g^\top Ag-\operatorname{tr}A$ is a centred polynomial of degree two in g with second moment $2\operatorname{tr}A^2$. By Gaussian hypercontractivity (Diakonikolas et al., 2010, Thm. 2.1), $\|p\|_q\le(q-1)^{d/2}\|p\|_2$ for any degree-d Gaussian polynomial and $q>2$, so with $q=3$, $d=2$,

$$\mathbb{E}\big|\|Z\|^2-\operatorname{tr}A\big|^3\le 8\,(2\operatorname{tr}A^2)^{3/2}=2^{9/2}\,(\operatorname{tr}A^2)^{3/2}. \tag{23}$$

For the three squared distances,

$$X-Y\sim\mathcal{N}\Big(0,\,\Sigma_{x,d}\otimes(\Sigma_t^P+\Sigma_t^Q)\Big),$$

and

$$X-X'\sim\mathcal{N}\big(0,\,2\Sigma_{x,d}\otimes\Sigma_t^P\big),\qquad Y-Y'\sim\mathcal{N}\Big(0,\,2\Sigma_{x,d}\otimes\Sigma_t^Q\Big).$$

Thus each covariance has the form

$$A=\Sigma_{x,d}\otimes B,$$

where B is fixed as d varies. Consequently,

$$\operatorname{tr}A=(\operatorname{tr}\Sigma_{x,d})(\operatorname{tr}B),\qquad \operatorname{tr}A^2=\|\Sigma_{x,d}\|_F^2\|B\|_F^2.$$

Combining equation 17 and equation 23, and normalizing by $\sqrt{2t_d}$, gives

$$\frac{|r_\zeta|}{\sqrt{2t_d}}\le C_B\frac{\|\Sigma_{x,d}\|_F^3}{(\operatorname{tr}\Sigma_{x,d})^3}=C_BD_{\mathrm{eff},x}^{-3/2},$$

where C_B depends only on the fixed temporal covariance B. Since

$$R=2r_{PQ}-r_{PP}-r_{QQ},$$

it follows that

$$\frac{|R|}{\sqrt{2t_d}}=\mathcal{O}\Big(D_{\mathrm{eff},x}^{-3/2}\Big).$$

Combining this with equation 22 yields

$$\frac{D_E(P_d,Q_d)}{\sqrt{2t_d}}=\frac{\eta_t^2}{8D_{\mathrm{eff},x}}+\mathcal{O}\Big(D_{\mathrm{eff},x}^{-3/2}\Big),$$

which proves Proposition 1. □

E Derivation of the detection budget

Detection budget. For N independent forecast cases $y_n\sim P$, let

$$d_n=\mathrm{ES}_M(Q,y_n)-\mathrm{ES}_M(P,y_n),$$

with independent M-member ensembles from P and Q. Since the finite-ensemble ES estimator is unbiased,

$$\mu:=\mathbb{E}d_n=\tfrac{1}{2}D_E(P,Q).$$

A one-sided paired test at level α and power $1-\beta$ requires

$$N^\star\approx\frac{z^2\sigma_d^2}{\mu^2},\qquad z=z_{1-\alpha}+z_{1-\beta}, \tag{24}$$

with $\sigma_d^2=\mathrm{Var}(d_n)$ (Diebold & Mariano, 2002). We decompose

$$\sigma_d^2=\mathbb{E}_y[\mathrm{Var}(d_n\mid y)]+\mathrm{Var}_y(\mathbb{E}[d_n\mid y])=\sigma_{\mathrm{ens}}^2+\sigma_{\mathrm{tgt}}^2. \tag{25}$$

We analyze these two contributions separately, first quantifying the finite-ensemble noise σ_{ens}^2 and then the target-to-target variability σ_{tgt}^2.

Ensemble noise. To quantify σ^2_{ens}, we condition on the target y, so that the remaining randomness in d_n comes only from finite ensemble sampling, compute $\mathrm{Var}(d_n \mid y)$, and then average over $y \sim P$.

Fix y and let $X_1, \ldots, X_M \sim Q$. Since

$$\|X_m - y\| = \sqrt{\|X_m\|^2 + \|y\|^2 - 2X_m^\top y},$$

and

$$\mathbb{E}\|X_m - y\|^2 = 2t,$$

linearizing around $2t$ gives

$$\|X_m - y\| \approx \sqrt{2t} + \frac{\|X_m\|^2 + \|y\|^2 - 2X_m^\top y - 2t}{2\sqrt{2t}}.$$

Thus, conditional on y, the X_m-dependent contribution to the observation-distance term is

$$\frac{\|X_m\|^2 - 2X_m^\top y}{2M\sqrt{2t}}.$$

For the pairwise term, linearizing each distance around $2t$ in the same way gives

$$\|X_m - X_n\| \approx \sqrt{2t} + \frac{\|X_m\|^2 + \|X_n\|^2 - 2X_m^\top X_n - 2t}{2\sqrt{2t}}.$$

Each index appears in $2(M-1)$ ordered pairs, so the radial part of $-\frac{1}{2M(M-1)} \sum_{m \neq n} \|X_m - X_n\|$ contributes

$$-\frac{\|X_m\|^2 - t}{2M\sqrt{2t}}$$

per member, which cancels the radial fluctuation of the observation-distance term at first order. What remains is

$$-\frac{1}{M\sqrt{2t}} \sum_m X_m^\top y \; + \; \frac{1}{2M(M-1)\sqrt{2t}} \sum_{m \neq n} X_m^\top X_n.$$

The two sums are uncorrelated, and $\mathrm{Var}\big(\sum_{m \neq n} X_m^\top X_n\big) = 2M(M-1)\,\mathrm{tr}\,\Sigma_Q^2$. Hence

$$\mathrm{Var}\big(\mathrm{ES}_M(Q, y) \mid y\big) \approx \frac{y^\top \Sigma_Q y}{2tM} + \frac{\mathrm{tr}\,\Sigma_Q^2}{4tM(M-1)}.$$

Adding the independent P-ensemble and averaging over $y \sim P$ gives, to leading order in the covariance perturbation,

$$\sigma^2_{\text{ens}} \approx \frac{\mathrm{tr}(\Sigma_P \Sigma_Q) + \mathrm{tr}\,\Sigma_P^2}{2tM} + \frac{\mathrm{tr}\,\Sigma_P^2 + \mathrm{tr}\,\Sigma_Q^2}{4tM(M-1)} \approx c_M \, \frac{\mathrm{tr}\,\Sigma_P^2}{tM}, \qquad c_M = \frac{2M-1}{2(M-1)}.$$

The factor $c_M \in (1, 3/2]$ tends to one as $M \to \infty$; $c_{16} \approx 1.03$.

