CONSTRUCT-FIRST FLOW MATCHING FOR FEW-PASS TIME-SERIES GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Short-window time-series generation requires reproducing both temporal dependence and non-Gaussian structure, some of which can be estimated directly from training data. How much learned dynamics remain necessary once these components are supplied explicitly? We investigate this question through construct-first flow matching, which combines a fitted Gaussian reference, a learned residual velocity, and fixed, structure-dependent output transformations. The Gaussian component is propagated analytically, and the teacher is distilled into one or two neural maps before the output transformations are applied. For the Gaussian reference flow with an isotropic source, we show that uniform-grid Euler requires $\Theta(\sqrt{\kappa})$ steps to resolve the smallest-variance direction to fixed relative accuracy, where κ is the ratio of source variance to the smallest data variance. Analytic propagation eliminates this component's discretisation error. On six short-window benchmarks, we evaluate generation using splits without timestamp overlap, memorisation diagnostics, and re-evaluation on a separate test split. Among configurations not flagged by the diagnostics, our method achieves the lowest RFF-C2ST score on four datasets on both evaluation splits, using zero to two neural evaluations. Ablations reveal distinct contributions: learned dynamics improve ETTh1 and Stock, interact beneficially with the transformations on Energy, and are not required to attain the reported scores on Sines, MuJoCo, and fMRI. These results show where direct statistical construction can replace neural generation and where combining the two remains beneficial.

1 INTRODUCTION

Time-series generators are evaluated on short windows that carry strong temporal dependence and dataset-specific marginal structure. A flow-matching generator learns these properties jointly, which can require repeated neural evaluations at sampling time: generating each sample with an iterative sampler integrates a probability-flow ODE, and existing approaches reduce that cost through better solvers, transport paths or distillation. Some of the structure, however, can be estimated directly from the training windows before any gradient step. This raises the question we study: *how much learned dynamics remain necessary once those components are supplied explicitly?*

We study this question with *construct-first flow matching*. A Gaussian reference $\mathcal{N}(\mu, \Sigma)$ fitted to the training windows supplies the analytic part of the velocity field; a residual velocity is learned on top of it (the teacher); the teacher is distilled into one or two student maps; and fixed, structure-dependent output transforms act on the student's output: each is screened in the training split, and each dataset's configuration records which of them it uses (Sec. 5).

The theory concerns the Gaussian part. The window covariance of standard benchmarks is highly anisotropic – the ratio κ of source to smallest data eigenvalue reaches 4.6×10^7 on Energy, higher on the trajectory datasets – and for Gaussian data with those moments the reference velocity field has a closed form (Prop. 3.1). Integrating it by forward Euler on a uniform grid needs $\Theta(\sqrt{\kappa})$ steps to resolve the smallest-eigenvalue direction (Thm. 4.3), whereas the telescoping identity (Lemma 4.1) integrates it exactly at any step count; this is why the Gaussian component is propagated analytically and only the residual is left to the network. The distillation splits the teacher's trajectory at an intermediate time whose marginal has a variance floor in every direction (Prop. 5.1).

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.

The principle is that **what can be constructed from the training data with zero gradient steps need not be integrated by the network at inference time.** The construction combines two always-on statistical components (a covariance skeleton and a per-window level-and-scale match) with seven structure-dependent ones (constant runs, quantisation knots, tied variates, single-cell extremes, a generative family, a within-window third-moment law, and exact difference pairs with roughness and first-step laws), each with a screening rule at a fixed threshold and a positive and a negative control. A rule can fire on a dataset whose configuration does not use its transform, so Tables 1 and 6 report firing and use separately.

Evaluation must separate the generator's training data from both the model-selection and the test data. The standard benchmark scores generated windows against a held-out split of the *generator's own training data*; we verify this in eleven official codebases, measure that the row-level split they run inside the scorer leaves 100.0% of the classifier's test windows overlapping its training windows (a day-level split still leaks 48–53%), and adopt a block split with zero timestamp overlap. A memorisation gate, validated by a positive control, is applied to every arm including our own (it flags three baselines whose emitted windows duplicate one another 28–38% of the time, and three candidate variants of our own); and because the comparison split was used during development, every arm is re-scored on a third split that no reported configuration was selected on.

Under this protocol, and among the arms the gate does not flag, construct-first flow matching has the lowest score on four of six datasets – Stock, Energy, Sines, fMRI – at zero to two network passes against one hundred to two thousand; on ETTh1 it trails ImagenTime by $.025$ on the development split while both obtain $.630$ on the confirmatory split, and on MuJoCo it leads at matched budget while a sampler taking 250 times as many steps edges it by $.006$. The gains are heterogeneous, and a component ablation identifies their sources: learned dynamics improve the score on ETTh1 and Stock, contribute through their interaction with the output transforms on Energy, and are not required to attain the reported scores on Sines, MuJoCo and fMRI. Every remaining gap is reported with its diagnostic chain.

The contributions, in the order argued:

- **C1 (Analysis).** A step-complexity account of flow matching on time-series covariances: for the Gaussian reference problem, uniform-grid Euler needs $\Theta(\sqrt{\kappa}/\varepsilon)$ steps to resolve the stiffest direction (Thm. 4.3), while the fitted skeleton's probability-flow ODE telescopes *exactly* at any step count and on any grid, so that part costs zero integration steps and a learned residual can be interleaved inside the propagator (Lemma 4.1, Cor. 4.2; closed-form Gaussian propagators themselves are known, Sec. 2). We also evaluate geometrically spaced grids numerically – thirty-four to seventy-four steps across our six datasets – as an alternative, without a proven rate (Obs. 4.4). The analytic identities are accompanied by numerical consistency checks.
- **C2 (Method and attribution).** Construct-first flow matching: a fitted Gaussian reference, residual flow learning, teacher distillation into one or two network passes, and output transformations screened in the training split with positive and negative controls, each dataset's configuration recording which it applies (the covariance skeleton enters the field, the rest act on the output); and an ablation that attributes the reported score, per dataset, to the reference, the student pipeline or the transforms (Table 3).
- **C3 (Protocol and evidence).** A split with zero timestamp overlap (0.00% against 100.0% for the row-level split eleven codebases actually run, and 48–53% for a stricter day-level one) and a matched-units memorisation gate applied to every arm including our own – it flags three baselines and three of our own candidate variants – under which we have the lowest unflagged score on four of six datasets at zero to two passes against one hundred to two thousand network calls, tie on a fifth on the third split, lead MuJoCo at matched budget, and report each remaining gap with its diagnostic chain, a computed MDE and a re-score on a third split.

2 Related Work

Exponential/semilinear integrators for diffusion. Exponential integrators (Hochbruck & Ostermann, 2010) integrate exactly whatever linear operator a semilinear problem supplies; what fixes the operator is the model. In the diffusion samplers built on them – DEIS (Zhang & Chen, 2022), DPM-Solver (Lu et al., 2022) – the operator comes from the *noise schedule*: a data-independent scalar

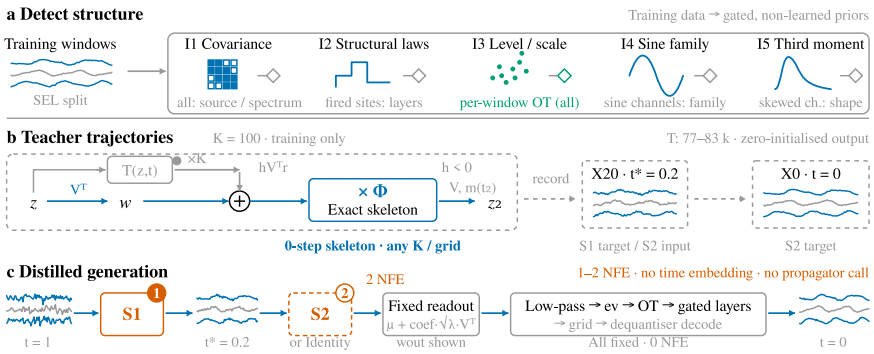


Figure 1: **Construct-first flow matching.** *(a)* The detectors (Table 1) read the training windows and fire only where their structure exists; everything they install is *estimated from data and then fixed*. *(b)* The teacher integrates the skeleton with the exact propagator $\Phi(t_2, t_1)$, the residual entering as an Euler increment *inside* Φ (Eq. 4); with $r \equiv 0$ the factors telescope. Two states are recorded, at $t^* = 0.2$ and $t = 0$. *(c)* Each student map regresses one full stage and calls no propagator; on Sines the family layer replaces the student's output (App. P).

integrating factor, with the data-dependent Gaussian part of the score left inside the extrapolated term (plain flow matching lacks this schedule structure, Tan et al., 2025). Our skeleton supplies the *data-covariance* operator $V\,\mathrm{diag}(c^u(t))\,V^\top$ as the linear part and integrates it exactly (Lemma 4.1); the small-variance directions that carry the $\Theta(\sqrt{\kappa})$ cost of Thm. 4.3 are, for a schedule-only operator, inside the extrapolated term.

Few-step time-series flow matching. The closest prior work is FM-TS/FlowTS (Hu et al., 2024), an early application of rectified flow to time series: it learns the whole field and relies on rectification for few-step sampling, with no data-dependent structure installed before training. We differ on three concrete axes – where the structure comes from (detected from the training split rather than learned), what is integrated exactly (a data-covariance operator, against no explicit propagator), and under what protocol the result is scored (a split with no timestamp overlap and a memorisation gate, on six datasets). Re-run under our split with its released recipe, FlowTS trails our arm at every step count on every dataset and is reported in App. L (at its default 100 steps it would be the closest baseline on Stock, .636, and no other comparison changes); Table 2 carries the published flow-matching baselines TimeFlow (He et al., 2025), an Euler sampler, and SpectralMeanFlow (Kim et al., 2025), whose released configurations use a midpoint solver at 500 steps (ETTh1, Stock, Sines) and 1000 (Energy, MuJoCo, fMRI), each midpoint step costing two network calls; we run both at those settings and compare at ours (Sec. 7).

Vector-quantised autoregressive generation. MSDformer (Feng et al., 2025) and TimeMar (Xu et al., 2026) autoregress over a learned discrete codebook at nine-to-eighteen and six passes per window; our diagnostics (Sec. 6) flag both, on duplicate rate and on nearest-neighbour distance to the training set.

Unconditional time-series generation. The line we evaluate against runs from adversarial and variational generators – TimeGAN (Yoon et al., 2019), TimeVAE (Desai et al., 2021), Koopman VAE (Naiman et al., 2024b) – through diffusion and flow models differing mainly in the representation they diffuse in (Yuan & Qiao, 2024; Naiman et al., 2024a; Wang & Milenkovic, 2025; Park et al., 2026; Cai et al., 2026; Li et al., 2026; He et al., 2025; Kim et al., 2025). The diffusion and flow models estimate the whole field without an explicit data-covariance propagator and sample with one hundred to two thousand network calls; the adversarial and variational generators sample in one pass but, on this protocol, at scores far from the real-vs-real reference.

Few-step sampling by distillation. Progressive distillation (Salimans & Ho, 2022) and consistency models (Song et al., 2023) also reach one or two passes by training a student to reproduce a multi-step teacher's map. Our students do the same (Sec. 5); the difference is what the teacher integrates – there, the entire field, including the stiff directions Thm. 4.3 concerns; here, a residual on top of an exactly propagated skeleton.

Memorisation in generative models. Diffusion models are known to reproduce training examples (Carlini et al., 2023; Somepalli et al., 2023); we flag it with the nearest-neighbour criterion of Gu et al. (2025), the authenticity statistic of Alaa et al. (2022), and a within-generated-set duplication count (Sec. 6), which is not novel but needs no threshold and no reference to the training set.

Exact-solution/complexity results for Gaussian dynamics. Four recent papers have narrow technical overlap with Sec. 4: Zhang (2026) fits a terminal map exact on normal modes in one diffusion step with a schedule-only linear part, where our $\Phi(t_2, t_1)$ is valid between any two times with a data-dependent operator, so the residual can be interleaved at every step; Pierret & Galerne (2025) is the closest: their Proposition 3 gives the probability-flow ODE's exact solution for *Gaussian data* under a VP-SDE as the covariance-square-root map $\Sigma_T^{-1/2}\Sigma_{T-t}^{1/2}$, so a closed-form Gaussian propagator is not new and we do not claim it; what differs is an independent-endpoint flow-matching interpolant rather than a noise schedule, a skeleton *fitted* to non-Gaussian data with a learned residual interleaved inside the propagator, and a step-complexity rate, which they do not state; Stéphanovitch (2026) bounds a Wasserstein distance for geometric grids; Wainwright (2026) analyses $\log\det(I + \Sigma/\delta)$. None states a $\sqrt{\kappa}$ step-complexity result for the uniform grid.

3 PRELIMINARIES

A window is $x \in \mathbb{R}^n$, $n = LC$, $L = 24$, C dataset-dependent (ETTh1 7, Stock 6, Energy 28, Sines 5, MuJoCo 14, fMRI 50). Data lives in Z-units, $x = 2\,\mathrm{D}(\text{raw}) - 1 \in [-1, 1]^n$, after a per-channel min–max over the series and with D the dequantiser fitted on the training split (App. E; on Stock the network operates on per-window standardised shapes instead, App. U); source and intermediate states are unbounded and the output is clipped at the decode (App. T). For $x_1 \sim p_{\text{data}}$, independent source $x_0 \sim \mathcal{N}(\mu_0, \Sigma_0)$, the interpolant is

$$z_t = (1-t)\,x_1 + t\,x_0, \qquad u := \tfrac{d}{dt} z_t = x_0 - x_1, \qquad t \in [0, 1], \tag{1}$$

$t = 0$ is the data end, $t = 1$ the source end; $u^*(z, t) = \mathbb{E}[x_0 - x_1 \mid z_t = z]$, integrated from $t = 1$ to $t = 0$ (Lipman et al., 2023; Liu et al., 2023). The CFM loss is minimised by $v = u^*$; we train $v = u_{\text{skel}} + T$, u_{skel} a **closed-form skeleton field**, T the learned residual, $T^* = u^* - u_{\text{skel}}$ vanishing exactly when the data is Gaussian with the skeleton's moments.

We call ETTh1, Stock and Energy the *standard* datasets and Sines, MuJoCo and fMRI – collections of independent trajectories – the *trajectory* datasets; they are split differently. Each is cut into three disjoint parts: a **training** split (the generator's windows) and two held-out splits, **validation** and **test**. Every score is generated-vs-validation and every null validation-vs-test. **The validation split was used for development on all six datasets** (95 configurations of our own arm were scored against it), so every arm is also re-scored on the test split, which no reported configuration was selected on (App. Q). Our comparison with the best unflagged baseline is unchanged on five of six (rankings among individual baselines do change, App. Q); our score moves by at most .011 on four datasets, by $+.032$ on Stock and by $+.070$ on ETTh1, where our "indistinguishable from the reference" reading does not survive.

The skeleton $\mathcal{N}(\mu, \Sigma)$ is estimated without gradient-based optimisation (Table 6), with eigendecomposition $\Sigma = V \Lambda V^\top$.

Proposition 3.1 (Closed-form skeleton field)**.** *With* $D_k(t) := (1-t)^2\lambda_k + t^2\lambda_k^0$,

$$c_k^u(t) = \frac{t\lambda_k^0 - (1-t)\lambda_k}{D_k(t)} = \tfrac{1}{2}\tfrac{d}{dt}\log D_k(t), \qquad u_{skel}(z,t) = (\mu_0 - \mu) + V\,\mathrm{diag}(c^u(t))\,V^\top(z - m_t). \tag{2}$$

Tag: proven; checked against ordinary least-squares estimates on simulated Gaussian data. This fixes notation once, matching the released code, for Lemma 4.1 and Thm. 4.3.

4 STEP COMPLEXITY OF FLOW MATCHING ON TIME-SERIES COVARIANCES

Lemma 4.1 (Exact decoupling and telescoping)**.** *In* $w := V^\top(z - m(t))$*, the skeleton ODE decouples per direction,* $\dot{w}_k = c_k^u(t)\,w_k$*, with exact solution* $w_k(t_2) = \Phi_k(t_2, t_1)\,w_k(t_1)$*,* $\Phi_k(t_2, t_1) =$

$\sqrt{D_k(t_2)/D_k(t_1)}$. *Because Φ_k telescopes along any partition of $[0,1]$, the sampler's discrete update reproduces this exact product for* ***every step count*** *$K \geq 1$* ***and every time grid*** *(Cor. 4.2): with the residual zero, K steps of any spacing give $z_K = \mu + V\,\mathrm{diag}(\sqrt{\lambda_k/\lambda_k^0})\,V^\top(z_0 - \mu_0)$, z_0 the source draw – an exact $\mathcal{N}(\mu, \Sigma)$ sample in one step for an isotropic source. (Tag: proven.)*

Corollary 4.2. *(Tag: proven for the local identity; the global first-order bound is stated for an* exogenous *residual $\rho(t)$ and is not derived for a state-dependent one, where the error feeds back through $z(t)$.) The residual enters the update only through a linear, variation-of-constants term: the sampler is a Lawson (integrating-factor) Euler scheme whose local error vanishes identically when $\rho \equiv 0$ and is homogeneous of degree one in ρ; for an exogenous residual the global error is $O(h(\sup|\rho'| + \frac{1}{2}\log\kappa_k \sup|\rho|))$ – a $\log\kappa_k$, not $\sqrt{\kappa_k}$, dependence, verified numerically in App. A.*

The constructed part of the field therefore costs zero integration steps at any resolution; the cost below is what a network would otherwise integrate on its own.

