CoBiST: A Coupled Bidirectional Survival Transport Operator for Probabilistic Time-Series Generative Forecasters

Anonymous authors
Paper under double-blind review

Abstract

Probabilistic time-series forecasting targets the joint distribution of a segmentation of the future path. Modern generators approximate this distribution through sampling full trajectories. Most existing generative forecasters delegate the interaction between forecast positions to a generic backbone and typically lack a unified, normalized accounting of information routing across forecast positions. To capture such internal interaction during generation, we propose Coupled Bidirectional Survival Transport (CoBiST), a generator-agnostic cross-horizon transport operator with explicit probabilistic semantics for directional propagation, locality, and memory mechanisms. CoBiST can be plugged into multiple types of generator without changing the outer objective or sampler, where a shared gate produces directional preferences and one-step survival probabilities to jointly determine the budget assigned to causal and anti-causal recurrent scans and to a direct-local branch. The resulting transport operator is theoretically verified to admit an exact augmented row-stochastic kernel, a linear-time semiseparable realization, frozen-gate non-expansiveness, and a finite-horizon sensitivity bound for state-dependent gates. We instantiate the same operator in GAN, flow-matching, and diffusion forecasters, achieving average Projected CRPS reductions relative to capacity-matched dense controls of 6.72%, 21.48%, and 20.78% across eight datasets, respectively. CoBiST-Flow and CoBiST-Diffusion also achieve the best two average Projected CRPS, Marginal CRPS and Energy Score ranks among all evaluated forecasters.

1 Introduction

Probabilistic time-series forecasting focuses on the conditional distribution of a future path segment given an observed history. A generative forecaster approximates this distribution by sampling full trajectories. Examples include diffusion forecasters such as CSDI (Tashiro et al., 2021), TimeGrad (Rasul et al., 2021) and TimeDiff (Shen & Kwok, 2023), as well as flow-matching models such as TSFlow (Kollovieh et al., 2025). These models define how trajectories are generated, but typically delegate hidden-state information exchange among different forecast positions to a generic sequence backbone, where direction, locality, and memory are not necessarily expressed through a common accounting. This makes the cross-horizon interaction harder to audit and leaves computational scaling dependent on the chosen backbone.

We introduce *Coupled Bidirectional Survival Transport* (CoBiST), a generator-agnostic cross-horizon transport core that jointly handles bidirectional propagation, local retention, and adaptive memory decay through a conserved route simplex, rather than treating memory and routing as independent mechanisms. CoBiST uses a single shared gate to emit direction-preference logits and retention logits at each forecast position. The retentions $\rho^p, \rho^f \in (0, 1)$ are one-step survival probabilities, whose negative logarithms are forgetting hazards. Their geometric mean $q = \sqrt{\rho^p \rho^f}$ is the recurrent-scan budget, and the preferences α^p, α^f further divide it between causal and anti-causal survival scans. The remaining mass is assigned to a direct-local branch. The kernel induced by this architecture gives exact past, local, future, and boundary attributions. Increasing either directional hazard will simultaneously reduce recurrent transport and increase the direct-local probability, providing a common accounting of persistence, direction, and locality.

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.

This conserved coupling turns cross-horizon mixing into an explicit transport operator. We represent each scan by a survival kernel whose coefficients expose intermediate forgetting events and boundary sinks. The semiseparable structure of the kernel enables the transport interaction to be evaluated by two linear scans, rather than a dense horizon-by-horizon matrix. We also separate the frozen-gate transport guarantee from gate feedback: the former is non-expansive, while the latter contributes a finite-horizon correction whose size is controlled by gate regularity. These results state both the useful structural guarantees and the conditions under which they apply.

Consequently, CoBiST has three complementary interpretations. As a transport operator, it is a convex route mixture on an augmented sequence graph. As a memory mechanism, its hazards control survival probabilities and effective memory lengths. As a probabilistic gate, it is a hierarchical categorical distribution with a conserved recurrent-scan budget, while the induced kernel supplies mutually exclusive source-level attributions. These interpretations jointly describe one mechanism, not a post-hoc combination of independent modules.

We evaluate the performance of this operator by inserting the same CoBiST core into three generators: a latent path in a GAN, an interpolated coefficient in a flow-matching model, and a noised state in a diffusion model, without changing their outer objectives and samplers. Across eight datasets, the average reductions in Projected CRPS relative to the corresponding capacity-matched dense controls are 6.72%, 21.48%, and 20.78% for GAN, Flow, and Diffusion, respectively, while CoBiST-Flow and CoBiST-Diffusion obtain the best average Projected CRPS ranks compared to six existing probabilistic forecasters.

Our contributions are as follows. First, we introduce a generator-agnostic cross-horizon transport operator that couples directional propagation, adaptive retention, and recurrent-versus-local routing while remaining interpretable. We then theoretically derive its exact augmented stochastic kernel, semiseparable scan representation, transport-level attribution, and qualified stability properties. Finally, we demonstrate plug-in improvements across GAN, flow-matching, and diffusion forecasters on eight datasets, where CoBiST-Flow and CoBiST-Diffusion reach the best average ranks in comparison to six evaluated time-series probabilistic forecasters. The interpretability of CoBiST transport semantics is further demonstrated through intervention-based validation.

2 Related work

CoBiST integrates directional transmission and adaptive memory into generative probabilistic time-series forecasters. Its contribution is a shared normalized formulation that connects these ingredients and exposes how the transported hidden state is assembled.

Probabilistic time-series forecasting. Probabilistic forecasting methods differ in how they represent predictive uncertainty through distributional summaries, likelihood evaluation, and trajectory sampling. The Temporal Fusion Transformer (TFT) (Lim et al., 2021) predicts horizon-wise quantiles, but is not designed as an explicit sampler of the joint future path. DeepAR (Salinas et al., 2020) instead factorizes the joint distribution into autoregressive conditionals, while TACTiS-2 (Ashok et al., 2024) models cross-position dependence with an attentional copula. Generative forecasters learn the joint conditional distribution of a future path through sampling. TimeGrad (Rasul et al., 2021) combines autoregressive forecasting with conditional diffusion. CSDI (Tashiro et al., 2021) treats the forecast block as missing and jointly samples it with diffusion. TimeDiff (Shen & Kwok, 2023) also uses a non-autoregressive conditional diffusion process, and TSFlow (Kollovieh et al., 2025) uses a flow-matching method with a Gaussian-process prior.

These approaches offer different probabilistic formulations and sampling procedures, while implementing temporal information exchange between forecast positions through their respective neural backbones. CoBiST is complementary: it supplies a common transport core for latent GAN paths, flow-matching states, and diffusion states without changing their outer objectives or samplers. It decouples cross-horizon interaction from the probabilistic formulation, attributing improvements to the shared transport structure rather than generator-level changes.

Directional transmission. Directional transmission mechanisms specify how information is propagated along a temporal axis. Bidirectional recurrent networks (Schuster & Paliwal, 1997) maintain forward and backward states when a complete sequence is available. Standard self-attention (Vaswani

et al., 2017) also allows bidirectional exchange through a row-normalized dense matrix with quadratic complexity in the forecast horizon. Structured state-space models provide efficient alternatives: S4 (Gu et al., 2021) uses a structured linear state-space system, Mamba (Gu & Dao, 2023) introduces input-selective updates, and the state-space-duality framework of Mamba-2 (Dao & Gu, 2024) connects recurrent computation with semiseparable matrices. Hydra (Hwang et al., 2024) develops bidirectional state-space mixing through quasiseparable matrices. CoBiST realizes such a directional transmission mechanism through explicitly interpretable route masses.