Target variability. The second contribution in equation 25,

$$\sigma^2_{\text{tgt}} = \mathrm{Var}_{y \sim P}(\mathbb{E}[d_n \mid y]),$$

vanishes when $Q = P$. For a smooth covariance perturbation $\Sigma_Q = \Sigma_P + \Delta\Sigma$, with perturbation magnitude η_t, the conditional mean score difference satisfies

$$\mathbb{E}[d_n \mid y] = O(\eta_t)$$

in $L^2(P)$, and hence

$$\sigma^2_{\text{tgt}} = O(\eta_t^2).$$

By contrast, the ensemble contribution derived above has a nonzero $O(M^{-1})$ limit as $\eta_t \to 0$. Therefore, for fixed finite M and sufficiently small covariance perturbations,

$$\sigma_d^2 = \sigma^2_{\text{ens}}\{1 + o(1)\}, \qquad \eta_t \to 0,$$

which is the regime used in the detection-budget scaling below. The approximation is not uniform in M: as $M \to \infty$, the ensemble contribution vanishes and target-to-target variability eventually determines the detection variance.

Separable case. For

$$\Sigma_P = \Sigma_{x,d} \otimes \Sigma_t^P,$$

$$\operatorname{tr} \Sigma_P^2 = \frac{t_d^2}{D_{\mathrm{eff},x} D_{\mathrm{eff},t}},$$

so

$$\sigma_{\mathrm{ens}}^2 \approx \frac{c_M \, t_d}{M D_{\mathrm{eff},x} D_{\mathrm{eff},t}},$$

Proposition 1 gives

$$\mu \approx \frac{\sqrt{2t_d}\, \eta_t^2}{16 D_{\mathrm{eff},x}},$$

and therefore

$$N^\star \approx \frac{128\, c_M \, z^2 D_{\mathrm{eff},x}}{M D_{\mathrm{eff},t}\, \eta_t^4},$$

which is equation 5. Thus the detection budget grows linearly with effective spatial dimension: the squared ES signal decays as $D_{\mathrm{eff},x}^{-2}$, while the finite-ensemble variance decays only as $D_{\mathrm{eff},x}^{-1}$.

For a small relative variance error ϵ,

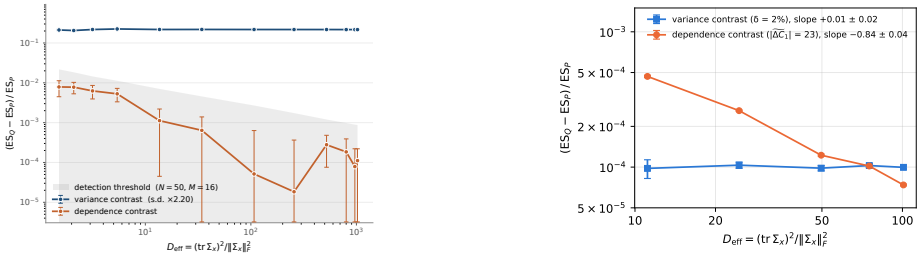
$$\mu_{\mathrm{var}} \approx \frac{\sqrt{2t}\, \epsilon^2}{32},$$

giving

$$N^\star_{\mathrm{var}} \approx \frac{512\, c_M \, z^2}{D_{\mathrm{eff},x} D_{\mathrm{eff},t} M \epsilon^4}.$$

E.1 Empirical validation

Figure 6 tests the scaling predicted by Proposition 1 and equation 5, both under the separable-Gaussian assumptions and in nonlinear dynamics. For both the Gaussian control and 1D Kuramoto–Sivashinsky (KS1D), the temporal-dependence ES signal decreases with $D_{\mathrm{eff},x}$ similarly to the predicted inverse scaling, while the matched variance signal remains essentially flat. In KS1D, this behavior persists over $D_{\mathrm{eff},x} = 11$–96, showing empirically that the dimension-dependent loss of temporal resolution extends beyond the Gaussian setting. Our larger systems have $D_{\mathrm{eff},x} = 44.3$ for Kolmogorov flow, 78.4 for 2D Kuramoto–Sivashinsky, and 287.4 for Hasegawa–Wakatani. At these dimensions, equation 5 predicts detection budgets far above the available $N = 50$, making this effect relevant to the experiments below.



(a) Gaussian control **(b)** 1D Kuramoto–Sivashinsky (KS1D)

Figure 6: **ES resolution of temporal dependence degrades with effective spatial dimension.** **(a)** Gaussian trajectories: relative ES separation for a temporal-correlation perturbation and a KL-matched variance perturbation. Gray: empirical detection threshold, i.e. the smallest gap resolvable with $N = 50$ cases and $M = 16$ (Equation 5). Mean $\pm$ s.e. over 16 replicates. **(b)** KS1D: true 17-frame windows at $L \in \{50, 100, 200, 300, 400\}$ ($D_{\mathrm{eff}} = 11$–96). The variance contrast scales fields by 2%; the dependence contrast swaps frames with an independent true trajectory, preserving one-time statistics, at matched $|\widetilde{\Delta C}_1| = 23$. Mean $\pm$ s.e. over 32 trajectory splits.

F ADDITIONAL EXPERIMENTS

This appendix provides additional experiments supporting the training- and sampling-time results in the main text. We first examine the finite-ensemble effect in dependence-sensitive training. We then present controls and robustness tests for inference-time temporal coupling, including state-space diagnostics, alternative temporal criteria, and architectural and temporal-resolution ablations. Finally, we report complete long-horizon results and additional spectral and climatological comparisons.

F.1 FINITE-ENSEMBLE VARIOGRAM TRAINING ON HW2D

We repeat the finite-ensemble estimator experiment of Section 5.2 on HW2D. We train with ensemble size $M = 4$ and vary $\lambda \in \{0, 0.3, 1, 3\}$ over three seeds, using either the plug-in or the unbiased U-statistic estimator of the squared variogram residual. The two variants share the $\lambda = 0$ ES-only baseline, and VS_t is evaluated with the unbiased estimator for all models.