Theorem 4.3 (Boundary layer, uniform grid)**.** *Write $\kappa := 1/\lambda_{\min}$ in prior-variance (Z) units* – not $\mathrm{cond}(\Sigma)$*, which inflates the reported ETTh1 κ roughly $4\times$ if used in its place. For $K \geq \sqrt{\kappa}$,*

$$\log \rho_K = -\frac{1 + I_2(\lambda)}{K} + O\Big(\frac{\kappa}{K^2}\Big), \qquad I_2(\lambda) = \frac{\pi}{8}\sqrt{\kappa} - \frac{1}{2} + O(\kappa^{-1/2}), \tag{3}$$

where ρ_K is the ratio of uniform-grid Euler amplitude to the exact amplitude $\sqrt{\lambda_{\min}}$ in the stiffest direction, so $K^*(\varepsilon) = \Theta(\sqrt{\kappa}/\varepsilon)$ steps are needed for relative accuracy ε.

Tag: proven for the structure; remainder constants not tracked; the asymptotic constant is numerically verified by bisection against the exact product to within 0.1% for $\kappa \in [10, 10^{10}]$. A companion lemma shows the uniform-grid step factor lies strictly in $(0,1)$ at every node with $t_j > \lambda/(1+\lambda)$, hence at every node whenever $K < \kappa$, which holds for every step count we run on the five datasets with $\kappa > 10^3$ and, on fMRI ($\kappa \approx 72$), for $K < 72$: in that regime the integrator can only *starve* a stiff direction, never amplify it, so $K^*(\varepsilon)$ is an accuracy, not a stability, requirement (Fig. 2). A negative control (a matched, not isotropic, source) gives $K^*(.05) = 26$ regardless of κ: the $\sqrt{\kappa}$ cost is the isotropic boundary layer, not a generic conditioning artefact. That control is also the obvious alternative remedy, and it is our own construction: matching the source requires exactly the covariance skeleton we build, and it does not supply the exact propagator that lets a learned residual be interleaved with it.

κ must be estimated with care: an in-sample minimum eigenvalue at $N/d \approx 7$–8 is a biased estimate of the population quantity. We fix the eigenbasis on the training split and measure *held-out* variance in those directions, which removes the selection bias of using the same windows twice but is still a finite-sample estimate; the in-sample estimate of κ is larger by $521\times$ on Stock and $2.6\times$ on Energy and agrees to 10% on ETTh1 ($N/d = 29.3$). Table 4 reports every κ as this held-out estimate.

The samplers we compare against integrate on grids uniform in their own time variable at 100–1000 steps; under forward Euler on the reference problem, five of six datasets would require $6.8\times$ to $7{,}800\times$ more than a 100-step sampler to resolve their stiffest direction ($1.4\times$–$1{,}570\times$ against a 500-step one; Fig. 3). This is a statement about the reference problem and that discretisation, not a step count for any particular published model, whose ODE, time parametrisation and integrator differ. fMRI is the exception: at $N/d = 2.4$ its eigenbasis is poorly estimated, so the held-out estimate can miss the truly smallest-variance direction and understate the population κ (it is not a certified bound in either direction), and $K^* = 74$ already falls inside what published samplers use. High κ is what gives the *integration* half of construct-first its leverage, not the detected-structure half – fMRI has the lowest κ in the suite and the largest construction-only margin we report.

Observation 4.4 (Geometric grid)**.** *(Tag:* ***numerical, not proven****. The fixed shape of the layer in $\log t$ is an exact* sech *identity; the step count itself is obtained by numerically optimising the grid, and we state it as an achievability estimate for one schedule family, not as a bound and not as a lower bound over all schedules.) A geometric grid $t_j = r^j$, truncated at $t_{\min}$ and closed by one final step to $t = 0$, appears to resolve the same layer in $O(\log\kappa/\varepsilon)$ steps, because the layer occupies an interval of fixed length in $\log t$ while the grid is uniform in $\log t$.*

Across the $\log\kappa$ range spanned by our six datasets (4.3 to 23.1), a geometric grid whose ratio, depth and truncation $t_{\min}$ are chosen numerically to minimise the step count needs **thirty-four to seventy-four** steps at $\varepsilon = .05$ (34/39/40/60/69/74 at fMRI/Stock/ETTh1/Energy/MuJoCo/Sines, the

fMRI value extrapolated below the tabulated range; App. O). This is a numerical illustration for one schedule family, not a bound and not a claim of optimality; the samplers we compare against integrate on grids uniform in their own time variable, verified in released code (App. O), and the rate of Thm. 4.3 is for forward Euler on that grid, not for every solver.

5 CONSTRUCT-FIRST FLOW MATCHING

The generator has three types of components (Fig. 1). *Constructed*: the skeleton $\mathcal{N}(\mu, \Sigma)$, its field u_{skel}, and a set of post-sampler layers, all functions of the training windows alone, computed with zero gradient steps and then held fixed; by Lemma 4.1 the skeleton is integrated exactly at any step count. *Learned*: the residual T, the only object the teacher stage trains by gradient descent, with $v = u_{\text{skel}} + T$ and the source $\mathcal{N}(\mu_0, \Sigma_0)$ kept isotropic, $\mathcal{N}(0, I)$, in every reported configuration (the matched-source variant appears only as the negative control of Thm. 4.3). *Distilled*: two student maps trained separately on the teacher's trajectories. The construction comes from a *fixed* detector set (Table 1; the candidates its controls rejected are in App. S). Each detector (i) screens the training split alone for its structure at a fixed threshold; (ii) passes a positive control, its transform measurably removing an injected perturbation of the kind it repairs; and (iii) is characterised by a negative control, the transform being harmful or inert where its structure is absent. The controls test the transform, not that the screen would switch it off, and a rule that fires need not be used: Table 6 records which transforms each dataset applies, and Energy's tied-channel constraint is specified by hand.

The teacher minimises the flow-matching loss with the skeleton field in the target,

$$\mathcal{L}(\theta) = \mathbb{E}\big\| u_{\text{skel}}(z_t, t) + T_\theta(z_t, t) - (x_0 - x_1) \big\|^2, \quad w_2 = \Phi(t_2, t_1) \odot \big\{ w_1 + (t_2 - t_1) V^\top T(z_1, t_1) \big\}, \tag{4}$$

in the coordinate $w := V^\top (z - m_t)$ of Lemma 4.1, t decreasing from 1 to 0. The second expression is the sampler: a Lawson (integrating-factor) Euler step whose residual increment sits *inside* Φ, so with $T \equiv 0$ the propagators telescope and the skeleton endpoint is reached exactly at any K. **With $T \not\equiv 0$ it is a first-order approximation of the full flow**: exactness applies to the skeleton component, never to the sampler as a whole. Zero-initialising T makes the residual vanish at the *start* of training only. Six error sources stay distinct – skeleton estimation; skeleton discretisation; the fit of T_θ to the target residual T^*; the discretisation of the learned residual; distillation, i.e. whether the students reproduce the teacher; and the distribution change the post-sampler layers introduce – and exact integration removes only the second; the residual's discretisation error remains, first order in the step size and homogeneous of degree one in a prescribed residual (Cor. 4.2). These are conceptual sources, not terms of a proven additive decomposition.

Level and scale are not part of the covariance story. The skeleton reproduces level's *variance* but not the non-Gaussian joint law of $(m, \log s)$, and for non-Gaussian data matching the pooled mean and covariance does not in general recover the joint distribution of per-window means and scales. We match the pair by square linear assignment on standardised coordinates, or to a fitted Gaussian-copula cloud where the exact transplant trips the memorisation gate; the assignment reproduces the training multiset of $(m, \log s)$ pairs before clipping and rounding. At inference the layers run in a fixed order – a covariance-based Wiener shrinkage $\Sigma(\Sigma + \alpha\bar{\sigma} I)^{-1}$ in Z-units with a scalar regularisation parameter α chosen on an internal subset of the training split (App. U), the level-and-scale assignment, then the structural, family and shape layers (App. U).

DISTILLATION TO ONE OR TWO PASSES

We split the teacher's trajectory at a fixed intermediate time $t^* = 0.2$ and train one student per stage. The Gaussian boundary layer peaks at $t_{\text{peak}} = (\sqrt{\lambda} + \lambda)/(1 + \lambda)$, which ranges from 10^{-5} to 0.13 across our datasets, so t^* is a split point chosen for conditioning, not the boundary-layer time:

Proposition 5.1 (Intermediate-marginal variance floor)**.** $\mathrm{Cov}(x_{t^*}) \succeq t^{*2} I = 0.04\, I$ *in Z-units for an isotropic source, in every direction and regardless of dataset (proven for the skeleton flow; under a marginal-preservation assumption for the teacher), against data eigenvalues as small as* 2.2×10^{-8} *(Energy) and* 10^{-10} *(Sines) at the* $t = 0$ *end.*

Table 1: The detectors, as implemented (App. U); rules are evaluated on training windows only and "fires on" is the measured outcome. Which transforms a dataset applies is recorded in Table 6; where a rule fires on a dataset that does not apply its transform, the row says so.

detector	fires when (training windows only)	installs	fired on
covariance skeleton	always; its value scales with how far the held-out spectrum reproduces the in-sample one	field u_{skel} (source kept isotropic)	all six (ETTh1: in/out eigenvalues agree to 10%)
constant runs	$\geq 15\%$ of the channel's training increments are exactly zero (and it is not a knot channel)	hold layer, run-length law	Energy 17/28, ETTh1 1/7
quantisation knots	second differences below 10^{-5} in $\geq 60\%$ of cells, the nonzero ones on one residue class mod 6 in $\geq 95\%$ of windows	knot layer on the station grid	Energy 6/28
tied variates	two channels equal on every training cell (found by inspection, hard-coded for Energy)	copy constraint	Energy, one pair
level and scale	always (the joint law of per-window mean and scale, which the pooled moments fix only for Gaussian data)	OT match of $(m, \log s)$	all six
generative family	mean best single-sinusoid $R^2 > .999$	family-draw layer	Sines 5/5 (next-highest .9977)
single-cell extremes	$P(\max_t \lvert z_{t,c}\rvert > 3) \geq .01$, knot channels excluded	one cell set to the assigned training window's extreme	fires on five; installed on Energy 20/28, ETTh1 7/7
third moment	mean within-window skew, $\lvert t\rvert > 3$ on each half of the training split with the same sign; law if $\geq C/2$ channels	quadratic skewness-adjustment layer	fires fMRI 31/50, Sines 5/5, MuJoCo 9/14; installed on fMRI
difference pairs, roughness, first-step jump	exact affine backward-difference pairs $v_j = a\,\Delta x_i + c$ on every training cell (max error $< 10^{-4}$); the roughness and first-step laws are then applied to the source channels of the detected pairs (switched on for MuJoCo, App. U)	difference constraint, roughness and first-step laws	MuJoCo 7 pairs / 14

The first map carries the source to t^* and the second to the data end. Neither is a residual network: each regresses the teacher's *full* stage map from $32{,}768$ pairs generated by the teacher sampler (exact skeleton propagation, $K{=}100$ residual updates), so "only the residual is learned" describes the teacher stage, not the students. Neither is time-conditioned and neither calls the propagator, which reaches them through their targets and, in two of the three readout modes, through a fixed skeleton-coloured decode (App. K). On Stock the second stage is the identity, so that dataset samples in one pass. The second stage's training metric – a per-direction weighting by $1/\sqrt{\lambda_k}$ (floored at $10^{-2}\lambda_{\max}$), or whitening that colours the student's error by the skeleton's spectrum – is selected **per dataset on an internal subset of the training split, disjoint from both scoring sets** (whitened on Energy, Sines and MuJoCo, weighted on ETTh1, Stock and fMRI); we have no rule that predicts the choice in advance.

6 Evaluation Protocol

The standard discriminative-score protocol trains a classifier to separate generated windows from a "held-out" set. **We use a different classifier-based metric**: an **RFF-C2ST accuracy** (Lopez-Paz & Oquab, 2017) defined in App. F, where the conventional score is computed by a two-layer GRU classifier and reported as $\lvert\text{acc} - .5\rvert$, so we re-run every baseline under ours and never mix the two scales. It is an accuracy: lower is better, .5 is chance under the balanced exchangeable null, and a value *below* .5 shows that the null is not calibrated by the exchangeability argument (grouped folds over few groups can produce it), which is why the fMRI null of .395 is a diagnostic and not a ranking. Two things are conflated in the standard set-up. **(i)** The real windows the scorer is given are the *generator's own training windows*: eleven official codebases divide the comparison set *inside* the scoring routine, on an array the generator is already fit to. **(ii)** That division is an 80/20 cut of randomly permuted window *rows* of stride-1 windows, so 100.0% of the classifier's test windows overlap one of its own training windows by twelve or more of twenty-four timestamps. Even a day-level split – stricter than anything a repository runs – leaks 48–53%, against 0.00% on the block split we adopt. App. E names the routine in each repository and gives the instrument's two controls; the claim covers those eleven codebases, not every model (TSFlow (Kollovieh et al., 2025) has a temporal holdout).

Why one primary metric. The field reports a battery – GRU discriminative score, predictive score, context-FID, correlational score – all computed against the same comparison set, so one defect moves them all. Under *our* split we ran faithful ports of the first two (App. R): they rank arms differently, and the discriminative score puts the copying arm ahead of us, so a longer list of the

Table 2: **Generation quality under the protocol of Sec. 6.** RFF-C2ST (lower is better) at each method's own published step count, grouped folds; the fMRI column is read against its own uncalibrated null. The real-vs-real row is an empirical reference, not a bound. **Bold**: the best unflagged baseline per column, and our arm where it beats that baseline. Parentheses: flagged arms (App. F) – self-duplicating with a high memorised fraction (†), or a high memorised fraction alone (¶); § no published recipe. Baselines are single runs except where a ± (standard deviation over seeds) is shown; ours: three seeds, five on ETTh1; the fMRI arm uses no network and its ± is classifier noise (App. P). SpectralMeanFlow's 500/1000 midpoint steps are 1000/2000 network calls.

arm	ETTh1	Stock	Energy	Sines	MuJoCo	fMRI
null (real vs. real)	.566	.513	.633	.493	.504	.395
Ours (0–2 passes)	.560±.015	**.586**	**.727**	**.503**	.895	**.535**±.001
in the comparison						
TimeFlow (100 passes)	.619±.006	.909	1.000	.998	1.000	**.759**
TimeBridge	.653	.803	.915	.931	.980	.773
KoVAE	.679	.690	.947	.959	.921	.970
SpectralMeanFlow (500/1000)	.687	.750	1.000	.978	.999	.965
Diffusion-TS (500)	.747±.010	.841	**.871**±.008	.836	**.889**	.891
WaveletDiff	.753	.725	.983	.965	.996	.949
PaCoDi	.808	.741	.949	.958	.977§	.871
TimeVAE	.978	.858	.977	**.764**	.965	.912
TimeGAN	.996	.999	1.000	1.000	1.000	1.000
ImagenTime	**.535**	**.685**	.998	.959	1.000	(.583)§¶
flagged by the memorisation gate (duplicate rate 28–38%*, memorised fraction* .96–1.00*)*						
MSDformer†	(.552)	(.572)	(.828)	(.866)	(.966)	(.465)
SDFlow†	(.548)	(.635)	(.853)	(.722)§	(.962)§	(.476)§
TimeMar†	(.552)	(.584)	(.862)	(.738)§	(.959)§	(.479)§

same kind would not have caught what the memorisation gate catches. We fix one *two-sample* statistic as the primary endpoint – the kind whose null we can measure in the same run – and report complementary diagnostics of independent kinds (App. F).

A second failure mode survives a test set with no timestamp overlap: near-copying the training set. We use a matched-units nearest-neighbour **copy-rate** and Gu et al.'s memorised fraction (Gu et al., 2025), both validated by a dose-response control, plus a statistic needing no calibration: the fraction of a generator's *own* windows that duplicate one another exactly, 0.0% for real data on every dataset. An arm is flagged when its duplicate rate exceeds 1% or its memorised fraction exceeds $.10$ (App. F). On ETTh1 it is bimodal with nothing between: MSDformer, TimeMar and SDFlow sit at 35–37%, every other arm at exactly 0.0% (Fig. 6). Near-copying pays where the copy's perturbation falls below what the discriminator resolves (App. I). On ETTh1 four baselines and our own arm sit *below the real-vs-real reference*, which is an empirical estimate on two particular splits and not a bound: three of the four baselines are flagged, ImagenTime and our arm are not (App. F).