Adaptive memory and retention. Adaptive memory mechanisms determine how long propagated information remains influential. LSTM (Hochreiter & Schmidhuber, 1997) introduced multiplicative gated retention. RetNet (Sun et al., 2023) and RWKV (Peng et al., 2023) make exponential decay explicit, while Mega (Ma et al., 2022) feeds features from a learned multidimensional exponential moving average into a gated-attention module. Hyena (Poli et al., 2023) uses long convolutions, and Mamba (Gu & Dao, 2023) and gated linear attention (Yang et al., 2023) make propagation input-dependent. Adaptive Attention Span (Sukhbaatar et al., 2019) learns a dependency range, while Pathformer (Chen et al., 2024) routes across multi-scale pathways. CoBiST further interprets directional retentions as survival probabilities to determine the recurrent-versus-local transport budget.

CoBiST unifies these two mechanisms as complementary components of hidden-state transport, where directional transmission determines how information propagates and adaptive retention determines how long information persists. CoBiST's contribution is not bidirectionality or an adaptive memory mechanism alone, but their probabilistically interpretable coupling within a shared transport operator.

3 Problem setup and CoBiST transport

3.1 Problem setup and evaluation

Given a history x and a forecast horizon H, a conditional generator maps noise to a forecast sample

$$\widehat{y} = G_\theta(\varepsilon, x), \qquad \varepsilon \sim p_\varepsilon.$$

The push-forward distribution $(G_\theta(\cdot, x))_{\#}p_\varepsilon$ is used to approximate the future distribution $p^\star(\cdot \mid x)$.

We use Projected Continuous Ranked Probability Score (PCRPS) as the primary proper scoring rule for its advantages in evaluating the cross-horizon distributional structure. Marginal Continuous Ranked Probability Score (MCRPS), Energy Score (ES) and Variogram Score (VS) are secondary diagnostics. Formal definitions for the above metrics are provided in Appendix D.

3.2 Conservative transport construction

Let $c = H_\theta(x) \in \mathbb{R}^w$ be a fixed-length context vector summarizing x, and $u = (u_1, \ldots, u_H)$ be the sequence state presented to CoBiST, whose semantics depend on the chosen generative interface. Each position is embedded as

$$z_k = E_\theta(u_k) \in \mathbb{R}^w, \qquad (a_k^p, a_k^f, b_k^p, b_k^f) = \Gamma_\theta(z_k, c) \tag{1}$$

where a_k^p, a_k^f are direction-preference logits and b_k^p, b_k^f are retention (hazard) logits. This single gate is the only interface between the conditional generator and the transport operator.

We impose four requirements on the transport. (i) The branch weights must be nonnegative and sum to one. (ii) The recurrent-scan budget must come from directional retention, rather than an independent scale. (iii) Directional branches admit a first-order survival interpretation. (iv) The global dependency should be computable without materializing a quadratic kernel. The branch accounting below satisfies these requirements, and Theorem 1 characterizes the geometric-mean choice under explicit assumptions.

The logits are unconstrained real numbers, so we convert the direction logits into a probability vector through softmax, while using the sigmoid function for each retention logit. The retention can be further represented by a hazard (the forgetting amount on the log scale) expressed using the softplus function:

$$(\alpha_k^p, \alpha_k^f) = \mathrm{softmax}(a_k^p, a_k^f), \quad \rho_k^d = \sigma(b_k^d), \quad \lambda_k^d = -\log \rho_k^d = \zeta(-b_k^d) \geq 0, \quad d \in \{p, f\}, \tag{2}$$

where α_k^p and α_k^f are conditional probabilities given that a recurrent scan branch is selected, and ρ_k^d is the fraction of the direction-d state that survives one step.

The recurrent-scan transport budget q_k is the geometric mean of the two directional retentions,

$$q_k = \sqrt{\rho_k^p \rho_k^f} = \exp\left[-\frac{\lambda_k^p + \lambda_k^f}{2}\right], \tag{3}$$

which has no separate budget parameter. We justify the use of the geometric mean in Theorem 1. The three branch weights are consequently

$$\pi_k^p = q_k \alpha_k^p, \qquad \pi_k^f = q_k \alpha_k^f, \qquad \pi_k^0 = 1 - q_k. \tag{4}$$

The branch vector $\pi_k = (\pi_k^p, \pi_k^0, \pi_k^f)$ can be interpreted as a probability vector since $\pi_k^p + \pi_k^0 + \pi_k^f = 1$. Changing a hazard therefore changes both the recurrent-scan and direct-local branch masses. The corresponding source-position masses are defined from the kernel below and they need not equal the branch probabilities because both scans inject the current state.

3.3 Survival scans and readout

With boundary values $\xi_p, \xi_f \in \mathbb{R}^w$, define

$$P_0 = \xi_p, \qquad P_k = \rho_k^p P_{k-1} + (1 - \rho_k^p) z_k, \tag{5}$$

$$F_{H+1} = \xi_f, \qquad F_k = \rho_k^f F_{k+1} + (1 - \rho_k^f) z_k. \tag{6}$$

Here P_k is a causal (past-to-current) exponentially weighted state and F_k is an anti-causal (future-to-current) state. The factors $1 - \rho_k^d$ inject the current embedding, whereas ρ_k^d carries information from the neighboring state in direction d. Thus the causal and anti-causal labels describe scan direction, not an exclusive partition of source positions: each scan contains a current-state injection. The implementation uses $\xi_p = \xi_f = 0$, interpreted as virtual zero-valued sink nodes. The conservative transport summary and its signed contrast are

$$T_k = \pi_k^p P_k + \pi_k^0 z_k + \pi_k^f F_k, \qquad D_k = \pi_k^f F_k - \pi_k^p P_k. \tag{7}$$

The vector T_k is the state passed to the readout, and D_k records the contrast between the anti-causal and causal scan branches. It is a representation-space contrast, while source-position masses are given by the kernel attribution below. The shared readout is

$$h_k = \mathrm{Fuse}_\theta([z_k, T_k, D_k, T_k \odot D_k]) + C_\theta(c) + e_k, \qquad r_k = \mathrm{Res}_\theta(h_k), \qquad o_k = W_{\mathrm{out}} r_k, \tag{8}$$

where $[\,\cdot\,]$ denotes concatenation, $\odot$ is the Hadamard product, e_k is a learned positional embedding, $C_\theta(c)$ is the broadcast context projection, and W_{out} maps the residual representation to the interface-specific output width. The map Res_θ contains the residual blocks and normalization. This ordering matches the implementation: context and position are added after the four-way fuse projection.