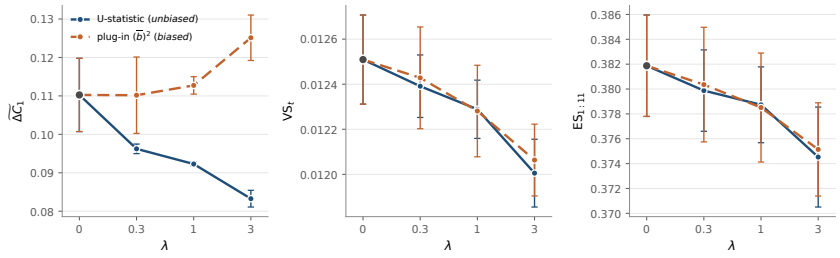


Figure 7: **Finite-ensemble estimator effect on HW2D.** Same setup as Figure 3, with $\lambda \leq 3$ and $H = 12$. **Left:** normalized lag-one correlation error. **Middle:** VS_t, evaluated with the unbiased estimator. **Right:** joint ES. Error bars show mean $\pm$ s.d. across three seeds.

Figure 7 shows that the effect of the finite-ensemble estimator is not specific to Kolmogorov flow. As λ increases, both estimators reduce VS_t and joint ES by nearly identical amounts, yet their effects on temporal dependence differ. The unbiased estimator moves $\widetilde{\Delta C}_1$ toward zero, whereas the plug-in estimator tends to move it away from zero, although this trend remains within seed-to-seed variability.

F.2 EFFECT OF TRAINING ENSEMBLE SIZE

Equation 11 shows that the plug-in variogram estimator adds a finite-ensemble dispersion term proportional to $1/M$. This bias vanishes as $M \to \infty$, whereas the U-statistic is unbiased for every M. We therefore repeat the $\lambda = 10$ experiment across training ensemble sizes $M \in \{2, 4, 8, 16\}$.

Table 7: **The estimator gap decreases with training ensemble size.** Kolmogorov single-jump results with Jumper-generated endpoints and $\lambda = 10$, over three seeds (mean $\pm$ s.d.). The final column is the absolute difference in $\widetilde{\Delta C}_1$ between the plug-in and U-statistic models.

	plug-in $(\bar{b})^2$		U-statistic		
training M	$\mathrm{ES}_{1:15}^{\mathrm{joint}}$	$\widetilde{\Delta C}_1$	$\mathrm{ES}_{1:15}^{\mathrm{joint}}$	$\widetilde{\Delta C}_1$	$\lvert\Delta_{\mathrm{est}}\rvert$
2	0.3195 ± 0.0011	$+0.0586 \pm 0.0018$	0.3500 ± 0.0047	$+0.0367 \pm 0.0028$	0.0218
4	0.3180 ± 0.0052	$+0.0509 \pm 0.0009$	0.3250 ± 0.0050	$+0.0273 \pm 0.0022$	0.0236
8	0.3147 ± 0.0056	$+0.0399 \pm 0.0033$	0.3156 ± 0.0053	$+0.0254 \pm 0.0012$	0.0145
16	0.3103 ± 0.0034	$+0.0280 \pm 0.0049$	0.3115 ± 0.0023	$+0.0215 \pm 0.0029$	0.0065

Table 7 supports the finite-ensemble interpretation. The difference between the plug-in and U-statistic models decreases overall as the training ensemble grows, from 0.0218 at $M = 2$ to 0.0065 at $M = 16$, although it increases slightly between $M = 2$ and $M = 4$. Most of the reduction comes from the plug-in model, whose $\widetilde{\Delta C}_1$ decreases from 0.0586 to 0.0280, while the U-statistic model changes from 0.0367 to 0.0215. This is consistent with the plug-in bias weakening as M increases, while the U-statistic remains unbiased. We do not expect the downstream model gap to follow an

exact $1/M$ rate, since equation 11 characterizes the training objective rather than the response of the optimized model. At $M = 16$, the remaining difference is comparable to the seed-to-seed variation.

F.3 State-space controls under inference-time coupling

The construction in Section 4.2 preserves the one-time latent marginal exactly for every inference coupling, $z_h \sim \mathcal{N}(0, I)$. For a nonlinear joint generator, however, this does not imply that the corresponding one-time output distributions are invariant to ρ_{inf}. We therefore evaluate one-time and state-space diagnostics over the same inference-time coupling sweep as in the Kolmogorov experiments of Figure 4, with the learned weights held fixed.

In addition to state variance and the resolved spatial power spectrum, we report a one-time ES obtained by evaluating each forecast lead separately and then averaging over the 15 interior leads,

$$\mathrm{ES}_{1:15}^{\mathrm{1time}} = \frac{1}{15}\sum_{h=1}^{15} \mathrm{ES}\left(\{\hat{x}_{t+h}^{(m)}\}_{m=1}^{M}, x_{t+h}\right). \tag{26}$$

Unlike the joint trajectory ES used in the main text, this quantity is evaluated independently at each lead and therefore does not directly measure cross-time dependence.

Table 8 shows that these one-time diagnostics vary only weakly across the coupling sweep. The mean one-time ES changes only marginally from independent to nearly shared latent sampling, while the state variance remains close to the reference value throughout the sweep. The resolved spatial power-spectrum ratio also changes only modestly with ρ_{inf}. Thus, the substantial effect of inference-time coupling on temporal correlation is not accompanied by a comparable change in the measured one-time predictive or state-space properties.

Table 8: **State-space controls for the inference-time coupling sweep.** Kolmogorov-flow results for fixed $\rho_{\mathrm{train}} = 1$ checkpoints over three seeds, with learned weights held fixed across rows. One-time state-space diagnostics vary only weakly with ρ_{inf}.

ρ	$\mathrm{ES}_{1:15}^{\mathrm{1time}} \downarrow$	$\mathrm{Var}/\mathrm{Var}_{\mathrm{ref}}$	$\overline{P_{\mathrm{gen}}/P_{\mathrm{ref}}}$
0	0.07912 ± 0.00098	1.0052 ± 0.0068	1.1805 ± 0.0417
0.5	0.07897 ± 0.00098	1.0051 ± 0.0052	1.1836 ± 0.0372
0.75	0.07898 ± 0.00102	1.0048 ± 0.0037	1.1827 ± 0.0316
0.9	0.07888 ± 0.00101	1.0041 ± 0.0025	1.1763 ± 0.0249
0.95	0.07877 ± 0.00097	1.0037 ± 0.0021	1.1717 ± 0.0219
0.99	0.07862 ± 0.00087	1.0032 ± 0.0016	1.1660 ± 0.0194

The spatial-spectrum ratio in Table 8 is averaged over resolved modes $1 \le k \le 24$; higher modes are excluded because the reference power approaches the numerical floor. The fact that this ratio remains above one reflects a standing model bias, but importantly that bias changes little as ρ_{inf} varies.