The same gate excluded three candidate variants of our own (App. F).

7 Experiments

Setup. Six benchmarks at $L = 24$, three disjoint parts each (App. E); thirteen baselines from the authors' released code with their published recipe where one exists; the RFF-C2ST of Sec. 6 against the real-vs-real null of the same run; the same memorisation gate on every arm. Ours: three training seeds, five on ETTh1; baselines: single runs unless a ± is shown; ± is the standard deviation over seeds, classifier randomness for the zero-network fMRI arm (Table 2).

Headline (Table 2). Among the arms the gate does not flag we have the lowest score on Stock, Energy, Sines and fMRI, improving over the best unflagged entry in Table 2 by .099/.144/.261/.224; on ETTh1 ImagenTime leads us by .025 on the validation split and ties us on the test split (App. Q); on MuJoCo we trail Diffusion-TS at 500 steps by .006 and lead it by .101 at two passes each. Three gated arms score lower than us on ETTh1, Stock and fMRI (Sec. 6, App. I). Figure 5 adds authenticity (Alaa et al., 2022): of the ten arm/dataset cells at or below ours, eight sit at $\leq .02$.

Attribution (Table 3). Rows 1–3 change *only* the network, all post-sampler layers off; rows 4 and 5 turn the same layers on above the student and above the bare skeleton. Row 3 against row 1: the student pipeline (teacher training plus distillation) is worth $-.18$ on ETTh1 and Stock and increases the score on the other four ($+.001$ to $+.050$). Row 4 against row 5, the 2×2 contrast with the layers on: it is worth $-.200/-.184$ on ETTh1/Stock, nothing on Sines and MuJoCo, $+.108$ on fMRI,

Table 3: **Where the score comes from.** RFF-C2ST, lower is better; five seeds on ETTh1, three elsewhere; same scorer and split as Table 2, re-scored from the stored samples (hence third-decimal differences). Rows 1–3 disable the output transforms but keep the decode needed to obtain samples (on Stock it includes the level-and-scale reconstruction, held fixed), so they differ only in the network; rows 4–5 add the same layers above the student and above the bare skeleton, with the Wiener coefficient pinned to row 4's per seed. Row 5's fMRI entry (.519) uses the student configuration's coefficient; the reported fMRI arm is row 5's configuration with the coefficient its own rule selects (.535). [§]On Sines the family draw sets every cell, so rows 4 and 5 share one output distribution (App. P).

	ETTh1	Stock	Energy	Sines[§]	MuJoCo	fMRI
skeleton only, **0** network passes	.806	.773	.884	.961	.992	.762
+ teacher, 100 steps	.611±.011	.570±.012	.932±.004	.934±.024	.992±.000	.767±.003
+ student, 1–2 passes	.627±.009	.585±.014	.934±.012	.969±.003	.993±.000	.801±.003
+ the constructed layers (reported, except fMRI)	**.560**±.014	**.585**±.017	**.727**±.022	**.504**±.002	**.895**±.002	.627±.007
skeleton + the same layers, **0** passes	.760±.002	.769±.001	.787±.010	.504±.001	.895±.000	**.519**±.000

and $-.060$ on Energy – where on its own it made the raw sample worse (.934 against .884) but combined with the same transforms it beats the skeleton pipeline (.727 against .787), so its benefit there depends on the combination. So the reported number needs the student pipeline on ETTh1 and Stock, benefits from it only jointly with the layers on Energy, and is reached by the construction alone on Sines, MuJoCo and fMRI – where the reported arm has no network (.535 against .627 for the two-pass arm). Grafted onto Diffusion-TS the construction reaches .695 on Energy at two passes (Table 7). **Sines is a positive control, not a hard win:** its family detector fires on all five channels and a firing channel is *replaced* by a sinusoid drawn from the training split's own (ω, φ) cloud, so the reported sample carries no flow output and the .261 margin says nothing about modelling capacity (App. P).

Matched budget. Running each baseline's own few-step sampler at our step count compares quality under a restricted budget; it does not isolate architecture, since solver, objective and post-processing differ too. At $K = 2$, and still at $K = 50$, neither Diffusion-TS nor TimeFlow comes near us on ETTh1, Stock or Energy (Table 7); on MuJoCo Diffusion-TS scores .9963 at $K = 2$ against our .8952; on fMRI its best step count (.709 near $K = 10$) trails both of our reported fMRI variants. Network time per 1,000 windows (RTX 4090) is 0.002 s for our at most two passes against 0.6–4.1 s for the baselines, but end to end our cost is dominated by the level-and-scale assignment, so the end-to-end margin is $1.6\times$ to $11\times$ rather than three orders of magnitude (App. V). Figure 7 shows the low-variance eigenbins that published generators starve or noise-fill and that our sampler resolves by construction.

8 Limitations

The benefit of the learned dynamics is dataset-dependent: without the output transforms the student improves the primary score on two datasets, with them also on Energy (an interaction, Table 3), and on the trajectory datasets it is not required to attain the reported scores; we claim construct-first as a principle and only this for the residual. **Scope.** Everything is at $L = 24$ with a dense $n \times n$ covariance and a per-batch $O(N^3)$ assignment; we claim nothing at longer horizons (App. K). Three choices are made on an internal subset of the training split, the skeleton family and each dataset's set of transforms are chosen without a rule (Table 6), and the detector thresholds are screens, not tests (App. T). The scorer was used throughout development; the third-split re-score checks transfer, not adaptation to it (App. Q). **Open.** Unresolved are MuJoCo's remaining .006 and fMRI's margin to the gated arms, diagnosed as subject-level dispersion of the moment statistics that none of the transforms we tested supplied while passing its controls (App. P).

9 Conclusion

Directly fitted statistical components can reduce the neural computation short-window generation needs. The analytic propagator removes the Gaussian component's discretisation error, and the ablation separates the contributions: learned dynamics help on ETTh1 and Stock, interact with the output transforms on Energy, and are not required for the reported results on the three trajectory datasets. These results show where statistical construction can replace neural generation and where combining the two remains beneficial.

Reproducibility statement

Every number in this paper comes from one pipeline invoked per dataset with the configuration of Table 6; the residual network, optimiser and training budget are given in App. K, the splits and their leakage instrumentation in App. E, and the scorer, its nulls and the memorisation diagnostics in App. F. Baselines are the authors' released code run with their own published recipe where one exists for that dataset; cells where a repository ships no recipe are marked and reported for completeness only. Every baseline export carries a record of its units that the scorer refuses to run without. The code, the detector implementations, the scorer and the per-seed logs will be released; App. V says how each number is reproduced and reports the three components of the inference cost.

AI use statement

Generative AI tools were used to polish the prose of this manuscript. The method, the theorems and their proofs, the experimental design, the code and every number reported here were produced and verified by the authors, who take responsibility for the final content.

References

Ahmed M. Alaa, Boris van Breugel, Evgeny Saveliev, and Mihaela van der Schaar. How faithful is your synthetic data? sample-level metrics for evaluating and auditing generative models. In *International Conference on Machine Learning (ICML)*, 2022.

Rongyao Cai, Yuxi Wan, Kexin Zhang, Ming Jin, Zhiqiang Ge, Qingsong Wen, and Yong Liu. Parallel complex diffusion for scalable time series generation. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '26)*, 2026.

Luis M. Candanedo, Véronique Feldheim, and Dominique Deramaix. Data driven prediction models of energy use of appliances in a low-energy house. *Energy and Buildings*, 140:81–97, 2017.

Nicholas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, Florian Tramèr, Borja Balle, and Daphne Ippolito. Extracting training data from diffusion models. In *USENIX Security Symposium*, 2023.

Abhyuday Desai, Cyrus Freeman, Zeqi Wang, and Ilya Beaver. Timevae: A variational auto-encoder for multivariate time series generation. *arXiv preprint arXiv:2111.08095*, 2021.

Shibo Feng, Zhicheng Chen, Xi Xiao, Zhong Zhang, Qing Li, Xingyu Gao, and Peilin Zhao. MSDformer: Multi-scale discrete transformer for time series generation. *arXiv preprint arXiv:2505.14202*, 2025.

Xiangming Gu, Chao Du, Tianyu Pang, Chongxuan Li, Min Lin, and Ye Wang. On memorization in diffusion models. *Transactions on Machine Learning Research*, 2025.

Panjing He, Mingyue Cheng, Li Li, and XiaoHan Zhang. TimeFlow: Towards stochastic-aware and efficient time series generation via flow matching modeling. *arXiv preprint arXiv:2511.07968*, 2025.

Marlis Hochbruck and Alexander Ostermann. Exponential integrators. *Acta Numerica*, 19:209–286, 2010.

Yang Hu, Xiao Wang, Zezhen Ding, Lirong Wu, Huatian Zhang, Stan Z. Li, Sheng Wang, Jiheng Zhang, Ziyun Li, and Tianlong Chen. FlowTS: Time series generation via rectified flow. *arXiv preprint arXiv:2411.07506*, 2024.

Jinwoo Kim, Max Beier, Petar Bevanda, Nayun Kim, and Seunghoon Hong. Sequence modeling with spectral mean flows. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. arXiv:2510.15366.

Marcel Kollovieh, Marten Lienen, David Lüdke, Leo Schwinn, and Stephan Günnemann. Flow matching with gaussian process priors for probabilistic time series forecasting. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2410.03024.

Wei Li, Shibo Feng, Pengcheng Wu, Xingyu Gao, Min Wu, and Peilin Zhao. SDFlow: Similarity-driven flow matching for time series generation. *arXiv preprint arXiv:2605.05736*, 2026.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations (ICLR)*, 2023. arXiv:2210.02747.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations (ICLR)*, 2023. arXiv:2209.03003.

David Lopez-Paz and Maxime Oquab. Revisiting classifier two-sample tests. In *International Conference on Learning Representations (ICLR)*, 2017.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *arXiv preprint arXiv:2206.00927*, 2022.

Ilan Naiman, Nimrod Berman, Itai Pemper, Idan Arbiv, Gal Fadlon, and Omri Azencot. Utilizing image transforms and diffusion models for generative modeling of short and long time series. *Advances in Neural Information Processing Systems*, 37:121699–121730, 2024a.

Ilan Naiman, N. Benjamin Erichson, Pu Ren, Michael W. Mahoney, and Omri Azencot. Generative modeling of regular and irregular time series data via Koopman VAEs. In *International Conference on Learning Representations (ICLR)*, 2024b.

Jinseong Park, Seungyun Lee, Woojin Jeong, Yujin Choi, and Jaewook Lee. TimeBridge: Better diffusion prior design with bridge models for time series generation. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2026. arXiv:2408.06672.

Emile Pierret and Bruno Galerne. Diffusion models for Gaussian distributions: Exact solutions and Wasserstein errors. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, 2025.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations (ICLR)*, 2022.

Stephen M. Smith, Karla L. Miller, Gholamreza Salimi-Khorshidi, Matthew Webster, Christian F. Beckmann, Thomas E. Nichols, Joseph D. Ramsey, and Mark W. Woolrich. Network modelling methods for FMRI. *NeuroImage*, 54(2):875–891, 2011.

Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Diffusion art or digital forgery? investigating data replication in diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *International Conference on Machine Learning (ICML)*, 2023.

Arthur Stéphanovitch. Lipschitz regularity in flow matching and diffusion models: Sharp sampling rates and functional inequalities. *arXiv preprint arXiv:2604.06065*, 2026.

Zheng Tan, Weizhen Wang, Andrea L. Bertozzi, and Ernest K. Ryu. STORK: Faster diffusion and flow matching sampling by resolving both stiffness and structure-dependence. *arXiv preprint arXiv:2505.24210*, 2025.

Martin J. Wainwright. Denoising growth complexity: Data geometry and certified schedules for diffusion sampling. *arXiv preprint arXiv:2607.26285*, 2026.

Yu-Hsiang Wang and Olgica Milenkovic. WaveletDiff: Multilevel wavelet diffusion for time series generation. *arXiv preprint arXiv:2510.11839*, 2025.

Xiangyu Xu, Qingsong Zhong, and Jilin Hu. TimeMar: Multi-scale autoregressive modeling for unconditional time series generation. *arXiv preprint arXiv:2601.11184*, 2026.

Jinsung Yoon, Daniel Jarrett, and Mihaela van der Schaar. Time-series generative adversarial networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Xinyu Yuan and Yan Qiao. Diffusion-ts: Interpretable diffusion for general time series generation. In *International Conference on Learning Representations (ICLR)*, 2024. arXiv:2403.01742.

Qinsheng Zhang and Yongxin Chen. Fast sampling of diffusion models with exponential integrator. *arXiv preprint arXiv:2204.13902*, 2022.

Shiheng Zhang. Asymptotic preservation and uniform accuracy of diffusion and flow-matching samplers. *arXiv preprint arXiv:2607.04113*, 2026.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 11106–11115, 2021.

A PROOFS AND NUMERICAL VERIFICATION

Throughout, $D_k(t) = (1-t)^2\lambda_k + t^2\lambda_k^0$, $w = V^\top(z - m_t)$ with $m_t = (1-t)\mu + t\mu_0$, and $\Phi_k(t_2, t_1) = \sqrt{D_k(t_2)/D_k(t_1)}$. Every statement is tagged *proven*, *assumed*, or *empirical*, and each headline claim is labelled by the tag of the step it rests on.

A.1 PROPOSITION 3.1 (CLOSED-FORM SKELETON FIELD)

Proven. For jointly Gaussian (x_0, x_1) with $x_1 \sim \mathcal{N}(\mu, \Sigma)$, $x_0 \sim \mathcal{N}(\mu_0, \Sigma_0)$ independent and Σ, Σ_0 co-diagonalised by V, the interpolant $z_t = (1-t)x_1 + tx_0$ is Gaussian with mean m_t and covariance $V\mathrm{diag}(D(t))V^\top$. Writing $\bar{w} = V^\top(z - m_t)$, the conditional expectation $\mathbb{E}[x_0 - x_1 \mid z_t = z]$ is linear in $\bar{w}$ with per-direction coefficient $\mathrm{Cov}(u_k, \bar{w}_k)/D_k(t) = (t\lambda_k^0 - (1-t)\lambda_k)/D_k(t)$, since $\mathrm{Cov}(x_{0,k} - x_{1,k},\, (1-t)x_{1,k} + tx_{0,k}) = t\lambda_k^0 - (1-t)\lambda_k$. The identity $c_k^u = \frac{1}{2}(\log D_k)'$ follows from $D_k'(t) = 2(t\lambda_k^0 - (1-t)\lambda_k)$. Known-answer gate: coefficients recovered by OLS on simulated Gaussian data agree with the closed form to the precision of the simulation (a check on the algebra, not a proof).

A.2 LEMMA 4.1 AND COROLLARY 4.2 (EXACT TELESCOPING)

Proven. $\dot{z} = \dot{m}_t + V\dot{w}$ with $\dot{m}_t = \mu_0 - \mu$, and $u_{\text{skel}} = (\mu_0 - \mu) + V\mathrm{diag}(c^u(t))w$, so the skeleton ODE decouples into $\dot{w}_k = c_k^u(t)w_k$. Since $c_k^u = \frac{1}{2}(\log D_k)'$ and $D_k > 0$ on $[0,1]$ whenever $\lambda_k, \lambda_k^0 > 0$, the function $t \mapsto w_k(t)/\sqrt{D_k(t)}$ has zero derivative by the product rule, so $w_k(t_2) = \Phi_k(t_2, t_1)w_k(t_1)$ for any $t_1, t_2 \in [0,1]$. (This argument does not divide by w_k, so it covers $w_k \le 0$; the linear ODE's uniqueness also gives $w_k \equiv 0$ as the only solution through $w_k(t_1) = 0$.) The sampler's update is $w_{k+1} = \Phi(t_{k+1}, t_k) \odot (w_k + (t_{k+1} - t_k)V^\top r_k)$ with r_k the network residual; with $r \equiv 0$ the product telescopes for *every* $K \ge 1$ and every grid: $w_K = \prod_k \Phi(t_{k+1}, t_k) \odot w_0 = \sqrt{D(0)/D(1)} \odot w_0 = \sqrt{\lambda/\lambda^0} \odot w_0$, i.e. $z_K = \mu + V\mathrm{diag}(\sqrt{\lambda/\lambda^0})V^\top(z_0 - \mu_0)$, which for an isotropic source ($\lambda^0 \equiv 1$) is $\mu + \Sigma^{1/2}(z_0 - \mu_0)$ with $\Sigma^{1/2} := V\mathrm{diag}(\sqrt{\lambda})V^\top$ the symmetric square root: an exact $\mathcal{N}(\mu, \Sigma)$ sample in one step. *Verification:* $|\prod_k \sqrt{D(t_{k+1})/D(t_k)} - \sqrt{\lambda}| \le 1.7 \times 10^{-18}$ for $K \in \{1, 2, 3, 7, 100, 1000\}$; end-to-end in the float32 pipeline, 5.6×10^{-11}.