Remark 1 (Hierarchical probability interpretation)**.** *At position k, a Bernoulli decision with probability q_k selects a recurrent scan branch rather than the direct-local branch. Conditional on that event, α_k selects the causal or anti-causal scan. For any probability vector v, write $\mathcal{H}(v) = -\sum_i v_i \log v_i$, with $0 \log 0 = 0$. The branch entropy is*

$$\mathcal{H}(\pi_k) = h_{\mathrm{b}}(q_k) + q_k \mathcal{H}(\alpha_k), \qquad h_{\mathrm{b}}(q) = -q \log q - (1 - q) \log(1 - q). \tag{9}$$

3.4 Generator interfaces

CoBiST is inserted between a generator's state embedding and output head, so the outer model determines the state semantics, training objective, and sampler while CoBiST determines how information is exchanged across the horizon. This separation lets us attribute changes in forecasting quality to the transport structure rather than to a different probabilistic loss. We instantiate the same core in GAN, flow-matching, and diffusion generators and keep each outer objective and sampler standard. Table 1 summarizes the interfaces and Appendix A gives their complete definitions.

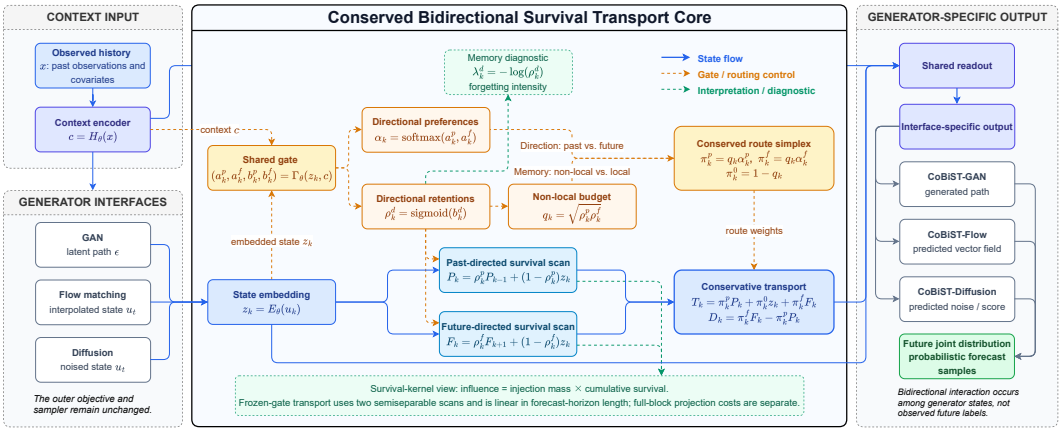

Figure 1: Architecture of CoBiST.

Table 1: Generator-agnostic CoBiST instances.

Instance	CoBiST state	Training target	Sampling
CoBiST–GAN	latent path ε	adversarial discriminator objective	one decoding pass
CoBiST–Flow	interpolated coefficient u_t	flow-matching velocity objective	ODE integration
CoBiST–Diffusion	noised state u_t	velocity v_t target	denoising process

4 Theory and Transport Interpretability

4.1 Standing Assumptions

Assumption 1 (Bounded states)**.** *The horizon H is finite, and all embedded states z_k and boundary values ξ_p, ξ_f have Euclidean norm at most $B < \infty$.*

Assumption 2 (Regular gates)**.** *The gate map Γ_θ is C^1 in the state and context. On the bounded domain in Assumption 1, the induced maps ρ^p, ρ^f, q and $\alpha = (\alpha^p, \alpha^f)$ have finite Lipschitz constants. In particular, all quantities below are finite and the recurrences are well-defined.*

The first results condition on the gate realization, treating ρ_k^d and π_k^d as fixed. We distinguish this frozen-gate kernel from the full state-dependent Jacobian below. Assumption 2 is a bounded-domain condition. The state-dependent result is conditional on globally finite gate constants, for example by bounding layer spectral norms, rather than a claim of global non-expansiveness.

4.2 Conservation and Uniqueness

Proposition 1 (Conservative route decomposition)**.** *For any finite gate logits,*

$$0 \le q_k, \pi_k^p, \pi_k^0, \pi_k^f \le 1, \qquad \pi_k^p + \pi_k^0 + \pi_k^f = \pi_k^0 + q_k = 1, \qquad \frac{\pi_k^p}{\pi_k^f} = \frac{\alpha_k^p}{\alpha_k^f} = e^{a_k^p - a_k^f}. \tag{10}$$

The proposition follows directly from related definition. At each location, the gate values determine route mass and directional odds without an additional post-hoc attribution method.

Theorem 1 (Characterization of geometric-mean coupling)**.** *Let $q : \mathbb{R}_+^2 \to (0, 1]$ be a continuous function of two nonnegative hazards (λ_p, λ_f), where the subscripts again denote the past and future directions. Suppose it satisfies, for all $(\lambda_p, \lambda_f), (\mu_p, \mu_f) \in \mathbb{R}_+^2$, $q(0, 0) = 1$, $q(\lambda_p + \mu_p, \lambda_f + \mu_f) = q(\lambda_p, \lambda_f) q(\mu_p, \mu_f)$, $q(\lambda_p, \lambda_f) = q(\lambda_f, \lambda_p)$, and $q(\lambda, \lambda) = e^{-\lambda}$. Then*

$$q(\lambda_p, \lambda_f) = e^{-(\lambda_p + \lambda_f)/2} = \sqrt{\rho_p \rho_f}, \qquad \rho_d = e^{-\lambda_d}. \tag{11}$$

Together, these results give each forecasting position a finite route budget. The coupling theorem is an axiomatic uniqueness statement under hazard additivity, directional symmetry, diagonal calibration,

and continuity and is not an empirical superiority claim. We further verified the geometric mean can serve as a simpler and more intuitive proxy with comparable accuracy in experiments in Section 5.

Corollary 1 (Hazard sum and directional asymmetry)**.** *At each location k, writing*

$$\lambda_\Sigma = \frac{\lambda^p + \lambda^f}{2}, \qquad \lambda_\Delta = \frac{\lambda^p - \lambda^f}{2},$$

where indices are suppressed for readability, gives

$$q = e^{-\lambda_\Sigma}, \qquad \rho^p = e^{-\lambda_\Sigma - \lambda_\Delta}, \qquad \rho^f = e^{-\lambda_\Sigma + \lambda_\Delta}, \qquad \lambda_\Sigma \geq |\lambda_\Delta|. \tag{12}$$

Thus λ_Σ controls the total recurrent-scan budget and λ_Δ controls memory asymmetry. Moreover,

$$\frac{\partial q}{\partial \lambda^p} = \frac{\partial q}{\partial \lambda^f} = -\frac{q}{2}, \qquad \min(\rho^p, \rho^f) \leq q \leq \frac{\rho^p + \rho^f}{2}. \tag{13}$$

Consequently, λ_Σ is an interpretable scalar for recurrent-scan communication, whereas λ_Δ records directional bias. The derivatives quantify how a small gate change alters the total branch budget.