Taken together, these controls support interpreting ρ_{inf} primarily as a mechanism for changing how generated states are organized across time, rather than as one that substantially changes the measured one-time state-space distribution.

F.4 Cross-time coupling versus latent amplitude

Latent amplitude is the other inference-time knob acting on the stochastic inputs with learned weights fixed. We compare the two on Kolmogorov flow using the $\rho_{\mathrm{train}} = 0$ checkpoint. For the coupling sweep, the noise scale α is held at its training value; for the amplitude sweep, we fix $\rho = 0$, the regime in which both knobs act on the same independently resampled per-frame noise.

Changing the latent amplitude introduces a clear tradeoff between temporal correlation and probabilistic fidelity (Figure 8). Reducing the amplitude moves $\widetilde{\Delta C}_1$ toward zero, but only partially, while joint ES deteriorates sharply, rising from about 0.322 at α to 0.43 at 0.25α. By contrast, adjusting the cross-time coupling ρ can bring $\widetilde{\Delta C}_1$ through zero while leaving joint ES nearly unchanged. This highlights the advantage of coupling calibration over amplitude rescaling for correcting temporal dependence.

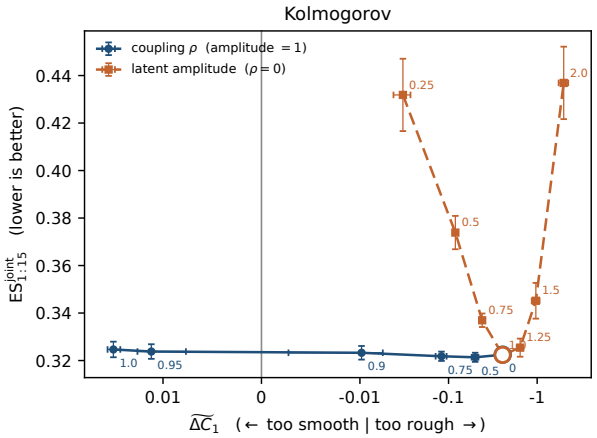


Figure 8: **Cross-time coupling isolates temporal dependence more cleanly than latent-amplitude scaling.** Kolmogorov completion forecasts under two inference-time interventions. The blue curve varies cross-time coupling ρ at fixed latent amplitude 1, whereas the orange curve varies latent amplitude at fixed $\rho = 0$. Labels indicate the corresponding value of ρ or latent amplitude. Increasing ρ moves the normalized lag-one correlation error $\widetilde{\Delta C}_1$ toward zero while joint Energy Score remains nearly unchanged. In contrast, changing the latent amplitude strongly affects both temporal correlation and joint ES, which deteriorates away from the training amplitude. The open marker denotes the shared configuration ($\rho = 0$, amplitude $= 1$). Thus, modifying cross-time dependence provides a direction for temporal calibration while preserving the one-time latent marginals, unlike rescaling the latent noise.

F.5 Metric-dependent temporal calibration

The main experiments use the temporal Variogram Score VS_t to select the inference-time coupling. However, no single metric fully characterizes temporal fidelity. Different temporal diagnostics capture different aspects of the dynamics and may therefore favor different couplings.

To make this explicit, Table 9 compares three complementary short-time diagnostics. The first is the raw lag-one state-correlation error ΔC_1. For a local-dynamics diagnostic, define the temporal increment

$$\Delta x_t = x_{t+1} - x_t, \tag{27}$$

and its lag-one correlation error

$$\Delta C_1^{\Delta} = \mathrm{Corr}_{\mathrm{gen}}\left(\Delta x_t, \Delta x_{t+1}\right) - \mathrm{Corr}_{\mathrm{ref}}\left(\Delta x_t, \Delta x_{t+1}\right). \tag{28}$$

Whereas ΔC_1 measures persistence of the states themselves, ΔC_1^{Δ} measures persistence or reversal of successive temporal changes.

We additionally consider the second temporal difference

$$\Delta^2 x_t = x_{t+2} - 2x_{t+1} + x_t, \tag{29}$$

which probes local temporal curvature. Its generated and reference distributions are compared using

$$W_{1,1}^{(2)} = W_1\left(\mathcal{L}_{\mathrm{gen}}(\Delta^2 x_t), \mathcal{L}_{\mathrm{ref}}(\Delta^2 x_t)\right), \tag{30}$$

with smaller values indicating closer agreement of the empirical second-difference distributions.

The diagnostics in Table 9 prefer different inference-time couplings. The lag-one state-correlation discrepancy is minimized at $\rho_{\mathrm{inf}} = 0.9$, the increment-correlation error at $\rho_{\mathrm{inf}} = 0.95$, and the second-difference Wasserstein discrepancy under fully shared sampling, $\rho_{\mathrm{inf}} = 1$. Thus, the experiments do not identify a single coupling that is optimal across all temporal diagnostics. Instead, the preferred value of ρ_{inf} depends on which aspect of temporal structure is chosen as the calibration target.

Table 9: **Different temporal diagnostics prefer different couplings.** The lag-one state-correlation discrepancy is minimized at $\rho_{\mathrm{inf}} = 0.9$, the increment-correlation discrepancy at $\rho_{\mathrm{inf}} = 0.95$, and the second-difference Wasserstein discrepancy at $\rho_{\mathrm{inf}} = 1$.

ρ_{inf}	ΔC_1 ($\lvert\cdot\rvert\downarrow$)	ΔC_1^{Δ} ($\lvert\cdot\rvert\downarrow$)	$W_{1,1}^{(2)}\downarrow$
0	-0.04234 ± 0.00382	-0.4946 ± 0.0263	0.05595 ± 0.00269
0.5	-0.01969 ± 0.00195	-0.2591 ± 0.0172	0.03673 ± 0.00194
0.75	-0.00778 ± 0.00099	-0.1262 ± 0.0106	0.02381 ± 0.00141
0.9	$\mathbf{-0.00068 \pm 0.00042}$	-0.0391 ± 0.0056	0.01327 ± 0.00084
0.95	$+0.00157 \pm 0.00027$	$\mathbf{-0.0090 \pm 0.0036}$	0.00882 ± 0.00064
0.99	$+0.00325 \pm 0.00021$	$+0.0152 \pm 0.0021$	0.00475 ± 0.00042
1.0	$+0.00363 \pm 0.00019$	$+0.0212 \pm 0.0017$	$\mathbf{0.00365 \pm 0.00036}$

We therefore use VS_t in the main experiments as a dependence-sensitive calibration criterion, rather than as a complete characterization of path fidelity. More generally, ρ_{inf} should be viewed as a post-training calibration degree of freedom whose preferred value depends on the temporal property one aims to match.