A.3 VARIATION OF CONSTANTS: HOW THE RESIDUAL ENTERS

Proven for the local-error identity; the global bound is stated for an exogenous residual and checked numerically, its derivation with tracked constants is not given, and it is not extended to a state-dependent residual. For the full field $u = u_{\text{skel}} + T$ with $\rho(t) = V^\top T(z(t), t)$ along the exact trajectory, $w(t_2) = \Phi(t_2, t_1) \odot w(t_1) + \int_{t_1}^{t_2} \Phi(t_2, s) \odot \rho(s)\, ds$. The sampler is the Lawson (integrating-factor) Euler scheme for this equation, so its local error $e_k = \int_{t_1}^{t_2} [\Phi_k(t_2, s)\rho_k(s) - \Phi_k(t_2, t_1)\rho_k(t_1)]ds$ is *identically zero* when $\rho \equiv 0$ and is homogeneous of degree one in ρ: the skeleton is integrated exactly at any K and any grid, and the residual's discretisation error is first order in the step size, homogeneous of degree one in a prescribed ρ, with a $\log\kappa$, not $\sqrt{\kappa}$, dependence. *Verification (exogenous residual $\rho = r_0(1 + \sin 3t)$, ETTh1 $\lambda_{\min}$):* error $3.7\times10^{-3}/3.7\times10^{-2}/3.7\times10^{-1}$ for $r_0 = 10^{-3}/10^{-2}/10^{-1}$ at $K = 4$ (linear in r_0 to the printed precision, for this prescribed exogenous residual) and $5.5\times10^{-4}/5.5\times10^{-5}/5.5\times10^{-6}$ at $K = 10^2/10^3/10^4$ (exactly $\propto 1/K$). Plain Euler's error at $K = 100$ is 0.37 and does *not* vanish as $r_0 \to 0$: that part is skeleton mis-integration, the subject of Thm. 4.3.

A.4 THEOREM 4.3 (THE DATA-END BOUNDARY LAYER)

Fix a direction, write $\lambda \in (0, 1)$ for its data eigenvalue in isotropic-source units and $\kappa = 1/\lambda$.

Exact peak (proven). c_λ attains its maximum on $[0,1]$ at $t^* = (\sqrt{\lambda} + \lambda)/(1 + \lambda)$ with $c_\lambda(t^*) = (1 + \lambda)/(2\sqrt{\lambda})$: extrema solve $(\log D)'' = 0$, i.e. $-2(1+\lambda)^2t^2 + 4\lambda(1+\lambda)t + 2\lambda(1-\lambda) = 0$, with roots $(\lambda \pm \sqrt{\lambda})/(1+\lambda)$. On ETTh1's smallest direction ($\lambda = 8.038 \times 10^{-5}$) this is a peak of height 55.77 at $t = 0.0090$, matching a grid search to 10^{-9}.

Unit-width bump in log-time (proven). $t/(\lambda+t^2) = \frac{1}{2\sqrt{\lambda}}\mathrm{sech}(\log t - \log\sqrt{\lambda})$ exactly, and $\|c_\lambda - t/(\lambda+t^2)\|_{L^2}/\|c_\lambda\|_{L^2} = O(\sqrt{\lambda})$ (measured $0.71\sqrt{\lambda}$). The layer is therefore a fixed-shape bump of width $\sqrt{\lambda}$ in t, i.e. unit width in $\log t$ – which is exactly why a geometric grid helps and a uniform one does not.

No step is unstable (proven). With $K \geq 2$, $h = 1/K$ and nodes $t_j = jh$, the Euler factor $f_j = 1 - hc_\lambda(t_j)$ satisfies $0 < f_j < 1$ at every node with $t_j > \lambda/(1+\lambda)$ and $1 < f_j \leq 1+h$ otherwise. The requirement in Thm. 4.3 is therefore an *accuracy* requirement – resolving a bump of height $\Theta(\sqrt{\kappa})$ and width $\Theta(1/\sqrt{\kappa})$ – not an A-stability requirement. Measured: $\max_j hc_\lambda(t_j) = 1.000000$ over all $K \leq 3000$ and $K \in \{10^4, 10^5, 10^6\}$, attained only at $K = 1$.

Starvation law and the rate (proven for the structure). Let $\rho_K = \prod_j f_j/\sqrt{\lambda}$ be the Euler-to-exact amplitude ratio, so ρ_K^2 is the variance ratio. For $K \ll \sqrt{\kappa}$, $\rho_K = (K-1)\sqrt{\lambda}\,(1+O(K^2\lambda)+O(1/K))$. For $K \geq \sqrt{\kappa}$,

$$\log\rho_K = -\frac{1+I_2(\lambda)}{K} + R_K, \quad I_2(\lambda) = \tfrac{1}{2}\int_0^1 c_\lambda^2 = \tfrac{\pi}{8}\sqrt{\kappa} - \tfrac{1}{2} + O(\kappa^{-1/2}), \quad |R_K| \leq C\kappa/K^2,$$

obtained by expanding $\log\rho_K = [\int_0^1 c_\lambda - h\sum_j c_j] - \frac{h^2}{2}\sum_j c_j^2 - \sum_{m\geq 3}\frac{h^m}{m}\sum_j c_j^m$ (using $\log\sqrt{\lambda} = -\int_0^1 c_\lambda$ from the telescoping identity) and applying Euler–Maclaurin with one correction term to c_λ and c_λ^2. Solving $|\rho_K - 1| < \varepsilon$ gives $K^*(\varepsilon) = \frac{\pi}{8}\sqrt{\kappa}/\varepsilon\,(1+O(\varepsilon)+O(\kappa^{-1/2})) = \Theta(\sqrt{\kappa}/\varepsilon)$. Constants in the remainder are not tracked; the asymptotic constant is verified numerically.

Verification. Predicted versus exact, by bisection on the exact product: at $\kappa = 10$, $I_2 = 0.866$ and $K^*(.05) = 37$ (prediction 36.4); at $\kappa = 10^3$, $I_2 = 11.93$ and $K^*(.05) = 253$ (prediction 252); $K\log\rho_K$ at $K = 4\times10^6$ equals $-(1+I_2)$ to three decimals at both. Small-K law on ETTh1's smallest direction: $\rho_K = .0269/.0626/.1329/.2649$ at $K = 4/8/16/32$ against the predicted $(K-1)\sqrt{\lambda} = .0269/.0628/.1345/.278$; $\rho_{100} = .624$, $\rho_{1024} = .958$, $\rho_{4096} = .989$. At generation level, with the same trained field and only the integrator changed, the variance ratio over the smallest 10% of directions is $.995/.995/.997/1.000$ for the exact scheme against $.217/.101/.066/.096$ for Euler at $K = 4/8/16/32$ (empirical).

A.5 Observation 4.4 (geometric grid)

Numerical observation, not a proven rate. A geometric grid with ratio r placing J nodes inside the layer appears, numerically, to resolve a unit-width bump in $\log t$ with a node count growing roughly logarithmically in κ over the tested range, because the layer occupies an interval of fixed length in $\log t$ (Lemma above) while the grid is uniform in $\log t$. The grid is $t_j = r^j$ for $j = 0,\ldots,J-1$ down to $t_{\min} = 10^{-3}\sqrt{\lambda}$, closed by one final step to $t = 0$; the reported count is $J+1$, and the search is over (r, J) by bisection on J at the r that grid implies. Optimising numerically gives $34/43/50/62/74$ steps at $\log\kappa = 6.9/10.7/13.8/18.4/23.0$; interpolated to our six datasets' own κ (extrapolated below $\log\kappa = 6.9$ for fMRI) that is $34/39/40/60/69/74$; we do not establish an asymptotic rate. This is not a lower bound over all schedules (Sec. 4).

A.6 Proposition 5.1 (conditioning of the first distillation stage)

Proven for the skeleton flow; assumed for the teacher. $\mathrm{Cov}(z_{t^*}) = V\mathrm{diag}(D(t^*))V^\top$ with $D_k(t^*) \geq t^{*2}\lambda_k^0 = t^{*2} = 0.04$ in Z-units for an isotropic source, *in every direction*. The point is the contrast with the data end: the same directions carry variance $\lambda_{\min} = 1/\kappa$ at $t = 0$, which is 2.2×10^{-8} on Energy and 10^{-10} on Sines (Table 4), so the ratio of the S1 target's variance floor to the dataset's smallest data eigenvalue is 0.04κ – from 2.9 on fMRI to 4×10^8 on Sines. The stiff directions that make the $t\to0$ end hard are therefore not stiff at t^*. This is a lower bound on the target's variance, not a bound on its condition number, which would additionally need $\lambda_{\max}$; and for the teacher it requires that the marginal at t^* is preserved, which we assume rather than prove.

Table 4: Held-out κ and the uniform-grid step requirement, per dataset ($\varepsilon = .05$).

dataset	N/d	κ (held-out)	$K^*(.05)$, uniform Euler
ETTh1	29.3	1.13×10^4	823
Stock	7.0	7.70×10^3	682
Energy	8.4	4.59×10^7	51,859
Sines	27.8	1.05×10^{10}	784,008
MuJoCo	9.9	1.34×10^9	279,815
fMRI	2.4	71.5 (est.; see text)	74

B THE BOUNDARY LAYER

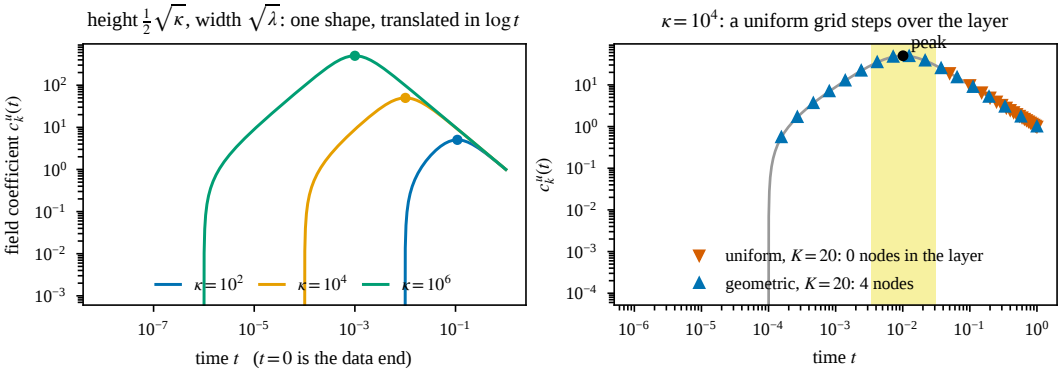


Figure 2: *Left:* the skeleton's field coefficient c_k^u for three values of κ – one shape, translated in $\log t$, of height $\frac{1}{2}\sqrt{\kappa}$ and width $\sqrt{\lambda}$. *Right:* at $\kappa = 10^4$ a uniform grid of twenty steps places no node inside the layer (shaded) while a geometric grid places four.

C STEP COUNTS AGAINST κ

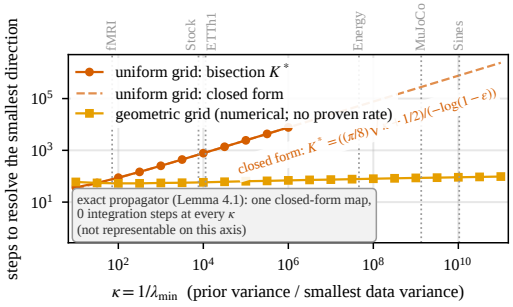


Figure 3: Steps required by uniform-grid Euler to resolve the stiffest covariance direction, $K^*(\varepsilon{=}.05) = \Theta(\sqrt{\kappa})$, evaluated at each dataset's held-out κ (Table 4). The exact propagator (Lemma 4.1) replaces the whole curve with a single closed-form map at every κ shown: zero integration steps. The zero-to-two passes we report elsewhere are calls to the distilled student maps, not integration steps.

D TIME EMBEDDINGS AT THE BOUNDARY LAYER

An empirical analysis of time-embedding resolution. The layer's coefficient function, in $\log t$, is a fixed-shape bump, so a *fixed*-frequency sinusoidal feature family supplies no feature at the layer's own scale, while a log-warped family does. At double-precision rank, a capped family ($\Omega = 2\pi$) leaves a linear-readout gap $\geq .22$ on ETTh1, $\geq .96$ on Energy; the network's own convention ($\Omega = 1000$) leaves .10–.17 / .92–.94; a sixteen-mode log-warped family reaches $\leq 10^{-3}$ everywhere. This bounds a **linear readout** of first-layer features; we make **no claim** about a nonlinear MLP on those features.

E Datasets, Splits, and the Leakage Measurement

Six datasets at $L = 24$: ETTh1 ($C = 7$; the hourly transformer-temperature series of Zhou et al. (2021)), Stock ($C = 6$; the Google stock file of the TimeGAN release (Yoon et al., 2019)), Energy ($C = 28$; the UCI appliances-energy data of Candanedo et al. (2017), as released with TimeGAN), Sines ($C = 5$; TimeGAN's synthetic sinusoids), MuJoCo ($C = 14$) and fMRI ($C = 50$; the NetSim simulations of Smith et al. (2011)), the last two as released with Diffusion-TS (Yuan & Qiao, 2024). Each is the benchmark file used by the baselines we compare against; the fMRI data are simulated, so no subject-level information is involved. Windows are stride-1, mapped to $[0, 1]$ per channel by the min and max of the whole series before splitting (a range-only use of the held-out parts; except for these global scaling parameters, every fitted quantity uses the training split alone), then to Z-units.

Each dataset is cut into three disjoint parts – the training split (the generator's windows) and two held-out splits, validation and test – so that every score in this paper is generated-vs-held-out and every null is held-out-vs-held-out. For ETTh1/Stock/Energy we use a **seven-day-block** split: the series is cut into contiguous blocks of seven days, whole blocks are assigned at random to the three parts, and windows crossing a block boundary are dropped (about 97% of the stride-1 windows are kept), so no window in one part shares a timestamp with a window in another. For Sines/MuJoCo/fMRI, which are collections of independent trajectories, we split by **trajectory**.

Leakage instrumentation. One instrument, two controls: a set of windows against itself shifted by one step must read 100% overlap, and a set against an artificially displaced one must read 0.0%; both do, on every configuration below. A window starting at index i covers timestamps $[i, i+L-1]$, so windows i and j share at least θ timestamps exactly when $|i - j| \leq L - \theta$.

Table 5: Overlap between the two sides of each split, measured with one instrument. "≥ 1" and "≥ 12" are the fraction of held-out (or classifier-test) windows sharing at least that many of their twenty-four timestamps with a window on the other side.

split	what it divides	ETTh1	Stock	Energy	controls
row-level permutation (the eleven codebases)	the classifier's train/test	100.0/100.0	100.0/100.0	100.0/100.0	100.0 / 0.0
day-level (our v1; no repository runs it)	generator vs. held-out	48.4/30.6	52.9/33.2	52.4/32.3	100.0 / 0.0
seven-day blocks (this paper)	generator vs. held-out	**0.00/0.00**	**0.00/0.00**	**0.00/0.00**	100.0 / 0.0

What the eleven repositories do, and what we checked. In each of Diffusion-TS, FM-TS/FlowTS, TimeBridge, SpectralMeanFlow, WaveletDiff, SDFlow, MSDformer, TimeMar, PaCoDi, KoVAE and ImagenTime, the unconditional discriminative score is computed by a routine that receives the *full* window array the generator was fit to and divides it by a random permutation of its rows followed by an $80/20$ index cut – a split over window rows, with no grouping by day, block or trajectory. Two consequences are distinct and we keep them distinct: the *generator* was trained on all of these windows, so no part of the comparison set is held out from it; and the *classifier* is trained and tested on stride-1 neighbours of one another. Our seven-day-block protocol changes the first, and grouped cross-validation inside the scorer changes the second.

The routine is shared across these repositories, most of which descend from one codebase; the location at which we read it in each is recorded with the released audit. TSFlow (Kollovieh et al., 2025) is the counterexample: it has no such routine and uses a temporal train/test holdout.

F Evaluation Protocol

What the remaining budget buys. Instead of further scores we report, for every cell, a real-vs-real reference and an MDE; three memorisation statistics with a stated decision rule; the authenticity of Alaa et al. (2022); and a per-feature-block decomposition that says *which* statistic separates an arm.