4.3 Kernel representation

Theorem 2 (Augmented survival transport kernel)**.** *For $1 \leq j, k \leq H$, let A^p_{kj} and A^f_{kj} denote the contributions of the injection z_j to the past and future scans at location k, respectively. Define*

$$A^p_{kj} = \mathbf{1}\{j \leq k\}(1 - \rho^p_j) \prod_{r=j+1}^{k} \rho^p_r, \qquad A^f_{kj} = \mathbf{1}\{j \geq k\}(1 - \rho^f_j) \prod_{r=k}^{j-1} \rho^f_r \tag{14}$$

$$\beta^p_k = \prod_{r=1}^{k} \rho^p_r, \qquad \beta^f_k = \prod_{r=k}^{H} \rho^f_r. \tag{15}$$

The factors β^p_k, β^f_k are the amounts of boundary mass that survive to location k, while the indicator functions describe the causal ($j \leq k$) and anti-causal ($j \geq k$) propagation. Then

$$P_k = \beta^p_k \xi_p + \sum_{j=1}^{H} A^p_{kj} z_j, \qquad F_k = \beta^f_k \xi_f + \sum_{j=1}^{H} A^f_{kj} z_j. \tag{16}$$

As each branch's real-node coefficients plus its virtual boundary remainder sum to one, the expansion is an exact convex decomposition.

The kernel coefficients are path-level explanations. A^p_{kj} and A^f_{kj} are the exact amounts injected at j that survive to k, and their products expose every intermediate forgetting event. The residual β^d_k is the mass still attached to the corresponding virtual boundary. A practitioner can therefore trace an output to individual horizon positions and quantify how much information remains outside the observed window.

Proposition 2 (Row conservation of the augmented kernel)**.** *Introduce two virtual boundary nodes 0 and $H + 1$, carrying values ξ_p and ξ_f. For real nodes $1, \ldots, H$, define*

$$K_{kj} = \pi^p_k A^p_{kj} + \pi^0_k \mathbf{1}\{j = k\} + \pi^f_k A^f_{kj}, \tag{17}$$

with $K_{k0} = \pi^p_k \beta^p_k$ and $K_{k,H+1} = \pi^f_k \beta^f_k$. Then

$$T_k = K_{k0} \xi_p + \sum_{j=1}^{H} K_{kj} z_j + K_{k,H+1} \xi_f, \qquad K_{k0} + \sum_{j=1}^{H} K_{kj} + K_{k,H+1} = 1. \tag{18}$$

Consequently, the augmented matrix (including columns 0 and $H + 1$) is row-stochastic, whereas the real-node submatrix is only substochastic when zero-boundary sinks receive nonzero mass.

The recurrence defines an affine transport operator on an augmented sequence graph. The branch vector gives normalized routing probabilities for the three branches. Exact source-position contributions are given by the induced kernel coefficients, which can be aggregated into past, local, future, and boundary masses. For clarity, define the source-level attributions at position k by

$$\mu_k^- = \sum_{j<k} K_{kj} = \pi_k^p(\rho_k^p - \beta_k^p), \qquad \mu_k^0 = K_{kk} = \pi_k^0 + \pi_k^p(1-\rho_k^p) + \pi_k^f(1-\rho_k^f), \tag{19}$$

$$\mu_k^+ = \sum_{j>k} K_{kj} = \pi_k^f(\rho_k^f - \beta_k^f), \qquad \mu_k^{\partial p} = K_{k0} = \pi_k^p\beta_k^p, \qquad \mu_k^{\partial f} = K_{k,H+1} = \pi_k^f\beta_k^f. \tag{20}$$

These quantities satisfy $\mu_k^- + \mu_k^0 + \mu_k^+ + \mu_k^{\partial p} + \mu_k^{\partial f} = 1$. Thus μ_k^-, μ_k^0, and μ_k^+ are mutually exclusive real-node past, local, and future masses, whereas $\boldsymbol{\pi}_k$ is the branch-level routing distribution.

This augmentation is important for faithful explanations. With zero boundaries, the real-node matrix is substochastic as probability mass is sent to a zero sink. Including the two virtual nodes restores row stochasticity and prevents boundary mass from being interpreted as missing source attribution.

4.4 Semiseparable structure

Theorem 3 (Rank-one semiseparable decomposition and scan cost)**.** *Let $s_k^p = \prod_{r=1}^{k} \rho_r^p$ and $u_k^f = \prod_{r=1}^{k-1} \rho_r^f$ be cumulative survival products (the empty product gives $u_1^f = 1$). Then*

$$A_{kj}^p = \mathbf{1}\{j \le k\}\,(1-\rho_j^p)\frac{s_k^p}{s_j^p}, \qquad A_{kj}^f = \mathbf{1}\{k \le j\}\,(1-\rho_j^f)\frac{u_j^f}{u_k^f}. \tag{21}$$

Hence the frozen-gate kernel is a diagonal term plus one lower and one upper rank-one semiseparable term. Applying this transport kernel to w-dimensional states requires $O(Hw)$ arithmetic operations and $O(Hw)$ working memory. This statement concerns the transport operation itself.

The semiseparable form gives CoBiST global reach without a dense sequence-by-sequence attention matrix. Products of retentions are evaluated with two linear scans, so "rank-one" refers to each triangular factor, not to the complete kernel.

The same factorization is also an explanation interface: the lower and upper scans separately expose past and future survival, while the diagonal term exposes the direct-local branch together with the current injections of the two scans. Hence global temporal reach and linear-time evaluation do not come at the cost of an opaque dense attention map.

Table 2: Theoretical per-example cost of the implemented transport block. d_{in} is the input channel count and L is the number of residual blocks. The Dense column describes the shared-prefix MLP control used in this paper, whose masked input is materialized before its first projection.

Operation	CoBiST	Dense prefix control
State embedding and gate/context MLPs	$O(Hd_{\text{in}}w + Hw^2)$	$O(Hw^2)$
Survival/transport interaction	$O(Hw)$	$O(H^2 d_{\text{in}}w)$
Fuse, L residual blocks, output	$O(HLw^2)$	$O(HLw^2)$
Total (fixed d_{in}, w, L)	$O(Hd_{\text{in}}w + H(L+1)w^2)$	$O(H^2 d_{\text{in}}w + H(L+1)w^2)$

4.5 Memory and stability

Theorem 4 (Frozen-gate non-expansiveness)**.** *Fix the retention values $\rho = (\rho^p, \rho^f)$, route weights $\boldsymbol{\pi}$, and boundaries ξ_p, ξ_f. For two input sequences $Z = (z_j)_j$ and $Z' = (z'_j)_j$, define their worst per-position perturbation by $\delta = \max_j \|z_j - z'_j\|$. Then for any k,*

$$\|P_k - P'_k\|,\ \|F_k - F'_k\|,\ \|T_k - T'_k\| \le \delta, \qquad \|D_k - D'_k\| \le q_k\delta. \tag{22}$$

The real-node submatrix K_{real} (the $H \times H$ block of the augmented kernel after removing the two boundary columns) satisfies $\|K_{\text{real}}\|_\infty \le 1$ and $\rho(K_{\text{real}}) \le 1$.

This is a conditional robustness statement for the transport core: once the gates are fixed, the scans and conservative fusion do not amplify input perturbations. Any additional sensitivity comes from gate feedback, the fuse projection, or the nonlinear readout, which are treated separately below.

Theorem 5 (Hazard sensitivity and effective memory)**.** *For a past injection at $j \le k$, the following derivatives quantify how the coefficient changes when one hazard is perturbed:*

$$\frac{\partial A^p_{kj}}{\partial \lambda^p_j} = \frac{\rho^p_j}{1-\rho^p_j} A^p_{kj}, \qquad \frac{\partial A^p_{kj}}{\partial \lambda^p_t} = -A^p_{kj} \quad (j < t \le k), \tag{23}$$

with mirror identities for A^f. For an infinite window and constant retention $\rho_k \equiv \rho$,

$$\mathbb{P}(L = \ell) = (1-\rho)\rho^\ell, \qquad \mathbb{E}[L] = \frac{\rho}{1-\rho}, \qquad L_H(\rho) = \frac{1-\rho^H}{1-\rho}. \tag{24}$$

Here, $L \in \{0, 1, \ldots\}$ is the random lag before forgetting under a constant retention ρ and $L_H(\rho)$ is the corresponding finite-window survival sum.