F.6 JUMP–COMPLETE ENDPOINT-CONDITIONING ABLATIONS

The endpoint plays two distinct roles in Jump–Complete: it provides information about the distant future at inference, and it defines the conditioning distribution seen by the Completer during training. We study both roles with the three variants in Table 10. As a first control, the *Joint-Forecast* model predicts the whole future block jointly from the observed context, without conditioning on a predicted future boundary. The two Jump–Complete variants instead first sample an endpoint with the Jumper and condition the Completer on that endpoint. They differ only in training: one sees oracle endpoints, while the other sees truth-compatible Jumper endpoints.

Table 10: Single-jump completion performance before chaining. For the Completer models, results are evaluated using the Jumper endpoint, whereas the forecast-only baseline uses its self-predicted endpoint. $\mathrm{ES}^{\mathrm{joint}}_{1:15}$ is computed over the 15 interior frames as a single joint target (lower is better). $\widetilde{\Delta C}_1$ is the generated-minus-reference lag-one correlation over the completed interior (closer to zero is better). Results are mean $\pm$ s.d. over three seeds.

	Kolmo		*KS*	
trained on	$\mathrm{ES}^{\mathrm{joint}}_{1:15}\downarrow$	$\widetilde{\Delta C}_1$	$\mathrm{ES}^{\mathrm{joint}}_{1:15}\downarrow$	$\widetilde{\Delta C}_1$
Jumper+Completer$_{\text{jumper}}$	0.3316 ± 0.0020	$+0.04051 \pm 0.00195$	0.0332 ± 0.0005	-0.00690 ± 0.00056
Jumper+Completer$_{\text{oracle}}$	0.3848 ± 0.0019	$+0.05054 \pm 0.00306$	0.0349 ± 0.0005	-0.00673 ± 0.00042
Joint-Forecast	0.4754 ± 0.0121	$+0.07558 \pm 0.00241$	0.0539 ± 0.0031	-0.01054 ± 0.00021

Is endpoint conditioning useful at inference? Explicitly predicting a distant endpoint and conditioning the interior on it improves both systems: $ES^{\mathrm{joint}}_{1:15}$ decreases from 0.4754 to 0.3316 on Kolmo and from 0.0539 to 0.0332 on Kuramoto–Sivashinsky. The normalized temporal-correlation error also moves closer to zero on both systems. Thus, the advantage of Jump–Complete is not merely that it generates several future states jointly; the predicted endpoint provides useful information for constructing the intervening trajectory.

Which endpoints should be used during training? At inference, both Completer variants receive Jumper-generated endpoints, so their comparison isolates the effect of the training conditioning distribution. Training only on oracle endpoints creates a mismatch with this deployment condition. On Kolmo, replacing oracle training with truth-compatible Jumper endpoints improves $ES^{\mathrm{joint}}_{1:15}$ from 0.3848 to 0.3316 and reduces $\widetilde{\Delta C}_1$ from 0.0505 to 0.0405. On Kuramoto–Sivashinsky, joint ES likewise improves from 0.0349 to 0.0332, while the temporal-correlation error is essentially unchanged at this precision. Thus, deployment-aware endpoint conditioning improves probabilistic completion quality, with its effect on temporal dependence varying across systems.

F.7 CROSS-TIME COUPLING IN THE ENDPOINT-FREE FORECAST MODEL

To test whether the coupling effect depends on endpoint conditioning, we repeat the fixed-weight inference sweep with the JOINT-FORECAST model. Unlike Jump–Complete, JOINT-FORECAST predicts the future block directly from the observed context. We apply the same marginal-preserving AR(1) coupling across future-query latents and vary only ρ_{inf}, keeping the learned weights fixed.

Figure 9 shows that inference coupling again strongly controls temporal dependence. On Kolmogorov flow, increasing ρ_{inf} moves the lag-one correlation from strongly under-correlated through the reference value, while the joint Energy Score changes comparatively little. On Kuramoto–Sivashinsky, the temporal correlation also moves monotonically toward the reference but remains under-correlated at $\rho_{\mathrm{inf}} = 1$; here the Energy Score becomes more sensitive at strong coupling. Thus, the effect of cross-time coupling persists in direct joint forecasting and is not specific to the Jump–Complete construction.

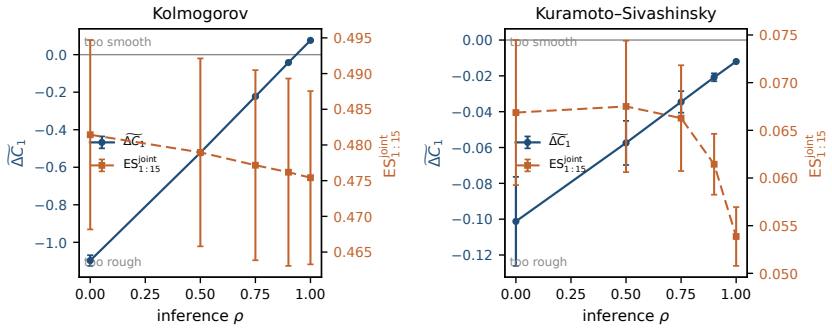


Figure 9: **Inference-time coupling controls temporal dependence in the endpoint-free JOINT-FORECAST model.** With the learned weights fixed, we vary only the AR(1) coupling ρ_{inf} across future-query latents. Blue curves show the normalized lag-one temporal-correlation error $\widetilde{\Delta C}_1$, for which zero is ideal, and orange curves show the joint Energy Score $ES^{\text{joint}}_{1:15}$. On Kolmogorov flow, increasing coupling moves the temporal correlation through the reference while joint ES varies comparatively weakly. On Kuramoto–Sivashinsky, temporal correlation moves consistently toward the reference but remains under-correlated at $\rho_{\text{inf}} = 1$, while joint ES favors stronger coupling. Error bars show mean $\pm$ s.d. across seeds.