Our scorer (RFF-C2ST accuracy). C2ST with a cross-fitted classifier on 512 random Fourier features of the flattened window, trained by logistic regression, averaged over 5 folds and 3 classifier seeds. Folds are assigned by *group* (day, block, or trajectory), never by row, so stride-1 near-duplicate windows cannot straddle train and test. The feature map additionally carries five per-window shape statistics (skew, kurtosis, range/sd, maximum standardised increment, zero-crossing rate), normalised with statistics *shared* between the two sides – normalising each side by its own scale divides out exactly the quantity under test, a bug we hit and fixed. **When the five statistics**

were added, and why that matters. They were added before any of the six datasets in this paper was scored under the block/trajectory protocol, after the raw random-Fourier map failed its own positive control (it could not separate a prior whose per-window kurtosis dispersion was .206 against a real-vs-real value of .692; appending the five statistics moved that comparison to .771), and the set of five has not changed since. Three of them nonetheless map onto components of our method, and on fMRI we used a decomposition of this feature map on validation to choose a layer – which is why the fMRI result is confirmed on the test split, unused at the time of that check (App. N), and why the ordering of every arm is also reported under the raw-value block alone (Table 8).

Why we do not quote published numbers. Published discriminative scores are not commensurable with ours for three reasons, and we re-ran every baseline rather than cite them. (i) They are computed against a held-out set drawn from the generator's own training series (Sec. 6), which is the failure this protocol exists to remove. (ii) The convention differs: the published quantity is usually $|\mathrm{acc} - .5|$, not accuracy, so Diffusion-TS's published Stock score of .086 corresponds to an accuracy of .586 (taking accuracy above chance, as the convention does) – numerically identical to our own Stock number, measured under a leaking split against an uncalibrated reference. (iii) A quoted number carries no samples, so the memorisation gate cannot be applied to it; the three arms this paper flags were flagged on samples we generated. Choosing per arm between quoting and re-running would also select the measurement on its outcome. What published numbers are good for is a fidelity check on our *pipeline*: we ported the official GRU discriminative score faithfully (a 2-layer GRU, $|\mathrm{acc} - .5|$, the repositories' own $80/20$ row divide), ran the official Diffusion-TS code, and scored its samples with that port under *its* protocol, reproducing its published Stock score of .086 as .0848 ($.090/.088/.077$ over three seeds). This is a consistency check on sample units and on the scoring implementation. It does *not* establish that the GRU score and our RFF-C2ST measure the same thing, and we do not use it to license any comparison between the two scales.

Our own arms under the gate. On ETTh1 our copy-rate/memorised-fraction is .0068/.000 (4096/4096 unique windows; authenticity .92). ImagenTime's sub-reference ETTh1 score is not flagged by any diagnostic (duplicate rate 0.0%, memorised fraction .000, authenticity .44); our wrapper of its repository disables the score-gated checkpointing its training loop performs and exports the last iterate, so no selection on the scoring windows took place, and we count it as the best unflagged arm on ETTh1. The three of our own arms the gate flags are an exact $(m, \log s)$ transplant on MuJoCo (.904, just behind our reported .895; memorised fraction .152, copy-rate .548), a layer-only MuJoCo arm (.951; .121, .517), and the projection variant of the family layer on Sines (.600; .262, .405).

Null and MDE. Every table reports the real-vs-real null (validation vs test) computed in the same run with the same folds. On ETTh1 the null is .566 and the endpoint's 80%-power minimum detectable effect over five seeds is .0199; our arm's gap on the validation split ($-.005$) is inside it, which is a statement about the test's power and not about distributional equality, and it does not transfer to the test split ($+.065$, App. Q).

Memorisation gate. Three statistics, all in matched units. (i) *Copy-rate*: the fraction of generated windows nearer to a training window than the first percentile of the real-vs-real nearest-neighbour distance. (ii) *Memorised fraction* (Gu): the fraction of generated windows whose nearest training window is at least three times nearer than the second-nearest. (iii) *Duplicate-window rate*: the fraction of a generator's own emitted windows that exactly duplicate another of its own – threshold-free, unit-free, and requiring no reference to the training set at all. Real data reads 0.0% on every dataset. *Decision rule.* An arm is flagged when its duplicate rate exceeds 1% or its memorised fraction exceeds .10; copy-rate is reported alongside as a magnitude, always against real data's own value on the same dataset, and is never used alone. The three vector-quantised baselines trip *both* clauses (duplicate rate 28–38%, memorised fraction .96–1.00); ImagenTime on fMRI and our three excluded variants trip the memorised-fraction clause alone (memorised fraction .12–.26, duplicate rate 0.0%); a finite-support distribution can duplicate itself legitimately, which is why the duplicate rate is read against real data's own value on the same grid (0.0% on all six). Under this rule our fMRI construction arm is admitted at $3.1\times$ real's copy-rate with duplicate rate 0.0% and memorised fraction .000, and we flag that it is inside the rule but not comfortably. Calibration: Diffusion-TS retrained on $5\%/20\%/100\%$ of its own data (at $5\times$ steps) reaches copy-rate $.995/.236/.208$. The control establishes gross responsiveness; it does not calibrate the statistic at the .005–.03 scale our own arms occupy, and we make no claim that it resolves differences there.

Discriminator sensitivity. On Sines and MuJoCo, injecting Gaussian noise of standard deviation 10^{-3} and 2×10^{-3} of the value range into *real* held-out windows already moves the C2ST to .984 and .997. Any single-factor ablation on an arm scoring above .95 on these datasets therefore has limited resolution near saturation, which is why we report mechanism repairs there rather than ablation deltas.

fMRI folds. fMRI has only 17 trajectory groups, so grouped 5-fold CV rests on 17 independent clusters and its null sits below chance (.395; one real set split in half scores .476). Grouped folds are the primary protocol on fMRI as everywhere (Table 2); the ungrouped scheme (null $\approx$ .48) is given for every fMRI arm we discuss in App. N. The shift belongs to the real-vs-real comparison and leaves every gap-above-null ordering unchanged (Sec. 6).

G fMRI BY FEATURE BLOCK

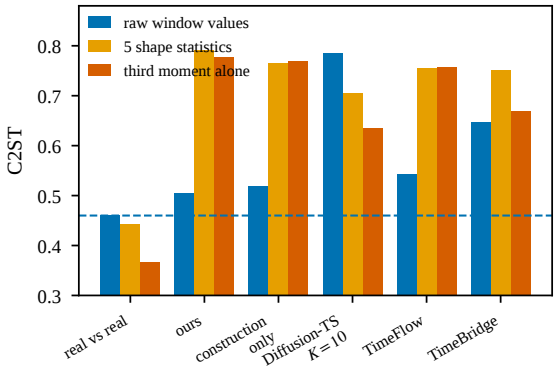


Figure 4: fMRI, grouped folds: C2ST computed on the raw window values, on the five appended shape statistics, and on the third moment alone. On the raw values our two arms are the closest of any arm to the real-vs-real row (.452 and .454 against .353; next closest .458), not equal to it; the separation is in one appended scalar.

H AUTHENTICITY AGAINST DISTANCE FROM THE REFERENCE

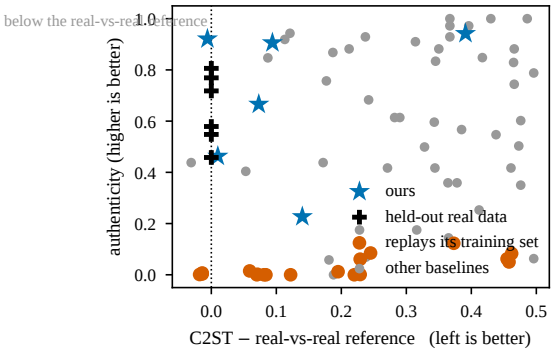


Figure 5: Every arm on every dataset, by distance from that dataset's real-vs-real reference (left is better) and authenticity (Alaa et al., 2022) (up is better). Eight of the ten cells that reach our C2ST sit at authenticity $\leq$ .02; the two that do not (ImagenTime on ETTh1, Diffusion-TS on MuJoCo) are the two cells where we do not claim a win.

I WHERE COPYING PAYS

Where copying pays, and where it does not. Near-copying pays where the copy's perturbation falls below what the discriminator resolves. On quantised ETTh1 (541 values/channel) and Stock (1,154) a perturbation at the grid's resolution is invisible and three copying arms score *better* than

us; on Sines, an exact two-parameter family, the same strategy scores only .8655, behind the non-copying TimeVAE and Diffusion-TS and far behind our .5025; on near-white fMRI the three VQ arms, 548–1151× closer to training than real windows are to each other, take the three best C2ST scores and outrank every arm not flagged by these diagnostics.

J Memorisation and Discriminator-Sensitivity Figures

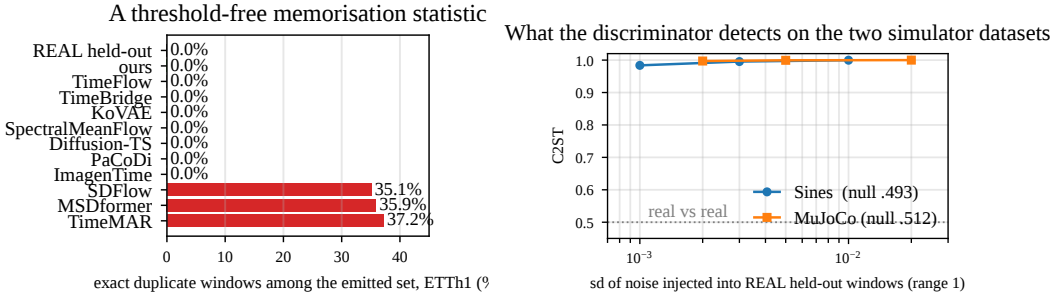


Figure 6: *Left:* duplicate-window rate on ETTh1 – real data and every non-VQ arm at exactly 0.0%, MSDformer/TimeMar/SDFlow at 35–37%, no arm in between. *Right:* discriminator sensitivity on Sines/MuJoCo – a generic 10^{-3}–10^{-2}-of-range perturbation injected into real held-out windows already separates them at .98–1.00, i.e. both test reproduction of a deterministic simulator to three decimal places. The nulls in the legend are the figure's own classifier draw and differ from Table 2's (by .008 on MuJoCo): different classifier draws give slightly different null estimates.

K Implementation and Per-Dataset Configuration

The residual network. T is a small dilated convolutional network over the window. The same architecture and the same width and depth serve the teacher and both student maps, and are unchanged across datasets, with one deliberate difference: the teacher is time-conditioned while both students are *not*, because each student map has a fixed pair of endpoints and therefore no time argument to condition on; the students' FiLM path is disabled accordingly. A 1×1 convolution lifts C channels to width $d = 64$; four residual blocks follow, each a depthwise 3-tap convolution with dilation $1, 3, 9, 1$, a pointwise convolution to $2d$, a gated activation $a \odot \sigma(b)$, GroupNorm, and FiLM modulation from a sinusoidal time embedding passed through a two-layer MLP; a 1×1 convolution maps back to C. In the teacher this last convolution is **zero-initialised**, so $T \equiv 0$ at initialisation and the sampler starts as the exact skeleton flow; the students, which regress onto teacher targets rather than onto a field, use a small random initialisation ($\sigma = 0.02$) with zero bias. **Parameter and storage budget.** The teacher holds 77.1k (Sines) to 82.9k (fMRI) parameters; each student map, having no time-embedding path, holds 36.0k to 41.8k, so the *deployed* model (S1 + S2) is 72.1k to 83.7k. The teacher is discarded after distillation. These are two to three orders of magnitude below the diffusion baselines, but the network is not the whole of what must be stored: the construction carries the eigenbasis V and spectrum λ of an $n \times n$ covariance with $n = LC$, which is 168^2 floats on ETTh1 (0.11 MB), 672^2 on Energy (1.8 MB) and 1200^2 on fMRI (5.8 MB) – on the two widest datasets the fixed construction is larger than the network it replaces, and we report it rather than quoting the parameter count alone. The detected laws add the level-and-scale cloud ($2C \times n_g$ numbers), per-channel quantile arrays and run-length multisets, and a few coefficients per channel – small beside the eigenbasis but not zero.

Scaling. Storage is $O(n^2)$ and the one-time eigendecomposition $O(n^3)$ in $n = LC$; the skeleton readout is $O(n^2)$ per sample. The level-and-scale assignment is a square linear assignment over the generated batch, $O(N^3)$ in the number of windows drawn at once. Everything we report is at $L = 24$ and $n \leq 1200$, and we claim nothing outside that range: at long horizons the dense covariance would have to be replaced by a structured one and the assignment by an approximate or minibatch solver.

The Wiener step. The construction-side Wiener step is a low-pass shape prior, not a denoiser: it improves the *noise-free* skeleton arm by -0.10, so "denoiser" would be false as used.

Training. Teacher: Adam, learning rate 6×10^{-3}, $\beta = (0.9, 0.96)$, batch 128, 18k steps (36k on ETTh1, selected by the held-out criterion), $t \sim \mathcal{U}[0.01, 0.99]$. Student: two stages, 8k steps each, same optimiser and batch. Teacher pairs for distillation are generated with the teacher sampler at $K = 100$ (32,768 pairs). No EMA, no learning-rate schedule, no gradient clipping. One seed trains all arms of a given dataset; reported cells average three seeds (five on ETTh1).

Table 6 lists the configuration behind every reported number; Table 1 gives the detector rules. Detector firing is decided on training windows only, and the two selected rows (student metric, teacher steps) by a criterion computed on an internal subset of the training split. The fMRI shape layer was additionally selected using the validation split (App. N), which was used for development on every dataset (Sec. 3); the test split carries the confirmatory result.

Table 6: Configuration behind every reported number. "–" means the layer is not installed; detection and installation are separate (Table 1), and the table records installation. *Sines: the family draw replaces the network output. extr. = single-cell extremes.

	ETTh1	Stock	Energy	Sines	MuJoCo	fMRI
skeleton	Toeplitz	centred	sample	sample	sample	Toeplitz
covariance skeleton	yes	yes	yes	yes	yes	yes
structural laws	1 hold, 7 extr.	–	17 hold, 6 knot, 1 tie, 20 extr.	–	–	–
level/scale OT	exact	copula	exact	exact	copula, cond.	exact
family draw	–	–	–	5/5 ch.	–	–
shape law	–	–	–	–	–	31/50 ch.
MuJoCo laws	–	–	–	–	7 pairs, rough., 1st step	–
student metric	weighted	weighted	ZCA	ZCA	ZCA	weighted
teacher loss	full	full	full	active (20/120)	full	full
teacher steps	36k	18k	18k	18k	18k	18k
reported NFE	2	1	2	2*	2	0

The student-metric row is the per-dataset outcome of the held-out selection rule of Sec. 5.

L Matched-NFE Comparison, Full Table

Table 7 runs each baseline's own few-step sampler at several step counts under the identical protocol. Two results stand out: on Energy, grafting our construction onto Diffusion-TS moves it from .922 to .695 at $K = 2$ – the output transformations also improve a different backbone on Energy – and on fMRI, Diffusion-TS is non-monotonic in K, best near ten steps. FlowTS, the closest prior work (Sec. 2), was re-run here because under an earlier day-level split its long-schedule sampler had been competitive with ours; under the block split it does not reach our arm at any step count on any dataset, on the validation or the test split, and on Stock, Sines and fMRI its score is non-monotone in K (best near 10–100 steps, worse at 400–1600). It passes the memorisation gate on every dataset (duplicate rate 0.0%, memorised fraction .000); on fMRI its raw-value copy-rate is 1.0, as Diffusion-TS's is there.

M Where Published Generators Lose the Spectrum

N Locating the Separating Statistic on fMRI

The scorer's feature map is 1200 raw window values plus five appended shape statistics. Scoring each block separately (Table 8) shows that our samples sat closest to the real-vs-real row on the raw values and that the third moment was by far the largest separating feature; the increment and zero-crossing statistics contribute .267 and .169 above their own nulls.

Two candidate explanations were tested and rejected. The box clamp: replacing every boundary-touching value by the linear interpolation of its in-window neighbours – the same windows, no selection – moves the skew score from .768 to .779, and our boundary fraction (.0014) is close to real data's (.0009). The decode grid: it injects the same repeat rate into every arm including the baselines (applying it to real held-out windows takes their repeat rate from .0025 to .0141), and Diffusion-TS carries a positive skew after passing through it. What remains is a genuine third moment in the data that a Gaussian skeleton cannot produce, which the shape layer supplies.