The derivatives identify which hazards strengthen or suppress a path. For constant retention, $\rho/(1-\rho)$ is the expected additional memory and $L_H(\rho)$ its finite-window analogue. The signed displacement and the state-dependent gate correction are given in Appendix B. Proofs are given in Appendix C.

5 Experiments

We evaluate probabilistic forecasting performance on eight public datasets: Electricity, Exchange, Traffic, M4-Hourly, Wiki2000, Taxi-30m, Wind, and Weather. The experiments are designed to evaluate CoBiST's improvements across different generative interfaces, its performance relative to existing probabilistic forecasters, and the interpretability of its transport quantities. All ablation variants and external baselines are fairly tuned under the same validation protocol, well-trained under their prescribed optimization budgets, and evaluated using four independent random seeds. Details of data, tuning, training and evaluation are given in Appendix D. Code files for all experiments are available here, where the results can be reproduced easily by following the instructions inside.

Ablation study. CoBiST provides average reductions in PCRPS relative to the corresponding capacity-matched dense controls by 6.72%, 21.48%, and 20.78% for GAN, Flow, and Diffusion, respectively. We also compare CoBiST's geometric mean-based budget with a budget adjusted by a neural network (Neural-q), while other coupling methods, including alternative boundary values or means were not evaluated as preliminary tests showed only marginal differences. The two variants perform comparably, suggesting that geometric mean can serve as a simpler and more intuitive proxy with satisfactory accuracy. Detailed dataset-specific PCRPS, MCRPS, ES, and VS results are provided in Appendix E.1, where both mean and standard deviation are reported, allowing the magnitude and run-to-run variability of the improvements to be assessed.

External comparison. We compare the three CoBiST instances with six complete probabilistic forecasters described in Section 2. (Models that cannot estimate the joint predictive distribution, like point-forecasting models, despite being mentioned in Section 2, are not considered for comparison as we focus on distributional forecasting.) Official implementations are used if available. Table 3 reports the main PCRPS results, where CoBiST-Flow and CoBiST-Diffusion achieve the best 2 average ranks across datasets with satisfactory stability. Additional detailed dataset-specific MCRPS, ES, and VS results are provided in Appendix E.2. The consistently small cross-seed standard deviations relative to the corresponding means confirm the stability of the reported results.

Intervention-based validation of transport semantics. We run a separate frozen-gate diagnostic on the selected CoBiST configurations to evaluate the interpretability of the normalized route and source masses. At a fixed generator state, we measure the absolute change in the output of the shared nonlinear readout after removing a source contribution from the conservative memory and signed contrast. All other quantities, including the generator state, context, local coordinate, and gates, are held fixed, thereby isolating the realized transport from compensating gate feedback. Figure 2 compares the exact kernel quantities with these intervention responses. Overall, the intervention responses exhibit a clear alignment with the kernel-based quantities, supporting the interpretability of CoBiST's transport semantics. Detailed dataset-level statistics are reported in Appendix E.3.

Table 3: PCRPS, mean (standard deviation) and average ranks, best two shown in bold.

Model	Elec.	Exch.	Traffic	M4	Wiki	Taxi	Wind	Weather	Avg. Rank
DeepAR	**0.3854** **(0.0025)**	**0.1297** **(0.0046)**	**0.4087** **(0.0027)**	0.3266 (0.0010)	1.8561 (0.0044)	**0.5978** **(0.0030)**	0.6098 (0.0411)	0.9733 (0.0117)	3.00
CSDI	0.3996 (0.0111)	0.1366 (0.0097)	0.4701 (0.0052)	0.3275 (0.0144)	1.8544 (0.0273)	0.6242 (0.0111)	0.5763 (0.0117)	0.9291 (0.0307)	3.75
TimeGrad	0.7491 (0.0003)	0.2609 (0.0011)	0.7818 (0.0003)	0.6673 (0.0006)	1.8992 (0.0015)	0.7920 (0.0005)	0.7470 (0.0007)	0.9530 (0.0022)	7.38
TimeDiff	0.4453 (0.0142)	0.1974 (0.0026)	0.5254 (0.0081)	0.3301 (0.0056)	2.0285 (0.0703)	0.6721 (0.0052)	0.6484 (0.0326)	0.9604 (0.0375)	6.25
TSFlow	0.9202 (0.0168)	0.3203 (0.0031)	0.9693 (0.0319)	0.9072 (0.0078)	1.9886 (0.0155)	1.0262 (0.0154)	0.7920 (0.0063)	1.0456 (0.0130)	8.88
TACTiS-2	0.6583 (0.0934)	0.1494 (0.0108)	0.5608 (0.0102)	0.3979 (0.0259)	1.8556 (0.0080)	0.6813 (0.0140)	0.7163 (0.0930)	0.9929 (0.0162)	6.50
CoBiST-GAN	0.4339 (0.0115)	0.1414 (0.0126)	0.4760 (0.0060)	0.3510 (0.0102)	1.9400 (0.0474)	0.6878 (0.0178)	0.5587 (0.0368)	**0.9273** **(0.0202)**	5.00
CoBiST-Flow	**0.3904** **(0.0013)**	0.1358 (0.0024)	0.4497 (0.0048)	**0.2974** **(0.0034)**	**1.8431** **(0.0206)**	**0.6131** **(0.0045)**	**0.4305** **(0.0115)**	0.9325 (0.0107)	**2.25**
CoBiST-Diffusion	0.3923 (0.0013)	**0.1328** **(0.0032)**	**0.4453** **(0.0027)**	**0.2987** **(0.0066)**	**1.8323** **(0.0019)**	0.6215 (0.0019)	**0.4325** **(0.0116)**	**0.9175** **(0.0213)**	**2.00**

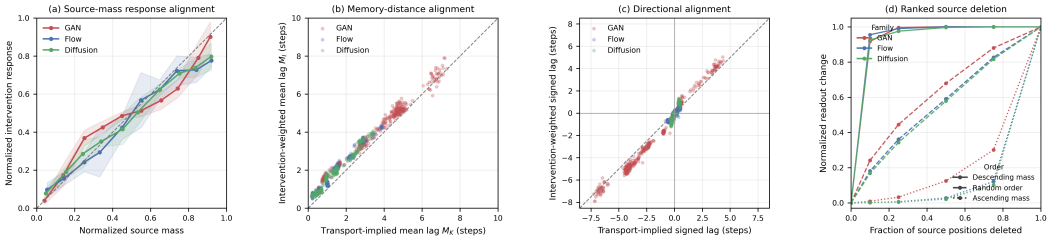

Figure 2: Frozen-gate validation of CoBiST transport semantics (seed 2027). (a) Normalized real-source mass versus normalized group-deletion response. Ribbons show 1.96 standard errors across datasets. (b) CoBiST-implied versus intervention-weighted mean source lag. The dashed diagonal denotes exact agreement. (c) CoBiST-implied versus intervention-weighted signed lag. Matching signs indicate agreement in direction. (d) Normalized readout change after deleting source positions. Colors identify GAN, Flow, and Diffusion, while line styles identify descending-mass, fixed-random, and ascending-mass ordering. All curves are normalized by the full-deletion response.