F.8 POST-TRAINING COUPLING ACROSS TEMPORAL SAMPLING SCALES

To test whether inference-time coupling remains useful when the temporal sampling scale changes, we repeat the fixed-weight coupling sweep on a Kuramoto–Sivashinsky dataset sampled at $\Delta t = 0.8$, four times the spacing used in the main experiments.

Figure 10 shows that increasing ρ_{inf} consistently moves the generated lag-one correlation toward the reference for checkpoints trained with $\rho_{\text{train}} \in \{0, 0.5, 1\}$. The effect therefore persists under a substantially coarser temporal sampling scale. In particular, the same post-training intervention remains effective without changing the learned weights, supporting the interpretation of cross-time coupling as a calibration degree of freedom rather than a setting tied to a single temporal resolution.

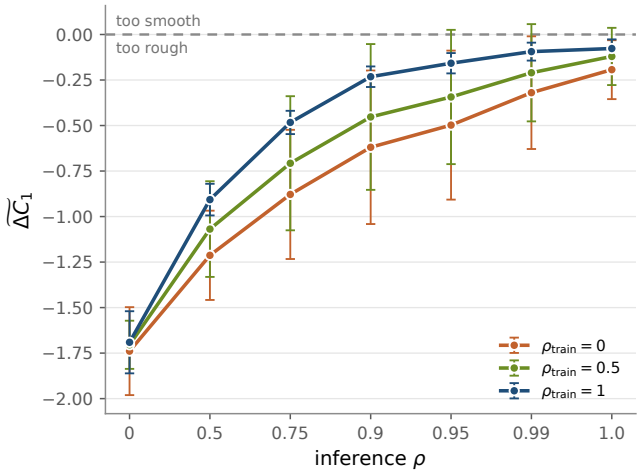


Figure 10: **Inference-time coupling remains effective at a coarser temporal sampling scale.** Kuramoto–Sivashinsky results for data sampled at $\Delta t = 0.8$, compared with $\Delta t = 0.2$ in the main experiments. For checkpoints trained with $\rho_{\text{train}} \in \{0, 0.5, 1\}$, we keep the learned weights fixed and vary only the inference-time coupling ρ_{inf}. Increasing ρ_{inf} consistently moves the normalized lag-one temporal-correlation error $\widetilde{\Delta C}_1$ toward the reference across all training couplings. Error bars show mean $\pm$ s.d. across seeds.

F.9 Interaction between training- and inference-time controls

Having established that temporal dependence remains adjustable after training, we next examine how dependence-sensitive training interacts with inference-time calibration. Figure 11 sweeps ρ_{inf} for models trained with and without the temporal Variogram Score, while Table 11 compares retraining against post-hoc calibration with learned weights fixed. Together, these experiments separate what is gained by changing the training objective from what can be recovered through the sampling process alone.

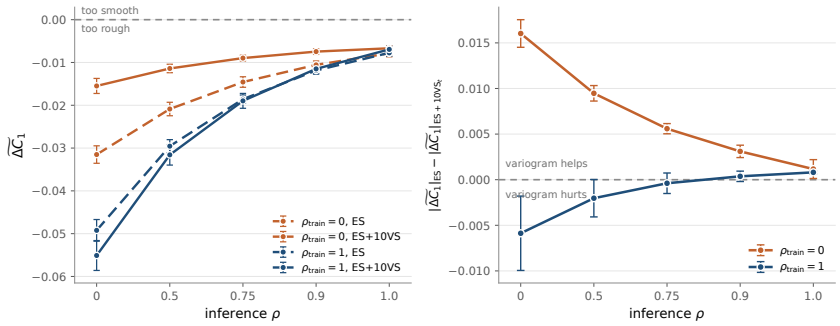


Figure 11: **Training- and sampling-time controls interact on KS. Left:** normalized lag-one correlation error across inference-time coupling for models trained with $\rho_{\text{train}} \in \{0, 1\}$, with and without the unbiased temporal variogram objective. **Right:** reduction in absolute lag-one error due to variogram training; positive values indicate improvement. Error bars show mean $\pm$ s.d. across seeds.

Figure 11 shows that sensitivity to the sampling coupling persists after dependence-sensitive retraining. On KS, increasing ρ_{inf} moves the lag-one correlation toward the reference throughout the tested range. Variogram retraining improves the lag-one match most at weaker coupling, but its advantage largely vanishes near the VS_t-selected operating point.

Kolmogorov provides a complementary regime in which the inference-coupling sweep crosses the reference correlation. Table 11 separates the effects of training- and sampling-time interventions by comparing an ES-trained checkpoint under native shared-latent sampling ($\rho_{\text{inf}} = 1$) and under the validation-selected coupling $\rho^\star = 0.95$, chosen using VS_t while keeping the learned weights fixed, with a variogram-trained checkpoint evaluated under its native sampling. The calibrated ES model reduces both $\widetilde{\Delta C}_1$ and $\widetilde{\Delta C}_2$ with little change in joint ES, while the variogram-trained model improves the same temporal diagnostics through retraining.

Table 11: Kolmo single-jump results over three seeds. The first two rows use identical ES-trained weights and differ only in the inference-time coupling. For the calibrated ES model, $\rho^\star$ is selected using validation VS_t and then frozen for test evaluation; all seeds select $\rho^\star = 0.95$. The variogram-trained model is reported under its native shared-latent sampling. Signed temporal-correlation discrepancies have ideal value zero. Values are mean $\pm$ s.d.

training	inference	$\text{ES}^{\text{joint}}_{1:15} \downarrow$	$\widetilde{\Delta C}_1$	$\widetilde{\Delta C}_2$
ES	$\rho_{\text{inf}} = 1$	0.3327 ± 0.0030	$+0.0440 \pm 0.0023$	$+0.0318 \pm 0.0018$
ES	calibrated $\rho^\star$	0.3320 ± 0.0038	$+0.0190 \pm 0.0032$	$+0.0239 \pm 0.0021$
ES + 10 VS_t	$\rho_{\text{inf}} = 1$	0.3250 ± 0.0050	$+0.0273 \pm 0.0022$	$+0.0178 \pm 0.0016$

Taken together, these results show that training- and sampling-time controls address the same temporal structure from different directions. Variogram training improves temporal dependence under the native sampling scheme, while post-hoc calibration can recover much of this correction without changing the learned model. Their benefits therefore need not add once ρ_{inf} is calibrated, suggesting that inference-time coupling acts as a useful calibration degree of freedom for temporal fidelity.