Development and confirmatory splits. The shape layer was chosen by inspecting this decomposition on the validation split, so that split was used for development here and we report the confirmatory numbers on the **test** split, which no design decision had touched at the time of this check. We

Table 7: C2ST at matched sampler passes. "+ours" grafts our construction onto the baseline's samples. FlowTS (Hu et al., 2024) is trained with its released recipe on our training split and sampled with its own shifted-time sampler; its "full" entry is the code's default of 100 steps; at 1600 steps it scores .604/.836/1.000/.996/1.000/.780 on ETTh1/Stock/Energy/Sines/MuJoCo/fMRI. The fMRI row uses ungrouped folds (null .485), every other row grouped (nulls: ETTh1 .566, Stock .513, Energy .633, Sines .494, MuJoCo .504; the Sines null differs from Table 2's .493 in the third decimal because this sweep re-drew the null with its own classifier seeds). **The "full" column is a single run of this sweep, not the seed mean of Table 2**: on ETTh1 it reads .757 against the three-seed mean $.747 \pm .010$ for Diffusion-TS and .622 against $.619 \pm .006$ for TimeFlow, and on Energy .863 against $.871 \pm .008$. Every comparison drawn from this table is within-row and within-sweep; nothing here should be differenced against Table 2.

dataset	sampler	$K{=}2$	$K{=}5$	$K{=}10$	$K{=}50$	full	ours
ETTh1	Diffusion-TS (DDIM)	.974 (+ours .959)	.912 (.883)	.851 (.812)	.724	.757	**.560** (2)
	TimeFlow (Euler)	.985 (+ours .934)	.905 (.789)	.803 (.732)	.640	.622	
	FlowTS	.919	.947	.855	.681	.612	
Stock	Diffusion-TS	.846 (+ours .842)	.823	.829	.778	.841	**.586** (1)
	TimeFlow	.997 (+ours .688)	.996 (.679)	.992 (.688)	.947	.909	
	FlowTS	.925	.875	.810	.648	.636	
Energy	Diffusion-TS	.922 (+ours .695)	.886 (.685)	.873 (.705)	.864	.863	**.727** (2)
	TimeFlow	1.000	1.000	1.000	1.000	1.000	
	FlowTS	.999	.996	.992	.983	.994	
Sines	Diffusion-TS	.966	.893	.897	.843	.836	**.5025** (2)
	TimeFlow	1.000	1.000	1.000	.999	.998	
	FlowTS	1.000	.992	.975	.944	.941	
MuJoCo	Diffusion-TS	.996	.981	.963	.912	**.889**	.895 (2)
	TimeFlow	1.000	1.000	1.000	1.000	1.000	
	FlowTS	.999	.999	.997	.990	.994	
fMRI	Diffusion-TS	.878	.721	.709	.778	.897	**.579** (0), .654 (2)
	TimeFlow	.998	.794	.780	.764	.765	
	FlowTS	.886	.806	.744	.770	.776	

Table 8: fMRI, grouped folds: C2ST by feature block, and the per-statistic one-feature score. Our two raw-value rows are the closest to the real-vs-real row of any arm measured.

arm	raw	raw (z)	5 stats	both	skew	kurt	range	max Δz	zero-cross
null (real vs. real)	.353	.460	.442	.394	.366	.386	.398	.343	.395
ours (before I5)	.454	.504	.791	.781	.776	.517	.484	.610	.564
construction only	.452	.518	.765	.762	.768	.513	.482	.460	.433
Diffusion-TS $K{=}10$	.571	.785	.704	.696	.634	.453	.509	.640	.576
Diffusion-TS full	.649	.719	.895	.891	.525	.469	.450	.645	.886
TimeFlow full	.458	.542	.755	.757	.757	.422	.499	.542	.408
TimeBridge	.579	.646	.751	.773	.668	.417	.483	.617	.725

record one later use of the test split, after this check: a candidate roughness layer for the remaining fMRI residue was fitted and applied to test windows as its front gate, failed it (App. S), and was not adopted. The confirmation below precedes that use and the reported configuration is unchanged, but test is no longer a split that no decision has ever touched. Ungrouped, the null moves from .4837 to .4835; our construction arm (the reported one; grouped $.535 \to .538$) from $.5789 \pm .0013$ to $.5817 \pm .0001$; our two-pass arm from $.654 \pm .006$ to $.666 \pm .007$; Diffusion-TS at $K{=}10$ improves from .709 to .691. The largest change is .018 (Diffusion-TS); ours move .003 and .012. The ordering is unchanged, and the test-split numbers are the ones we treat as final for fMRI.

O Step Counts by Integration Scheme

Table 9 tabulates the values plotted in Fig. 3: the uniform-grid Euler count from the closed form and by bisection on the exact scalar recursion, the geometric-grid count with ratio and depth chosen to minimise it, and the exact propagator, which is a single closed-form map at every κ. The samplers we compare against integrate on grids equally spaced in their own time variable, verified in their released samplers (Diffusion-TS and TimeFlow build the grid with a linear spacing over the diffusion or flow time; SpectralMeanFlow applies a midpoint rule on such a grid).

P Per-Dataset Diagnoses

On the three trajectory datasets the remaining gaps sit in the **constructed object**, not the sampler.

Sines. Four defects in the decode applied *after* the layers, each located by measurement and repaired behind a gated flag, take Sines from .9645 to .5025 against a null of .493; the largest rounded generated values onto the training value list, manufacturing repeats the real process never produces.

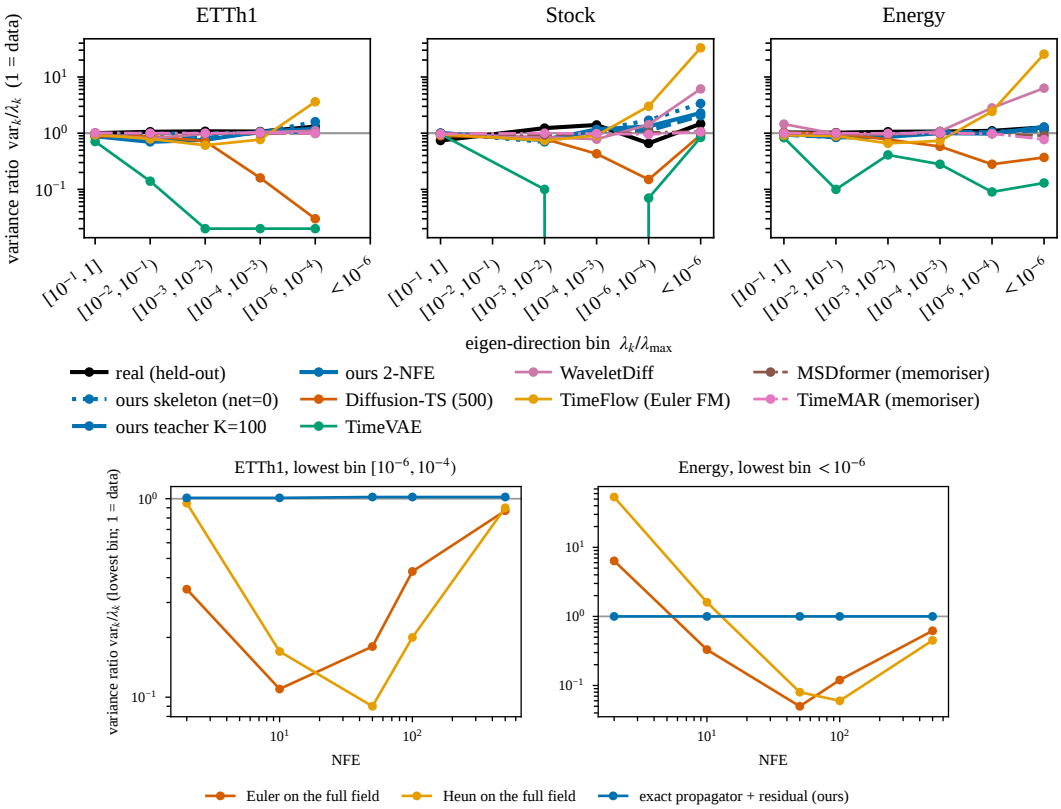


Figure 7: *Top:* per-eigenbin variance ratio (generated/data) in the training split's eigenbasis; published generators starve (ratio $\ll 1$) or noise-fill (ratio $\gg 1$) the lowest bins, while our construction sits near 1 throughout. *Bottom:* the same trained field integrated by uniform Euler vs. our exact/exponential update – Euler's ratio collapses in the stiff bins the exact update reproduces regardless of step count. The κ values of Table 4 are held-out estimates; the eigenbasis here is in-sample.

Table 9: Steps to resolve a stiff direction, by integration scheme, at representative κ ($\varepsilon = .05$). The last column counts *integration* steps; the exact propagator needs none, being one closed-form map at any step count and on any grid (Lemma 4.1). The zero-to-two passes reported in the body are calls to the distilled student maps.

$\log \kappa$	κ (approx.)	uniform Euler	geometric grid	exact propagator
6.9	$\sim 10^3$	253	34	0 (one map)
13.8	$\sim 10^6$	7,674	50	0 (one map)
23.0	$\sim 10^{10}$	766,422	74	0 (one map)

What actually produces the Sines number. The single-sinusoid detector fires on all five channels ($1 - R^2 = 10^{-5}$ against 2.3×10^{-3} on the best channel of any other dataset), and a firing channel is not corrected but *replaced*: the window becomes $(\sin(\omega t + \varphi) + 1)/2$ with (ω, φ) drawn from a Gaussian copula whose marginals are the training split's own empirical quantiles. On Sines that is every channel of every window, so the reported sample carries no output of the sampler, the teacher or the student – which is exactly what Table 3 shows (.961 / .934 / .969 for skeleton, teacher and student; .504 once the layers run). The Sines result therefore measures whether the family detector found the right family and whether the parameter cloud is well fitted. It is a positive control for the detector machinery on a two-parameter deterministic family, and we do not present it as evidence about modelling capacity.

Memorisation diagnostics for the draw and projection variants. Drawing the latents gives copy-rate .0066 against real held-out data's own .0102 and a Gu memorised fraction at real data's level. The alternative – *projecting* each generated window onto its nearest family member – scored .6004 under the protocol of the time (worse than the draw form) and failed the memorisation gate outright at copy-rate .4050 and Gu .2616, because projection pulls a wrong window into the region where the

training parameters are dense. The projection form is not used anywhere in this paper. What the draw form generalises over is the joint (ω, φ) law; its marginals are the training split's, by construction.

MuJoCo. Two structural laws and conditional matching take us from .9588 to .8952, leading Diffusion-TS at matched budget and trailing only its 500-step run. Matching the laws' *marginals* left velocity dispersion off by up to .22, because the per-window *pairing* of scale and roughness differed; transporting conditionally on the window's scale quantile repairs five of seven affected channels into the pre-specified $[0.9, 1.1]$ band at no cost on the score, and we adopt it as a default on one dataset's evidence. Conditional transport halves the dispersion error the marginal version leaves ($.13 \to .055$) without moving the score – where the discriminator saturates at 2×10^{-3} of the value range, a correct statistic need not be a visible one – so this correction did not measurably reduce the remaining gap.

fMRI. Decomposing the discriminator's feature map (App. N) isolates the loss in one appended scalar, the within-window third moment (.776 against a .366 null), with the box clamp and the decode grid excluded by paired controls. The shape layer removes it ($.791 \to .654 \pm .006$ ungrouped for the two-pass arm, against Diffusion-TS's best .709), and Table 3 shows the learned field is harmful here: the same construction applied to the bare skeleton scores .535 grouped (.538 on test, move $+.002$), against .627 for the two-pass arm, so the reported fMRI arm uses no network. That arm is deterministic up to the layer seed; its copy-rate is $3.1\times$ real data's own (.025 against .008), its duplicate rate 0.0% and its memorised fraction .000, inside the gate; its authenticity (.23) is below real data's (.56) and above the gated arms' (.00). The layer was developed on the validation split, so the test split carries the confirmation.

Why the memorisation-gated arms still score lower on fMRI, and why no further law closes it. On the 1200 raw window values our construction arm and the three gated arms are tied (.451 against .446–.452); their remaining margin (.467–.476 against our .557 on the full map, in this block analysis's own scorer run; Table 2 reads .535) lives entirely in the five appended moment statistics, where per statistic we trail on skewness (.541 vs. .49) and kurtosis (.502 vs. .41–.43) and tie on the other three. The marginals of those scalars already match real data in mean (within .2 sd) and spread (ratio .97–.98); what ours lack is the *between-subject* component – our scalars are 8–9% narrower, which equals $\sqrt{1 - \text{between-subject variance share}}$ to two decimals on both statistics. The cause is in the law: the shape coefficient is a per-subject quantity (across-subject spread $5\times$ split-half noise) with signs that vary by subject, so the pooled fit is attenuated by cancellation. Drawing the coefficient per window from the sixteen subject atoms restores the spread exactly as predicted (skewness-scalar sd .060 $\to$ .065, real .067) and buys .006 of C2ST – and applied to *real* held-out windows it moves them .32 from the null, failing the front gate by fifty times its gain. Per-subject connectivity is likewise real in the data (lagged cross-channel coupling cancels in the pooled skeleton to 27% of its per-subject norm) and invisible to the score (the raw block is a tie). What the gated arms have is the *joint* of window content and within-window moments with subject-level dispersion, associated with strong proximity to the training windows, as expected from emitting real windows; none of the transforms we tested supplied it while passing the front gate, and we stop here rather than name another.

Q Confirmatory Re-Score on the Test Split

Every reported arm is re-scored against the **test** split instead of the validation split, with n, folds, seeds, grouping, units and the feature map unchanged, so the only thing that differs is which held-out split is the reference. Two controls make the table readable. *(i) The positive control:* the vs-validation column is recomputed in the same process and diffed against a frozen snapshot of Table 2; a cell that does not reproduce its Table 2 entry is reported as void rather than interpreted. This matters because a units error shifts both columns by nearly the same amount, leaving the *move* small while both absolutes are wrong. *(ii) The run-to-run variation:* the C2ST accuracy is symmetric in its two arguments, so re-running the real-vs-real comparison with the roles swapped is the same comparison; the difference between the two runs is pure estimator noise and is reported per dataset.

What survives. Energy's move ($+.011$) and Stock's ($+.032$) are of the size of that dataset's single re-run difference (.013 and .032), and we do not interpret them. Table 11 gives every arm. On Sines, MuJoCo and fMRI the move is small ($-.003$, $+.007$, $+.002$) and the arm ordering is identical. On ETTh1 the move is $+.070$ against a run-to-run variation of .0012 and is real: measured against the

Table 10: Our arm (mean over its seeds: five on ETTh1, three elsewhere) scored against the validation split and against the test split, with the real-vs-real reference and the estimator's own run-to-run variation. The run-to-run variation is a single re-run difference, reported descriptively; it is not an uncertainty bound.

	ETTh1	Stock	Energy	Sines	MuJoCo	fMRI
real-vs-real reference	.5653	.5137	.6333	.4934	.5046	.3937
run-to-run variation (role-swapped)	$\pm.0012$	$\pm.0318$	$\pm.0129$	$\pm.0003$	$\pm.0005$	$\pm.0047$
ours vs. validation (reported)	.5604	.5844	.7274	.5035	.8953	.5354
ours vs. test (confirmatory)	.6302	.6168	.7382	.5008	.9027	.5376
move	$+.070$	$+.032$	$+.011$	$-.003$	$+.007$	$+.002$

Table 11: **Every arm re-scored against the test split** (the third split), same scorer, folds, seeds and units as Table 2; the vs-validation column of each arm recomputed in the same process reproduced its Table 2 cell (positive control; the WaveletDiff Sines/fMRI cells reproduce the corrected values). **Bold** marks the lowest unflagged test-split score per column; parentheses mark the arms flagged in Table 2. Nulls (validation vs test): $.567/.514/.633/.493/.505/.394$. Our row is the mean over its seeds.

arm (vs. test)	ETTh1	Stock	Energy	Sines	MuJoCo	fMRI
Ours (0–2 passes)	**.630**	**.617**	**.738**	**.501**	.903	**.538**
TimeFlow	.662	.910	1.000	.998	1.000	.738
TimeBridge	.732	.769	.894	.931	.982	.768
KoVAE	.745	.651	.947	.961	.926	.973
SpectralMeanFlow	.717	.769	1.000	.977	.999	.961
Diffusion-TS	.790	.814	.846	.839	**.895**	.881
WaveletDiff	.760	.740	.984	.967	.997	.945
PaCoDi	.791	.720	.945	.959	.977	.872
TimeVAE	.970	.835	.977	.767	.965	.919
TimeGAN	.996	1.000	1.000	1.000	1.000	1.000
ImagenTime	.630	.719	.999	.959	1.000	(.554)
MSDformer†	(.641)	(.607)	(.837)	(.865)	(.967)	(.505)
SDFlow†	(.639)	(.627)	(.846)	(.728)	(.963)	(.499)
TimeMar†	(.640)	(.620)	(.858)	(.741)	(.959)	(.494)

validation split our arm sits .005 *below* the real-vs-real reference (inside the .0199 MDE), and against the test split it sits $+.065$ above it, so **the ETTh1 "indistinguishable from the reference" reading is a property of the validation split and we do not claim it on the test split**. Our comparison with the best unflagged baseline changes in one place: on ETTh1, ImagenTime – the best arm on the validation split at .535 – and our arm both obtain .630 on the test split at the reported precision (Table 11); TimeFlow reads .662 there. On the other five datasets that comparison is unchanged, including MuJoCo, where Diffusion-TS's 500-step sampler keeps its .008 lead; rankings among individual baselines do change (Table 11).