6 Conclusion and Discussion

We introduced CoBiST, a generator-agnostic cross-horizon transport operator for probabilistic time-series generative forecasters, which couples the directional transmission and memory mechanisms into one interpretable conserved route budget. The resulting global interaction is semiseparable and linear-time at the transport level, while route conservation, memory sensitivity, and conditional stability admit explicit analyses. The same operator improves capacity-matched dense controls across adversarial, flow-matching, and diffusion forecasters without changing their outer objectives or samplers and shows competitiveness in comparison to existing probabilistic time-series forecasters. CoBiST therefore provides a common structural interface for studying the internal interaction within a generated path with an interpretable account of the information propagation process.

Our evaluation covers basic GAN, flow-matching, and diffusion generators, but does not yet establish the applicability of CoBiST to autoregressive, online, irregularly sampled, or streaming forecasting. The row-conservation and semiseparable results characterize the transport construction, whereas the non-expansiveness guarantee applies only to the frozen-gate transport kernel. Although the transport interaction is linear in the horizon length, the complete block still includes width-dependent projections and residual layers. Finally, although the geometric-mean coupling can work as a simple and interpretable proxy for the learned Neural-q budget in our experiments, its uniqueness follows only from the stated hazard-semantics axioms and does not imply universal optimality.

AI USE STATEMENT

Generative AI tools assisted with implementation and testing, literature discovery, grammar checking and language polishing. Sections concerning the experimental protocol and result tables are partially drafted with the assistance of Generative AI. All AI-assisted code, experimental setups, and generated texts were rigorously reviewed and verified by the authors. Authors take full responsibility for the content and conclusions of this paper.

REPRODUCIBILITY STATEMENT

We are committed to fully reproducible research and provide the following resources and details:

Codebase and Implementation. The complete source code is publicly accessible at `https://anonymous.4open.science/r/CoBiST-0800`. Architectural specifications, hyperparameter selections, random seeds and training procedures are detailed in Appendix D.

Data Availability. All experiments are evaluated on benchmark public datasets. Data downloading and preprocessing pipelines can be executed using the `prepare_data.py` script included in the codebase.

Computing Environment and Reproduction Pipeline. Full hardware specifications, software dependencies (e.g., PyTorch and CUDA versions), and environment configuration files (`environment.yml`) are documented in the repository's `README.md`. The pipeline can be reproduced easily within ten commands.

REFERENCES

Arjun Ashok, Étienne Marcotte, Valentina Zantedeschi, Nicolas Chapados, and Alexandre Drouin. Tactis-2: Better, faster, simpler attentional copulas for multivariate time series. In *International conference on learning representations*, volume 2024, pp. 22897–22924, 2024.

Peng Chen, Yingying Zhang, Yunyao Cheng, Yang Shu, Yihang Wang, Qingsong Wen, Bin Yang, and Chenjuan Guo. Pathformer: Multi-scale transformers with adaptive pathways for time series forecasting. In *International conference on learning representations*, volume 2024, pp. 10527–10545, 2024.

Tri Dao and Albert Gu. Transformers are ssms: Generalized models and efficient algorithms through structured state space duality. *arXiv preprint arXiv:2405.21060*, 2024.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396*, 2021.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

Sukjun Hwang, Aakash Lahoti, Ratish Puduppully, Tri Dao, and Albert Gu. Hydra: Bidirectional state space models through generalized matrix mixers. *Advances in Neural Information Processing Systems*, 37:110876–110908, 2024.

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*, volume 2025, pp. 95435–95457, 2025.

Bryan Lim, Sercan Ö Arık, Nicolas Loeff, and Tomas Pfister. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International journal of forecasting*, 37(4): 1748–1764, 2021.

Xuezhe Ma, Chunting Zhou, Xiang Kong, Junxian He, Liangke Gui, Graham Neubig, Jonathan May, and Luke Zettlemoyer. Mega: Moving average equipped gated attention. *arXiv preprint arXiv:2209.10655*, 2022.

Bo Peng, Eric Alcaide, Quentin Anthony, Alon Albalak, Samuel Arcadinho, Stella Biderman, Huanqi Cao, Xin Cheng, Michael Chung, Leon Derczynski, et al. Rwkv: Reinventing rnns for the transformer era. In *Findings of the association for computational linguistics: EMNLP 2023*, pp. 14048–14077, 2023.

Michael Poli, Stefano Massaroli, Eric Nguyen, Daniel Y Fu, Tri Dao, Stephen Baccus, Yoshua Bengio, Stefano Ermon, and Christopher Ré. Hyena hierarchy: Towards larger convolutional language models. In *International Conference on Machine Learning*, pp. 28043–28078. PMLR, 2023.

Kashif Rasul, Calvin Seward, Ingmar Schuster, and Roland Vollgraf. Autoregressive denoising diffusion models for multivariate probabilistic time series forecasting. In *International conference on machine learning*, pp. 8857–8868. PMLR, 2021.

David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. Deepar: Probabilistic forecasting with autoregressive recurrent networks. *International journal of forecasting*, 36(3): 1181–1191, 2020.

Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681, 1997.

Lifeng Shen and James Kwok. Non-autoregressive conditional diffusion models for time series prediction. In *International Conference on Machine Learning*, pp. 31016–31029. PMLR, 2023.

Sainbayar Sukhbaatar, Edouard Grave, Piotr Bojanowski, and Armand Joulin. Adaptive attention span in transformers. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 331–335, 2019.

Yutao Sun, Li Dong, Shaohan Huang, Shuming Ma, Yuqing Xia, Jilong Xue, Jianyong Wang, and Furu Wei. Retentive network: A successor to transformer for large language models. *arXiv preprint arXiv:2307.08621*, 2023.

Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. Csdi: Conditional score-based diffusion models for probabilistic time series imputation. *Advances in neural information processing systems*, 34:24804–24816, 2021.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Songlin Yang, Bailin Wang, Yikang Shen, Rameswar Panda, and Yoon Kim. Gated linear attention transformers with hardware-efficient training. *arXiv preprint arXiv:2312.06635*, 2023.

A GENERATIVE INTERFACES

This appendix specifies the three outer generator interfaces. In each case, CoBiST is inserted between the interface-specific state embedding and the output head. The transport equations and gate semantics defined in the main text are shared without modification.