F.10 FULL LONG-HORIZON FORECASTING RESULTS

Tables 12, 13 and 14 report the full 496-frame free-running results. We include joint Energy Score, and normalized lag-one correlation error within and across completion windows. Correlation errors are signed, with zero ideal. In the endpoint-anchored variant, each Jumper endpoint launches the next window; in the endpoint-conditioned variant, the endpoint conditions the completion but the final Completer state launches the next window.

The anchored variant therefore provides a fixed-endpoint-skeleton control: for a fixed Jumper realization, changing the coupling modifies only the states between successive endpoints.

Table 12: **Full 496-frame results on Kolmogorov flow.** $\widetilde{\Delta C}_1^{\mathrm{int}}$ and $\widetilde{\Delta C}_1^{\mathrm{bnd}}$ measure lag-one correlation error within completion windows and across window boundaries, respectively. †PDE-Refiner has no intrinsic interior/boundary distinction; for comparability, we impose the same 16-frame partition used for Jump–Complete when computing the corresponding correlation errors. Jump–Complete checkpoints are trained with $\rho_{\mathrm{train}} = 0$; within each pair, coupling changes only ρ_{inf}, with learned weights fixed. The validation-selected coupling, obtained by minimizing validation VS_t, is $\rho^\star = 0.95$. Values are mean ± s.d. across seeds.

Method	$\mathrm{ES}^{\mathrm{joint}}_{1:496}\downarrow$	$\widetilde{\Delta C}_1^{\mathrm{int}}$	$\widetilde{\Delta C}_1^{\mathrm{bnd}}$
Jump–Complete, endpoint-anchored	12.7756 ± 0.0005	-0.3693 ± 0.0223	-0.1921 ± 0.0103
+ coupling ($\rho^\star = 0.95$)	$\mathbf{12.7609 \pm 0.0004}$	$\mathbf{+0.0497 \pm 0.0067}$	-0.1811 ± 0.0123
Jump–Complete, endpoint-conditioned	12.7661 ± 0.0412	-0.3276 ± 0.0258	-0.0374 ± 0.0026
+ coupling ($\rho^\star = 0.95$)	12.7698 ± 0.0250	$+0.0687 \pm 0.0082$	$+0.0514 \pm 0.0129$
Joint-Forecast	12.7716 ± 0.0464	$+0.1711 \pm 0.0254$	$\mathbf{-0.0224 \pm 0.0414}$
DYffusion	12.8207 ± 0.0271	-0.1261 ± 0.1073	-0.6324 ± 0.1784
PDE-Refiner†	12.8712 ± 0.1860	$+0.0597 \pm 0.1016$	$+0.0591 \pm 0.1060$

Table 13: **Full 496-frame results on Kuramoto–Sivashinsky.** Metrics and notation follow Table 12. †PDE-Refiner has no intrinsic interior/boundary distinction; for comparability, we impose the same 16-frame partition used for Jump–Complete when computing the corresponding correlation errors. Jump–Complete checkpoints are trained with $\rho_{\mathrm{train}} = 0$; within each pair, coupling changes only ρ_{inf}, with learned weights fixed. The validation-selected coupling, obtained by minimizing validation VS_t, is $\rho^\star = 0.99$. Values are mean ± s.d. across seeds.

Method	$\mathrm{ES}^{\mathrm{joint}}_{1:496}\downarrow$	$\widetilde{\Delta C}_1^{\mathrm{int}}$	$\widetilde{\Delta C}_1^{\mathrm{bnd}}$
Jump–Complete, endpoint-anchored	$\mathbf{14.1212 \pm 0.0000}$	-0.0265 ± 0.0011	-0.0161 ± 0.0006
+ coupling ($\rho^\star = 0.99$)	$\mathbf{14.1212 \pm 0.0000}$	-0.0067 ± 0.0010	-0.0129 ± 0.0002
Jump–Complete, endpoint-conditioned	14.1221 ± 0.0160	-0.0258 ± 0.0065	-0.0182 ± 0.0065
+ coupling ($\rho^\star = 0.99$)	14.1415 ± 0.0182	$\mathbf{-0.0040 \pm 0.0042}$	$\mathbf{-0.0064 \pm 0.0044}$
Joint-Forecast	14.4281 ± 0.0212	-0.0116 ± 0.0061	-0.0449 ± 0.0071
DYffusion	14.8352 ± 0.0278	-0.1464 ± 0.0186	-1.7452 ± 0.3637
PDE-Refiner†	14.7776 ± 0.2490	$+0.2485 \pm 0.4820$	$+0.2490 \pm 0.4936$

Table 14: **Full 496-frame results on HW2D.** $\widetilde{\Delta C}_1^{\mathrm{int}}$ and $\widetilde{\Delta C}_1^{\mathrm{bnd}}$ measure lag-one correlation error within completion windows and across window boundaries, respectively. †PDE-Refiner has no intrinsic interior/boundary distinction; for comparability, we impose the same 12-frame partition used for Jump–Complete when computing the corresponding correlation errors. Jump–Complete checkpoints are trained with $\rho_{\mathrm{train}} = 0$; coupling changes only ρ_{inf}, with learned weights fixed. The validation-selected coupling, obtained by minimizing validation VS_t, is $\rho^\star = 0.75$. Values are mean ± s.d. across seeds.

Method	$\mathrm{ES}^{\mathrm{joint}}_{1:496}\downarrow$	$\widetilde{\Delta C}_1^{\mathrm{int}}$	$\widetilde{\Delta C}_1^{\mathrm{bnd}}$
Jump–Complete, endpoint-conditioned	21.3312 ± 0.0286	-0.0974 ± 0.0275	$+0.1333 \pm 0.0114$
+ coupling ($\rho^\star = 0.75$)	21.3436 ± 0.0285	$\mathbf{+0.0559 \pm 0.0516}$	$+0.2379 \pm 0.0254$
Joint-Forecast	21.3392 ± 0.0126	$+0.2641 \pm 0.0138$	$+0.2722 \pm 0.0135$
DYffusion	21.4055 ± 0.0074	$+0.2539 \pm 0.0239$	-0.6394 ± 0.1017
PDE-Refiner†	$\mathbf{21.2920 \pm 0.0191}$	$+0.0672 \pm 0.0138$	$\mathbf{+0.0681 \pm 0.0141}$

F.11 LONG-HORIZON TEMPORAL POWER SPECTRA

We complement the lag-one correlation diagnostics with the temporal power spectral density (tPSD) of 496-frame free-running rollouts, defined in Appendix B. Figure 12 shows the generated-to-reference spectral ratio across temporal frequencies.