Sensitivity to the evaluation split. The shift is structured. On the three trajectory datasets, whose held-out sets are independent trajectories, *every* unflagged arm moves by less than .03 and most by less than .01 (the gated arms on fMRI move by up to $+.040$). On ETTh1, where the training split, validation and test are contiguous blocks of one series, the arms that move most are those that reproduce the training windows most closely – the three memorisation-gated arms ($+.089$ to $+.091$) and ImagenTime ($+.094$) – while the arms furthest from the data barely move at all (TimeGAN $+.001$, TimeVAE $-.009$). Our own $+.070$ sits between them. The pattern is consistent with sensitivity to block-level distribution shift that grows with how tightly an arm fits the training split; it is also why the ETTh1 reference claim does not transfer.

What this cannot establish. No reported configuration was chosen on the test split (its one later use, a rejected candidate's front gate, is recorded in App. N), so this checks transfer from the validation split's particular sample to another split under the same scorer. It does *not* rule out having overfitted the evaluation protocol itself: the 95 configurations were selected by watching this scorer, and the test split is scored with the same scorer, the same feature map and the same decode grid fitted on the training split. A choice that exploits the instrument rather than the split would reproduce here unchanged.

R THE FIELD'S OWN METRICS UNDER THIS PROTOCOL

To check the claim that a battery of scores adds little, we ran faithful ports of the two metrics the time-series generation literature reports (ported from the official TimeGAN implementation) on *our*

samples and *our* zero-overlap split: the GRU **discriminative score** DS $= |\mathrm{acc} - .5|$ (lower better, 0 is chance) and the **predictive score** PS, a train-on-synthetic/test-on-real one-step MAE (lower better). The real side is the validation split; the reference row is real-vs-real, trained on the test split and tested on the validation split. DS is averaged over five classifier seeds and PS over three; our arm is the reported configuration, averaged over its three training seeds.

Table 12: The field's own two metrics, computed here on the zero-overlap split. Neither is the metric this paper reports. **Bold** marks the best arm in a column; a dagger marks the memorisation-gated arm.

	ETTh1		Stock		Energy	
arm	DS	PS	DS	PS	DS	PS
real vs. real (reference)	.142	.1394	.156	.0417	.311	.2548
Ours	.103	.1491	.116	.0451	**.288**	**.2504**
MSDformer†	**.089**	.1495	**.112**	.0346	–	–
Diffusion-TS	.159	.1525	.137	**.0343**	.323	.2585
TimeFlow	.116	.1522	.128	.0351	.312	.2524
WaveletDiff	–	–	.101	.0355	.331	.2512
KoVAE	.259	.1578	.116	.0351	.366	.2508
TimeBridge	.197	.1530	.210	.0361	–	–
PaCoDi	.117	.1501	–	–	–	–
TimeVAE	.192	.1589	.190	.0495	.499	.2923
TimeGAN	.383	**.1482**	.483	.0608	.435	.2548

Three readings, none of which we could have obtained by adding a metric and assuming it agreed.

The two metrics rank arms differently. On ETTh1, PS ranks TimeGAN *best* of every arm (.1482) while DS ranks it *worst* (.383, i.e. 88% classifier accuracy). On Energy, PS ranks KoVAE second (.2508) and DS ranks it second worst (.366). A reader given both numbers learns which of two disagreeing instruments to believe only from evidence outside them.

PS routinely rates generated data above real held-out data. On Stock every arm except TimeVAE and TimeGAN scores below the real-vs-real reference, and on Energy so do four of seven. Our own arm is among them. A one-step MAE measures the utility of the synthetic data for training a predictor tested on real data, which sample size, coverage and regularisation also affect, and a generator that smooths or repeats can score well on it; it is a utility score, not a fidelity score, and we do not report it as one.

A second metric from the same family does not fix the copying problem. On ETTh1 the field's own DS ranks the memorisation-gated MSDformer **first** of all arms (.089), ahead of us (.103) and below the real-vs-real reference (.142), and on Stock second (.112). This is the same failure our RFF-C2ST shows in Table 2, for the same reason, and it is why the paper carries a memorisation gate rather than a longer list of scores.

S Candidate Layers That Failed Their Gates

Each detector must pass a positive control before it is used: fitted on the training split, it must measurably remove an injected perturbation of the kind it claims to repair, and it must be inert where its structure is absent. Three candidates failed; they and their diagnostic outcomes are documented here.

A per-window order-2 recurrence was the first candidate for the generative-family slot. Injecting a perturbation and re-fitting reduced it by a factor of 1.00: the fit absorbed the perturbation into its own coefficients, so the control could not fail, which makes it a tautological non-repair rather than a repair. It was dropped and replaced by the single-sinusoid family rule, whose fit is over a two-parameter family that cannot absorb an arbitrary perturbation.

A moving-average mix and a low-order DCT truncation were both candidates for the MuJoCo roughness slot. Both failed their front gates – neither moved the target increment-sd ratio into its pre-specified band on held-out windows – before the roughness law passed one.

The roughness law itself failed the gate on fMRI, where the shape layer had left a residue in the increment and zero-crossing statistics and roughness was the obvious next repair. Fitted on the training split and applied to *real* held-out windows, it moved them .052 away from the null – more than the .046 it appeared to buy on our own samples. We therefore do not use it on fMRI, and we report the remaining fMRI residue as diffuse rather than naming a further law for it (Sec. 8).

The projection form of the family layer failed the memorisation gate (memorised fraction .262 against the .10 threshold; copy-rate .405 against real data's own .008) and scored worse than the draw form (.600 against .503). Its latents are therefore drawn from a Gaussian-copula cloud fitted to the training distribution instead of transplanted.

T Design Details the Results Depend On

Preprocessing. D is a dequantiser, not a Gaussianiser: each channel's distinct training values define a grid, a raw value is mapped to a uniform draw inside that value's Voronoi cell, and decoding rounds to the nearest grid value. It is exactly invertible on the grid and is fitted on the training split only. Channels are first mapped to $[0, 1]$ by the min and max of the whole series before splitting, then to $[-1, 1]$; generated values outside $[0, 1]$ are clipped, and we report the clipped-cell fraction for every layer that can produce one. The claim that all quantities lie in $[-1, 1]$ is a claim about the transformed *data*: the Gaussian source is unbounded, and the sampler's output is clipped at the decode. Every arm we score, ours and every baseline, passes through the same decode and the same grid, so the grid is not something we apply only to ourselves.

What the detection statistics are, and are not. The thresholds in Table 1 are *screening* thresholds evaluated on training windows, not calibrated significance tests, and we do not report p-values for them. The reason is concrete. The third-moment rule computes $|t| > 3$ over stride-1 windows, which are dependent, and screens fifty channels without correction; at the window level $|t| > 3$ alone holds on $44/50$ fMRI channels (median $|t| = 11.2$), and the shipped rule, which also requires the same sign on both halves of the training split, fires on $31/50$ (Table 1). Recomputed at the level at which the units are independent – one mean per training trajectory, sixteen of them (the validation split has seventeen, hence the seventeen fold groups of App. F) – the same statistic clears an uncorrected two-sided 5% threshold on $25/50$ channels and a Bonferroni-corrected one on $14/50$ (median $|t| = 2.1$), while a Gaussian sample with the same shape and the same trajectory labels clears the corrected threshold on $0/50$. So the effect is real and survives correct clustering, but the shipped $|t| > 3$ is a screen whose validity rests on the two-half same-sign stability requirement and on the layer's own positive and negative controls – not on the t distribution; it also fires on Sines and MuJoCo, where the layer is not installed (Table 1). We say the same of every other detector threshold.

The two student stages. S_1 is trained on (source $\to$ teacher state at t^*) and S_2 on (teacher state at $t^* \to$ teacher state at the data end), but at inference S_2 is applied to S_1's output, not to a teacher state. We do not correct for that exposure shift. Held-out regression errors on ETTh1 are 2.2×10^{-3} for S_1 and 9.6×10^{-4} for S_2 on its own inputs; the quantity that governs the reported samples is the *composite* error of $S_2 \circ S_1$ against the teacher's data-end state, which we measure on held-out pairs per skeleton direction as a diagnostic; the Wiener coefficient itself is selected by the internal-subset C2ST criterion of App. U, not fitted to this error.

Time endpoints. The teacher is trained on $t \sim \mathcal{U}[0.01, 0.99]$ while the sampler runs from $t = 1$ to $t = 0$, so the residual is evaluated outside its training interval at both ends. The skeleton is exact there regardless, and the students never see t at all; what is extrapolated is the residual on $[0, 0.01)$ and $(0.99, 1]$, where the teacher's targets are noisiest. We flag this as a real extrapolation rather than defending it: on the datasets where the residual is measurably inert, it cannot matter, and on the two where it helps we have not isolated how much of its contribution comes from those two intervals.

Steps, network calls and duplicates. Where we write "passes" we mean network calls. A midpoint or Heun step costs two calls, so for the baselines that use one we count calls and not steps; the adversarial and variational generators use one call and are not iterative samplers at all. The duplicate-window rate is the fraction of a generator's emitted windows that belong to a group of two or more exactly identical windows, measured after grid-rounding in the decoded units, over the same number of windows for every arm. A finite-support distribution can duplicate itself legitimately, which is why the statistic is read against real data's own value on the same grid (0.0% on all six); on every arm it flags here the memorised fraction exceeds .10 as well (App. F).

Training and selection cost. Per dataset and seed: teacher 18k steps (36k on ETTh1), generation of 32,768 distillation pairs at $K{=}100$, and 8k steps for each student – a few minutes on one RTX 4090, against hours for the diffusion baselines. Selections made on an internal subset of the training split: the student metric, the teacher step count and the Wiener coefficient. Selections without a

held-out rule, recorded in Table 6: the level-and-scale variant (the copula cloud replaces the exact assignment where the exact transplant trips the memorisation gate – a rule), the skeleton family (Toeplitz, centred or sample covariance – chosen per dataset without a rule), the installation flags, and Sines' active-direction teacher loss. The architecture and optimiser are shared across datasets; training budgets are listed in Table 6. Fast inference does not make training free, and we report both.

U Algorithmic Specification

This appendix specifies the released pipeline step by step in the order the operations are applied; every threshold and constant is the value implemented, and the per-dataset configuration is that of Table 6. A window is $x \in \mathbb{R}^{L\times C}$, $L = 24$, $n = LC$, entries $x_{t,c}$; Z-units are $x = 2\,\mathrm{D}(\text{raw}) - 1$; the skeleton is $\mathcal{N}(\mu, \Sigma)$, $\Sigma = V\Lambda V^\top$, eigenvalues λ_k; the source is $\mathcal{N}(0, I)$ in every reported run ($\lambda_k^0 \equiv 1$, $\mu_0 = 0$); $t = 0$ is the data end, $t = 1$ the source end, $D_k(t) = (1-t)^2\lambda_k + t^2$, $\Phi_k(t_2, t_1) = \sqrt{D_k(t_2)/D_k(t_1)}$, $m_t = (1-t)\mu$, $w = V^\top(z - m_t)$. For one window and channel, m and s are the mean and standard deviation over t and $z = (x - m)/(s + 10^{-9})$ its standardised shape.

Construction and training

1. **Preprocessing.** Each raw channel is mapped to $[0, 1]$ by its minimum and maximum over the *whole* series, before splitting (Sines is already in $[0, 1]$ and is not rescaled); stride-1 windows are assigned to the training split/validation/test by 7-day blocks (ETTh1, Stock, Energy) or by trajectory. D is fitted on the training split only: for channel c the sorted distinct training-split values $u_1 < \cdots < u_M$ define cells $[\ell_j, h_j]$ bounded by midpoints of consecutive values (end cells mirror the end gaps); D sends a value in cell j to $\ell_j + (0.01 + 0.98\,U)(h_j - \ell_j)$, $U \sim \mathcal{U}[0, 1]$ from a generator seeded with 0 whenever it is applied to data, and D^{-1} rounds to the nearest u_j (identity on Sines, where rounding is switched off). Training data are $x_1 = 2\mathrm{D}(\text{raw}) - 1$; in the centred mode (Stock) they are the standardised shapes z of $\mathrm{D}(\text{raw})$, per window and channel.
2. **Skeleton**, zero gradient steps. *Toeplitz* (ETTh1, fMRI): with $\bar\mu_c$ the mean of channel c over all the training split cells, $\Gamma(k)_{cd} = \frac{1}{N(L-k)}\sum_i \sum_{t<L-k}(x_{i,t,c} - \bar\mu_c)(x_{i,t+k,d} - \bar\mu_d)$, $k = 0, \ldots, L-1$, $\Gamma(0)$ symmetrised; $\Sigma_{(i,c),(j,d)} = \Gamma(j-i)_{cd}$ for $j \geq i$, $\Gamma(i-j)_{dc}$ otherwise, symmetrised, plus a ridge rI, $r = \max(0, 10^{-3}q - \lambda_{\min})$, $q = \mathrm{mean}_c[\Gamma(0)_{cc} - \Gamma(L-1)_{cc}]$; $\mu = \bar\mu$ tiled over t. *Sample covariance* (Energy, Sines, MuJoCo): μ the per-cell mean, S the unbiased sample covariance of the n-vectors, $\Sigma = V\mathrm{diag}(\max(\lambda_k, 10^{-10}\lambda_{\max}))V^\top$ from its eigendecomposition (floored: 0, 90/120, 116/336 directions). *Centred* (Stock): the same on the standardised shapes (6/144 floored). Then $\lambda_k \leftarrow \max(\lambda_k, 10^{-12})$; the source is drawn as i.i.d. standard normals; the field is Prop. 3.1.
3. **Detectors**, on the training split only, in the order applied. (a) *Difference pairs*: for every ordered pair (i, j), $i \neq j$, least squares of $x_{t,j}$ on $(x_{t,i} - x_{t-1,i}, 1)$ over all cells with $t \geq 1$; accepted if the maximum absolute residual is $< 10^{-4}$; j keeps its best i. (b) *Roughness* (MuJoCo-law flag; source channels of (a)): per-window $r = \mathrm{sd}_t(x_{t+1} - x_t)/s$; the training split's sorted r within each of 8 quantile bins of s. (c) *First-step jump* (target channels of (a)): $d = (x_0 - x_1)/\mathrm{sd}(x_{1:})$, stored as in (b), binned by $\mathrm{sd}(x_{1:})$. (d) *Sinusoid*: mean over training windows of the best R^2 of $A\cos\omega t + B\sin\omega t + c$ over 240 values $\omega \in [0.01, \pi]$ exceeds .999. The installed family is $x_t = (\sin(\omega t + \phi) + 1)/2$: nearest members on a 400×241 grid ($\omega \in [.002, .40]$, $\phi \in [-.4, .8]$) for the first 800 training windows fix the occupied range, widened by 5 grid steps and re-gridded to 900×450; every training window's nearest (ω, ϕ) forms a cloud stored as a Gaussian copula (correlation of normal scores of ranks, empirical marginals). The detector's three-parameter fit is broader than the installed two-parameter family, which fixes amplitude and offset, so a high R^2 certifies membership of the broader family only; the installed family is a Sines-specific configuration. (e) *Constant runs*: a fraction $\geq .15$ of exactly-zero increments over all the training split cells, unless a knot channel; law: $p_c = \#\text{runs}/\#\text{opportunities}$ from a left-to-right scan (a run counts once; scanning resumes after it) and the multiset of run lengths. (f) *Knots*: $|\Delta^2 x| \leq 10^{-5}$ in $\geq 60\%$ of cells and, over windows with any nonzero second difference, positions $(p+1) \bmod 6$ in one residue class in $\geq 95\%$ of them; law: the station grid (unique values at lattice positions $\phi, \phi + 6, \ldots,$ ϕ the modal residue over the union of knot channels, per window) and the training split's probability that consecutive knots are equal. (g) *Single-cell*

extremes (event flag): $P_{\text{train}}(\max_t |z_{t,c}| > 3) \geq .01$, knot channels excluded; law: the histogram over t of $\arg\max_t |z_{t,c}|$ in those windows. (h) *Third moment* (shape flag): with the within-window skew $\text{mean}_t z^3$ and the training windows split into two halves by group label (first half of the sorted labels vs. the rest), a channel fires if $|t| > 3$ on both halves with the same sign, the law if $\geq C/2$ channels do; a_c inverts, at channel c's mean training-split skewness, the curve $a \mapsto$ mean skew of a Gaussian window after $z \mapsto z + a(z^2 - 1)$ and re-standardisation (121 points on $[-.3, .3]$, 20,000 windows). (i) *Tied variates*: no run-time rule; $x_{\cdot,27} := x_{\cdot,26}$ is hard-coded for Energy, the equality verified offline on every cell. Measured on all six training splits: (a) MuJoCo 7 pairs ($x_{7..13}$ from $x_{0..6}$); (d) Sines $5/5$; (e) ETTh1 $\{5\}$, Energy $\{0\text{–}10, 13\text{–}16, 18, 19\}$; (f) Energy $\{20\text{–}25\}$; (g) ETTh1 $7/7$, Stock $6/6$, Energy $20/28$, Sines $0/5$, MuJoCo $7/14$, fMRI $34/50$; (h) Sines $5/5$, MuJoCo $9/14$, fMRI $31/50$; nothing else fires. Which transforms each dataset applies is recorded in Table 6, so (g) outside Energy and ETTh1 and (h) outside fMRI are detected but not installed. The recurrence layer, the projection forms of the sinusoid layer and the unconditional spike layer present in the released code are used by no reported run.