A.1 COBIST–GAN

Let $\varepsilon \in \mathbb{R}^{H\times m}$ be a base latent path, where m is the latent width and the entries of ε are sampled independently (or from a prescribed base time-series law). The decoder produces coefficients and reconstructs a forecast through a location map:

$$\delta = \mathrm{CoBiST}_\theta(\varepsilon, H_\theta(x)), \qquad \widehat{y} = A(x) + S\,\delta. \tag{1}$$

Here δ is the generated coefficient path, $A(x)$ is a history-dependent location or baseline, and S is a fixed or learned scale map that maps the coefficients back to observation units. With a discriminator D_ψ, a least-squares adversarial objective is

$$\begin{aligned}\mathcal{L}_D &= \frac{1}{2}\mathbb{E}_{y\sim p^\star}[D_\psi(y,x)-1]^2 + \frac{1}{2}\mathbb{E}_\varepsilon[D_\psi(G_\theta(\varepsilon,x),x)]^2,\\ \mathcal{L}_G &= \frac{1}{2}\mathbb{E}_\varepsilon[D_\psi(G_\theta(\varepsilon,x),x)-1]^2.\end{aligned} \tag{2}$$

CoBiST changes the latent interaction operator but not the adversarial loss. This realization coordinates all horizon positions in one non-autoregressive generator pass.

A.2 COBIST–FLOW

Let $y_1 \in \mathbb{R}^n$ be a standardized target coefficient, where $n = Hd_y$ after flattening and d_y is the number of target channels. Let $y_0 \sim \mathcal{N}(0, I_n)$ be a Gaussian base coefficient. For $t \sim \mathrm{Unif}[0,1]$, define the linear probability path and its target velocity by

$$u_t = (1-t)y_0 + ty_1, \qquad v_t^\star = y_1 - y_0. \tag{3}$$

Here u_t is the coefficient state seen by CoBiST at interpolation time t, while $v_t^\star$ is the instantaneous velocity that transports the Gaussian base toward the data coefficient. A conditional CoBiST vector field is trained with the flow-matching objective

$$\mathcal{L}_{\mathrm{FM}}(\theta) = \mathbb{E}\big\|v_\theta(u_t, t, H_\theta(x)) - v_t^\star\big\|_2^2. \tag{4}$$

At generation time, the model integrates

$$\dot{u}_t = v_\theta(u_t, t, H_\theta(x)), \qquad u_0 \sim \mathcal{N}(0, I_n), \tag{5}$$

and applies the inverse standardization transform to obtain the forecast. The CoBiST transport core is unchanged: only the presented state and the outer flow-matching loss differ. The Flow interface uses the standard flow-matching objective and ODE sampler, while CoBiST changes only the cross-horizon interaction operator.

A.3 COBIST–DIFFUSION

For a variance-preserving schedule (α_t, σ_t), where α_t and σ_t are respectively the signal and noise coefficients and satisfy $\alpha_t^2 + \sigma_t^2 = 1$, define the noise angle $\phi_t = \mathrm{atan2}(\sigma_t, \alpha_t)$. Thus $\alpha_t = \cos\phi_t$ and $\sigma_t = \sin\phi_t$,

$$u_t = \alpha_t y + \sigma_t \varepsilon, \qquad v_t^\star = \alpha_t \varepsilon - \sigma_t y. \tag{6}$$

Here y is the standardized target path and $\varepsilon \sim \mathcal{N}(0, I_n)$ is an independent Gaussian noise vector. $v_t^\star = \partial u_t/\partial\phi_t$ is the standard angular v-parameterization target. It is not identified with $\partial u_t/\partial t$ unless $\phi_t = t$. In general, $\partial u_t/\partial t = \dot{\phi}_t v_t^\star$. The CoBiST denoiser is trained with

$$\mathcal{L}_{\mathrm{diff}}(\theta) = \mathbb{E}\|v_\theta(u_t, t, H_\theta(x)) - v_t^\star\|_2^2. \tag{7}$$

For sampling, the schedule-specific denoiser quantities can be recovered from the predicted v through

$$\widehat{y}_t = \alpha_t u_t - \sigma_t v_\theta, \qquad \widehat{\varepsilon}_t = \sigma_t u_t + \alpha_t v_\theta. \tag{8}$$

The reverse sampler and noise schedule remain unchanged, using the schedule-specific conversion above. CoBiST supplies structured cross-horizon communication to the denoiser, while diffusion supplies the noise-removal objective and reverse-time sampler.

B Additional theoretical results

B.1 State-dependent gate qualification

Proposition 3 (State-dependent Jacobian decomposition). *Let $\mathcal{G}(Z, c) = (\alpha, \rho)$ be the C^1 gate map, where $Z = (z_1, \ldots, z_H)$ is the stacked $H \times w$ state and c is held fixed. With*

$$T(Z, c) = K(\mathcal{G}(Z, c))Z + K_\partial(\mathcal{G}(Z, c))\xi,$$

for every perturbation $\dot{Z}$,

$$D_Z T[\dot{Z}] = K(\mathcal{G}(Z, c))\dot{Z} + (D_Z K[\dot{Z}])Z + (D_Z K_\partial[\dot{Z}])\xi. \tag{1}$$

Here K contains real-node coefficients, K_∂ the two boundary columns, and $\xi = (\xi_p, \xi_f)$. The first term is the frozen-gate semiseparable transport, and the other terms are gate-response contributions.

Proposition 4 (Finite-horizon state-dependent stability bound). *For two bounded inputs Z, Z', define*

$$\begin{aligned} \delta &= \|Z - Z'\|_\infty, \qquad \delta_\rho = \max_k \big(|\Delta\rho_k^p| + |\Delta\rho_k^f|\big), \\ \delta_\alpha &= \max_k \|\Delta\alpha_k\|_1, \qquad \delta_q = \max_k |\Delta q_k|, \end{aligned} \tag{2}$$

where $\Delta\rho_k^d = \rho_k^d - \rho_k^{d\prime}$ and $\Delta\alpha_k = \alpha_k - \alpha_k'$. Let $q_\star = \max_k \max(q_k, q_k')$. Then

$$\|P - P'\|_\infty, \|F - F'\|_\infty \le \delta + 2BH\delta_\rho, \tag{3}$$

$$\|T - T'\|_\infty \le \delta + 2BH\delta_\rho + B(2\delta_q + q_\star\delta_\alpha), \tag{4}$$

$$\|D - D'\|_\infty \le q_\star(\delta + 2BH\delta_\rho) + B(2\delta_q + q_\star\delta_\alpha). \tag{5}$$

Thus gate feedback contributes explicit, horizon-dependent correction terms. The frozen-gate non-expansiveness theorem should not be extended to the unconstrained complete network.

B.2 Directional displacement

Corollary 2 (Expected directional displacement). *For constant directional retentions and an infinite window, counting the current injection as lag zero, let Δ denote the signed index displacement (future positions positive, past positions negative).*

$$\mathbb{E}|\Delta| = q\left[\alpha^p \frac{\rho^p}{1-\rho^p} + \alpha^f \frac{\rho^f}{1-\rho^f}\right], \qquad \mathbb{E}[\Delta] = q\left[\alpha^f \frac{\rho^f}{1-\rho^f} - \alpha^p \frac{\rho^p}{1-\rho^p}\right]. \tag{6}$$

The first quantity measures the expected absolute memory distance, while the second is a signed directional bias.

These two moments complement the effective-memory length in Theorem 5: the absolute displacement reports how far information travels, while the signed displacement reports which direction is preferred. Both can be computed from learned gates without perturbing or retraining the model.