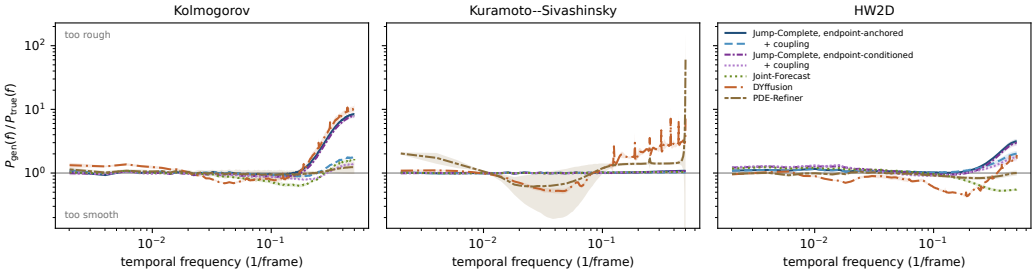


Figure 12: **Temporal spectra over 496-frame free-running forecasts.** Ratio of generated to reference tPSD for Kolmogorov flow (left), Kuramoto–Sivashinsky (center), and HW2D (right). We show endpoint-anchored and endpoint-conditioned JUMP–COMPLETE, with their calibrated coupling variants, together with JOINT-FORECAST, DYffusion, and PDE-Refiner. The horizontal line at one denotes exact spectral agreement; values above and below one indicate excess and deficient temporal variability.

On Kolmogorov flow, both uncoupled JUMP–COMPLETE variants develop strong excess power at high frequencies, reaching nearly an order of magnitude above the reference. Coupling suppresses most of this excess, bringing the high-frequency ratio close to one for both variants, with the endpoint-conditioned model closest to the reference. JOINT-FORECAST shows a deficit at intermediate-to-high frequencies followed by a mild excess at the highest ones, while DYffusion combines an intermediate-frequency deficit with a high-frequency excess comparable to uncoupled JUMP–COMPLETE. PDE-Refiner remains closest to the reference across all frequencies.

On Kuramoto–Sivashinsky, all JUMP–COMPLETE variants and JOINT-FORECAST remain close to the reference over the full frequency range, so that the effect of coupling is not visible at this scale and appears only in the aggregate tPSD error (Figure 5). In contrast, DYffusion and PDE-Refiner show a pronounced intermediate-frequency deficit and excess high-frequency power, with PDE-Refiner additionally overestimating the lowest frequencies.

On HW2D, uncoupled JUMP–COMPLETE again develops excess high-frequency power, which coupling reduces, most strongly for the endpoint-conditioned variant. JOINT-FORECAST and DYffusion instead underestimate intermediate-to-high frequencies, while PDE-Refiner stays closest to the reference, with only a mild deficit.

Together with Table 1, these results show that on Kolmogorov flow and HW2D, temporal coupling affects long-horizon dynamics across multiple time scales, beyond what is captured by lag-one correlation alone.

F.12 ENERGY-SCORE SKILL RELATIVE TO CLIMATOLOGY

To contextualize the magnitude of the long-horizon Energy Score, we compare each model with a climatological ensemble in Kolmogorov Flow and 2D Kuramoto–Sivashinsky. For each target, the climatological ensemble contains $M = 16$ trajectory segments drawn from test trajectories different from the target trajectory. We average over five such draws for each of the 50 test targets.

We define Energy-Score skill relative to climatology as

$$S_{\mathrm{ES}}(W) = 100 \left(1 - \frac{\mathrm{ES}^{\mathrm{joint}}_{1:W}}{\mathrm{ES}^{\mathrm{clim}}_{1:W}} \right),$$

so that 100% is perfect and 0% corresponds to climatological skill.

Table 15: **Climatological joint Energy Score.** Each climatological ensemble contains 16 trajectory segments sampled from test trajectories other than the target trajectory. Values are mean ± s.d. over repeated climatological draws.

Horizon	Kolmogorov	Kuramoto–Sivashinsky
1:16	2.4355 ± 0.0106	2.8023 ± 0.0092
1:32	3.4170 ± 0.0128	3.9535 ± 0.0232
1:128	6.8463 ± 0.0237	7.9662 ± 0.0170
1:496	13.6952 ± 0.0369	15.9904 ± 0.0301

Table 16: **Joint Energy-Score skill relative to climatology (%).** Skill decreases with horizon as forecasts lose realization-level synchronization with the reference trajectory. PDE-Refiner attains high short-horizon skill but degrades substantially at longer horizons. Coupling changes the Energy Score only weakly, even where it substantially changes temporal correlation and spectral fidelity.

Method	1:16	1:32	1:128	1:496
Kolmogorov				
Jump–Complete, endpoint-anchored	85.0 ± 0.1	66.1 ± 0.0	18.6 ± 0.0	6.7 ± 0.0
+ coupling	85.1 ± 0.1	66.6 ± 0.0	18.9 ± 0.0	6.8 ± 0.0
Joint-Forecast	78.1 ± 0.5	57.1 ± 1.1	17.2 ± 0.8	6.7 ± 0.3
DYffusion	69.3 ± 1.5	54.8 ± 1.2	16.9 ± 0.6	6.4 ± 0.2
PDE-Refiner	88.8 ± 0.7	71.6 ± 2.4	19.7 ± 0.7	6.0 ± 1.4
Kuramoto–Sivashinsky				
Jump–Complete, endpoint-anchored	98.7 ± 0.0	97.0 ± 0.0	46.0 ± 0.0	11.7 ± 0.0
+ coupling	98.7 ± 0.0	97.0 ± 0.0	46.0 ± 0.0	11.7 ± 0.0
Joint-Forecast	97.8 ± 0.1	93.0 ± 0.5	35.3 ± 0.6	9.8 ± 0.1
DYffusion	83.8 ± 2.6	75.3 ± 1.0	22.8 ± 0.4	7.2 ± 0.2
PDE-Refiner	99.0 ± 0.1	96.6 ± 0.5	32.4 ± 3.9	7.6 ± 1.6