4. **Teacher.** T_θ is the network of App. K, output zero-initialised; per step 128 training windows x_1, $x_0 \sim \mathcal{N}(0, I)$, $t = 0.01 + 0.98\,U$, $z_t = (1-t)x_1 + tx_0$, and Eq. (4) averaged over batch and all n coordinates; Adam, rate 6×10^{-3}, $\beta = (0.9, 0.96)$, 18k steps (36k on ETTh1). On Sines the residual is first projected onto the eigendirections with $\lambda_k \geq 10^{-6}\lambda_{\max}$ (20 of 120) and the loss averaged over those.

5. **Pairs.** 32,768 training and 4,096 held-out sources are integrated at $K = 100$, $t_k = 1 - k/K$, by $w_{k+1} = \Phi(t_{k+1}, t_k) \odot \{w_k + (t_{k+1} - t_k)V^\top T_\theta(z_k, t_k)\}$ with $z_k = m(t_k) + Vw_k$, recording the states $z(t^*)$ at $k = 80$ ($t^* = 0.2$) and $z(0)$ at $k = 100$ (the data-end state; z_0 in Lemma 4.1 is the source draw).

6. **Students.** S_1 (same network, no time conditioning, output weights $\mathcal{N}(0, 0.02^2)$, zero bias) regresses source$\to$ $z(t^*)$ with plain MSE; S_2 regresses $z(t^*) \to z(0)$ under one of two metrics. *Weighted* (ETTh1, fMRI): $\text{mean}_k[\omega_k((\hat{y} - y)^\top V)_k]^2$, $\omega_k = 1/\sqrt{\max(\lambda_k, 10^{-2}\lambda_{\max})}$. *Whitened* (Energy, Sines, MuJoCo): coloured output $\hat{y} = \mu + \Sigma_f^{1/2}\text{net}(x)$, $\Sigma_f^{1/2} = V\text{diag}(\sqrt{\max(\lambda_k, 10^{-6}\lambda_{\max})})V^\top$, loss $\text{mean}_k((\hat{y} - y)^\top V)_k^2 / \max(\lambda_k, 10^{-6}\lambda_{\max})$. Stock trains one map source$\to$ $z(0)$ with the weighted metric and S_2 is the identity. Each: 8,000 steps, batch 128, the teacher's optimiser.

7. **Target cloud** for level and scale: $\mathcal{V}(x) = (m_1, \ldots, m_C, \log(s_1 + 10^{-6}), \ldots) \in \mathbb{R}^{2C}$ of a window in $[0, 1]$ units. *Exact* (ETTh1, Energy, Sines, fMRI): the $\mathcal{V}$ of n_g random training windows, $n_g = \min(4096, |\text{val}|)$ the number of generated windows (4096, 4096, 3333, 2832; all of the training split where $|\text{train}| = n_g$). *Copula* (Stock, MuJoCo): n_g draws (1015, 3333) from a Gaussian copula of the training split's cloud (normal scores of per-coordinate ranks, their covariance $+10^{-8}I$, each coordinate mapped back through its empirical quantile function).

8. **Wiener coefficient.** $S_\alpha = \Sigma(\Sigma + \alpha\bar{\sigma}I)^{-1}$, $\bar{\sigma} = \text{median}_k \Sigma_{kk}$, applied in Z-units about μ (shape units on Stock). α is chosen on a per-dataset grid – ETTh1 $\{0, .001, .002, .003, .005, .008\}$, Stock $\{0, .0005, .001, .002, .003\}$, Energy $\{.003, .005, .01, .03\}$, Sines and fMRI $\{.0005, .001, .002, .003, .005, .008\}$, MuJoCo $\{.1, .2, .3, .5\}$ – by running the full inference chain once per value and minimising the RFF-C2ST accuracy (App. F; two classifier seeds, grouped folds) between the output and an internal subset of the training split – windows not used as level-and-scale targets, which did enter the skeleton, detector and teacher fits – namely the training windows outside the exact cloud (ETTh1 834, Energy 1559) or, when fewer than 200 remain, the windows of a random 20% of the training split's group labels (Stock 170, Sines and MuJoCo 666, fMRI 531). Selected: ETTh1 .001–.005 across seeds, Stock .002, Energy .003, Sines .0005–.005, MuJoCo .5, fMRI .008.

9. **Hold and knot calibration** (Energy; the hold multiplier also on ETTh1), on the arm's own base at the first pass of the chain (the first grid value above) and cached: per hold channel $m_c \in [0, 3]$ by 8 bisection steps so that the zero-increment fraction after the hold layer and rounding equals the training split's ($m_c = 0$ if the base already has at least the training split's; selected $m_c \in [0, 1.03]$ on Energy, $m_5 \in [0.28, 0.31]$ across seeds on ETTh1); per knot channel $q_c \in [0, .6]$ by 8 bisection steps so that the joint layer's probability of equal consecutive knots equals the training split's ($q_c = 0$ if snapping suffices; selected $q_{20} = .011$, $q_{24} = .247$, others 0).

INFERENCE

In this order; a step not named for a dataset is the identity there.

1. n_g sources $z \sim \mathcal{N}(0, I_n)$ (generator seed 999); $z \leftarrow S_2(S_1(z))$ (Stock: the single map; fMRI's reported arm uses no network: $z \leftarrow \mu + V\mathrm{diag}(\sqrt{\lambda})V^\top z$, the skeleton's exact endpoint).

2. *Box projection* (Sines): 20 rounds of clamp to $[-1, 1]^n$ then orthogonal projection onto $\mu_b + \mathrm{span}\{v_k : \lambda_k \geq 10^{-10}\lambda_{\max}\}$ of the training split's Z-unit sample covariance (30 of 120 directions), and a final clamp.

3. *Decode*: $X_s = \mathrm{clip}(\mathrm{D}^{-1}((z+1)/2), 0, 1)$. Stock: reshape to (L, C), re-standardise each channel of each window, borrow (m, s) from the cloud by a random bijection (seed 3), $X_s = \mathrm{clip}(\mathrm{D}^{-1}(\mathrm{clip}(m + sz, 0, 1)), 0, 1)$.

4. *Wiener*: $X_W = \mathrm{clip}(\mathrm{D}^{-1}(\frac{1}{2}[S_\alpha(2\mathrm{D}(X_s) - 1 - \mu) + \mu + 1]), 0, 1)$: re-dequantise (seed 0), shrink, decode again. On Stock S_α acts on the raw student output in shape space and step 3 is repeated. $\alpha = 0$ is the identity (selected nowhere).

5. *Spike injection*: off in every reported run. An earlier ETTh1 configuration had left a default injection rate (.10) active, injecting $\pm 6\hat{\sigma}_c$ single-cell spikes into 10% of windows before the assignment; this was found while deriving this appendix from the code, and the ETTh1 arm was re-run with the flag off and, since the single-cell-extremes detector fires on all seven ETTh1 channels, with that calibrated layer and the hold layer installed (Table 6). The injected arm had scored .571, the plain re-run scores .579, the reported arm .560.

6. *Level-and-scale assignment*: $\mathcal{V}(X_s)$ (the *pre-Wiener* decode) and the cloud are standardised by the cloud's per-coordinate mean and sd; a square linear assignment minimising the summed squared Euclidean distance gives each window a cloud vector idx_i; $X_O = \mathrm{clip}(\mathrm{D}^{-1}(\mathrm{clip}(m_{\mathrm{idx}} + s_{\mathrm{idx}} \odot z, 0, 1)), 0, 1)$ with z the shape of the step-5 window, so every output carries one cloud $(m, \log s)$ pair before clipping and rounding.

7. *Post-sampler chain*, code order. (a) **hold** (Energy, ETTh1): per hold channel scan $t = 1..L-1$; with probability $p_c m_c$ draw ℓ from the training split's run-length multiset, set $x_t, \ldots, x_{t+\ell-2} := x_{t-1}$, resume at $t + \ell$; then round to the grid and clip. (b) **single-cell extremes** (Energy, ETTh1), in continuous units $\tilde{X} = \mathrm{D}(X)$: window i is eligible for channel c if its own $\max_t |z_{t,c}| \leq 3$ while its assigned training window idx_i has $z^* = \max_t |z_{t,c}| > 3$; $\mathrm{round}([P_{\text{train}}(\max|z_c| > 3) - P_{\text{gen}}(\max|z_c| > 3)]n_g)$ of them, if positive, are chosen at random and each gets *one* cell at $t \sim$ the position law, $\tilde{x}_{t,c} := \mathrm{mean}(\text{rest}) \pm \mathrm{sd}(\text{rest})\, z^* \sqrt{L/(L - 1 - z^{*2})}$ (sign of the borrowed extreme, z^* capped at 4.75), so that the cell's standardised value equals the borrowed training window's extreme; only written cells are decoded back. (c) **knot** (Energy): per window a phase $\phi \in \{0..5\}$; knots at $\phi, \phi + 6, \ldots$ take the current values, plus outside knots at $\phi - 6$ and $k_{\text{last}} + 6$ by linear extension through the end values; knot values snapped to the station grid; with probability q_c a knot copies its predecessor; linear interpolation, clip, no re-rounding. (d) **tied variates** (Energy): $x_{\cdot,27} := x_{\cdot,26}$. (e) **roughness** (MuJoCo, channels 0–6): split $x - m$ into least-squares linear trend and residual; target r^* is the training split's quantile, within the window's s-bin, at the window's rank among generated windows of that bin (outside the training split's range r is kept); 30 bisection steps on $g \in [0, 4]$ give the window $m + s \cdot \mathrm{std}(\text{trend} + g\,\text{resid})$ with ratio r^*. (f) **difference law** (MuJoCo): $x_{t,j} := \mathrm{clip}(a(x_{t,i} - x_{t-1,i}) + c, 0, 1)$, $t \geq 1$, on the 7 pairs; $x_{0,j}$ untouched. (g) **first step** (MuJoCo): $d = (x_{0,j} - x_{1,j})/\mathrm{sd}(x_{1:,j})$ transported as in (e); $x_{0,j} := x_{1,j} + d^* \mathrm{sd}(x_{1:,j})$. (h) **sinusoid draw** (Sines): per channel (ω, ϕ) is drawn from the copula cloud (generator = post-sampler seed) and $x_{t,c} := (\sin(\omega t + \phi) + 1)/2$ replaces the channel; with all five channels detected the saved Sines sample is a function of the cloud and the seed only, steps 1–6 entering only through the sample count. (i) **shape** (fMRI, 31 channels): $w = z + a_c(z^2 - 1)$ re-standardised over t, $x := \mathrm{clip}(m + sw, 0, 1)$.

8. No final re-rounding (flag off in every reported run); output in $[0, 1]$ units. The saved sample uses post-sampler seed 0; the in-run score averages 4 such seeds (2 on Stock, 8 on the trajectory datasets). Tables 2, 3, 10 and 11 are scored from the saved samples; the in-run averages appear only in the released logs.

Table 13: The layers as implemented. "fires" is the measured detection on the training split, "installed" the reported configuration; "preserves" holds before the final clipping and grid rounding.

layer	detection rule (code)	fitted on the training split	transform	preserves	installed on
covariance skeleton	always	μ, Σ: Toeplitz + ridge, or sample cov. floored at $10^{-10}\lambda_{\max}$	source $\mathcal{N}(0, I)$, field u_{skel}, propagator Φ	–	all six
level and scale	always	$(m, \log s)$ cloud: training-split subset or Gaussian copula	square assignment on standardised $(m, \log s)$; transplant onto the window's shape	within-window shape	all six (copula: Stock, MuJoCo)
Wiener	always; α by internal-subset C2ST	$\Sigma, \mu, \bar{\sigma}$; α on an internal training subset	$\mu + \Sigma(\Sigma + \alpha\bar{\sigma}I)^{-1}(x - \mu)$, Z-units	–	all six
hold	$\geq$ 15% exactly-zero increments, non-knot	p_c, run lengths; m_c on the arm's base	random runs $x_t..x_{t+\ell-2} := x_{t-1}$, then rounding	cells outside runs	Energy $17/28$, ETTh1 $1/7$
knot	$\lvert\Delta^2 x\rvert \leq 10^{-5}$ in $\geq$ 60% cells; nonzero on one class mod 6 in $\geq$ 95% windows	station grid, P(equal); q_c on the base	lattice-6 knots (random phase) snapped to grid, linear between	end values	Energy $6/28$
single-cell extremes	$P_{\text{train}}(\max \lvert z_c\rvert > 3) \geq .01$, non-knot	rate, position histogram	one cell per eligible window, set to the assigned training window's extreme (cap 4.75); deficit-limited	all other cells	Energy $20/28$, ETTh1 $7/7$ (fires on 5 of 6)
tied variates	none (hard-coded)	–	$x_{\cdot,27} := x_{\cdot,26}$	–	Energy
difference constraint	exact pairs $x_{t,j} = a\Delta x_{t,i} + c$, max error $< 10^{-4}$	(i, a, c) per pair	overwrite $x_{t\geq 1,j}$ from generated $x_{\cdot,i}$	positions	MuJoCo 7 pairs
roughness	by flag, source channels of the pairs	r quantiles in 8 scale bins	rescale deviation from the linear trend (bisection) to r^*	m, s per window	MuJoCo $7/14$
first-step jump	by flag, target channels of the pairs	jump quantiles in 8 scale bins	$x_{0,j} := x_{1,j} + d^*\text{sd}(x_{1:,j})$	$x_{1:,j}$	MuJoCo $7/14$
sinusoid family	mean best single-sinusoid $R^2 > .999$	(ω, ϕ) copula on the 2-parameter grid	replace by $(\sin(\omega t{+}\phi){+}1)/2$, (ω, ϕ) drawn	nothing	Sines $5/5$
third moment	$\lvert t\rvert > 3$ on both halves, same sign; $\geq C/2$ channels	a_c from the Gaussian curve	$z \mapsto z + a_c(z^2 - 1)$, re-standardised, rescaled	m, s per window	fMRI $31/50$ (fires Sines $5/5$, MuJoCo $9/14$)

V REPRODUCIBILITY

(Model and optimiser details are in App. K.)

All numbers come from a single pipeline invoked per dataset with the configuration of Table 6; each reported cell averages three seeds, five on ETTh1 (the fMRI arm has no network and is deterministic up to the layer seed). Baselines are the authors' official repositories, run with their own published recipes where one exists for the dataset; cells where a repository ships no recipe for a dataset are marked § in Table 2 and reported for completeness only, never as that method's number. Every baseline export carries a record of the space its samples live in, and the scorer refuses to run without it – three separate scoring errors in this project traced to silently assumed units. Timings are on one RTX 4090 with a Ryzen CPU. We report three quantities separately rather than one, because they scale differently.

- **Network time.** 0.002 s per 1,000 windows for our at most two passes, measured the same way as the 0.6–4.1 s per 1,000 windows of the 100–1000-step baselines. This is the quantity the "one or two passes" claim is about, and it is the only one that is two to three orders of magnitude apart.
- **One-time construction, per dataset.** About two seconds of CPU work: eigendecomposition of the $n \times n$ skeleton covariance, detector evaluation and law fitting. Amortised over every window ever drawn from that dataset.
- **Per-batch post-sampler work.** Most layers are vectorised array operations, but the level-and-scale match is *not*: it is a square linear assignment over the drawn batch, measured at 1.6 s for 4,096 windows on ETTh1, 1.8 s on Energy and 0.9 s for 2,832 windows on fMRI, and it is paid on every batch. It dominates our end-to-end cost.

End to end, therefore, drawing 4,096 ETTh1 windows costs us about 1.6 s of assignment plus 0.01 s of network and a two-second construction paid once, against roughly 2.5–17 s of sampling for the baselines at the same count. The advantage is real but it is 1.6× to 11× end to end (2.5–17 s against 1.6 s), not the three orders the network-time figure alone would suggest, and we state it that way.

Entry points. The released code maps every row of Table 6 and every cell of Table 2 to a single command; the zero-network and teacher ablations of Table 3 are switches of the same pipeline, the detectors and layers are one function each with the threshold of Table 1 as its default, and the scorer refuses to run on a sample file that carries no record of its units.