C Proofs

C.1 Auxiliary facts and conventions

All inequalities in this appendix are pointwise in the conditioning context c. When gates are frozen, primes distinguish a second input sequence with the same gates. When gates are state dependent, primes also distinguish the corresponding gate values. We repeatedly use the elementary facts

$$0 \le \rho \le 1 \implies \|\rho v + (1-\rho)u\| \le \max\{\|u\|, \|v\|\}, \tag{7}$$

and, for a nonnegative matrix M,

$$\|M\|_\infty = \max_i \sum_j M_{ij}, \qquad \rho(M) \le \|M\|_\infty. \tag{8}$$

The first identity is the definition of the induced infinity norm and the second follows from the standard spectral-radius bound for any induced matrix norm. These two observations are the only norm facts needed for the stability results.

C.2 Proof of Theorem 1

Because $q > 0$, define $r(\lambda_p, \lambda_f) = -\log q(\lambda_p, \lambda_f)$. The normalization $q(0,0) = 1$ gives $r(0,0) = 0$, and the multiplicative composability assumption becomes the continuous additive equation

$$r(x+y) = r(x) + r(y), \qquad x, y \in \mathbb{R}_+^2. \tag{9}$$

To solve the additive equation above, first restrict r to each coordinate axis: $g_p(t) = r(t,0)$ and $g_f(t) = r(0,t)$. They are continuous additive functions on $\mathbb{R}_+$. For a positive integer n, $g_p(nt) = ng_p(t)$. And for a positive rational m/n, additivity gives $g_p((m/n)t) = (m/n)g_p(t)$. Continuity extends this identity from rational to real $t \geq 0$, so $g_p(t) = at$ and $g_f(t) = bt$, where $a = g_p(1)$, $b = g_f(1)$. Applying additivity to $(\lambda_p, \lambda_f) = (\lambda_p, 0) + (0, \lambda_f)$ therefore yields

$$r(\lambda_p, \lambda_f) = a\lambda_p + b\lambda_f. \tag{10}$$

Because $q \leq 1$, $r \geq 0$ on the quadrant and hence $a, b \geq 0$. Symmetry of q is symmetry of r, hence $a = b$. Diagonal calibration $q(\lambda, \lambda) = e^{-\lambda}$ implies $r(\lambda, \lambda) = \lambda$, so $2a = 1$. Thus $r = (\lambda_p + \lambda_f)/2$ and exponentiating proves $q = e^{-(\lambda_p + \lambda_f)/2}$. Since $\rho_d = e^{-\lambda_d}$, this is exactly $q = \sqrt{\rho_p \rho_f}$. The conclusion is unique within the explicit continuity, composability, symmetry, and calibration axioms. It is not a claim about arbitrary couplings that do not satisfy them.

C.3 Proof of Corollary 1

By definition,

$$\lambda_\Sigma = \frac{\lambda^p + \lambda^f}{2}, \qquad \lambda_\Delta = \frac{\lambda^p - \lambda^f}{2}.$$

Solving these two linear equations for the directional hazards gives $\lambda^p = \lambda_\Sigma + \lambda_\Delta$ and $\lambda^f = \lambda_\Sigma - \lambda_\Delta$. Nonnegativity of both hazards is therefore equivalent to $\lambda_\Sigma \geq |\lambda_\Delta|$. The identity $q = e^{-\lambda_\Sigma}$ follows from the geometric-mean definition, and exponentiating the two solved hazards gives the displayed expressions for ρ^p, ρ^f. Differentiating $q = e^{-(\lambda^p + \lambda^f)/2}$ gives $\partial q/\partial \lambda^p = \partial q/\partial \lambda^f = -q/2$. Finally, the arithmetic–geometric mean inequality for positive numbers gives $\min(\rho^p, \rho^f) \leq \sqrt{\rho^p \rho^f} \leq (\rho^p + \rho^f)/2$.

C.4 Proofs of Theorem 2 and Proposition 2

We prove the past identity by induction on k. For $k = 0$, $P_0 = \xi_p$, which agrees with the empty sum and the convention $\prod_{r=1}^{0} \rho_r^p = 1$. Assume the identity holds at $k-1$. Substituting it into the past recurrence gives

$$\begin{aligned} P_k &= \rho_k^p \left[\left(\prod_{r=1}^{k-1} \rho_r^p \right) \xi_p + \sum_{j=1}^{k-1} (1 - \rho_j^p) \left(\prod_{r=j+1}^{k-1} \rho_r^p \right) z_j \right] + (1 - \rho_k^p) z_k \\ &= \left(\prod_{r=1}^{k} \rho_r^p \right) \xi_p + \sum_{j=1}^{k} (1 - \rho_j^p) \left(\prod_{r=j+1}^{k} \rho_r^p \right) z_j. \end{aligned} \tag{11}$$

This proves the past coefficient expansion, including the causal support $j \leq k$. For the future branch, perform backward induction from $F_{H+1} = \xi_f$. The induction step in the future recurrence is

$$\begin{aligned} F_k &= \rho_k^f \left[\left(\prod_{r=k+1}^{H} \rho_r^f \right) \xi_f + \sum_{j=k+1}^{H} (1 - \rho_j^f) \left(\prod_{r=k+1}^{j-1} \rho_r^f \right) z_j \right] + (1 - \rho_k^f) z_k \\ &= \left(\prod_{r=k}^{H} \rho_r^f \right) \xi_f + \sum_{j=k}^{H} (1 - \rho_j^f) \left(\prod_{r=k}^{j-1} \rho_r^f \right) z_j. \end{aligned} \tag{12}$$

To verify the coefficient sums explicitly, define $S_k^p = \beta_k^p + \sum_{j=1}^{H} A_{kj}^p$. The recurrence gives $S_0^p = 1$ and $S_k^p = \rho_k^p S_{k-1}^p + (1 - \rho_k^p) = 1$. The same backward calculation gives $S_k^f = \beta_k^f + \sum_{j=1}^{H} A_{kj}^f = 1$. Thus both scans are affine averages of injections and boundary values.

Substituting the two scan expansions into $T_k = \pi_k^p P_k + \pi_k^0 z_k + \pi_k^f F_k$ collects the coefficient of each real node z_j as

$$K_{kj} = \pi_k^p A_{kj}^p + \pi_k^0 \mathbf{1}\{j = k\} + \pi_k^f A_{kj}^f, \tag{13}$$

and the two boundary coefficients are $K_{k0} = \pi_k^p \beta_k^p$ and $K_{k,H+1} = \pi_k^f \beta_k^f$. Consequently,

$$\begin{aligned} K_{k0} + \sum_{j=1}^{H} K_{kj} + K_{k,H+1} &= \pi_k^p \left(\beta_k^p + \sum_{j=1}^{H} A_{kj}^p \right) + \pi_k^0 + \pi_k^f \left(\beta_k^f + \sum_{j=1}^{H} A_{kj}^f \right) \\ &= \pi_k^p + \pi_k^0 + \pi_k^f = 1, \end{aligned} \tag{14}$$

which proves both the kernel representation and augmented row stochasticity. When $\xi_p = \xi_f = 0$, the boundary columns still carry probability mass but do not contribute numerically. Deleting them therefore makes the real block substochastic.

The past scan identity gives $\sum_{j<k} A_{kj}^p = (1-\beta_k^p) - (1-\rho_k^p) = \rho_k^p - \beta_k^p$, and the future identity gives the analogous $\sum_{j>k} A_{kj}^f = \rho_k^f - \beta_k^f$. The diagonal coefficient is $K_{kk} = \pi_k^0 + \pi_k^p(1-\rho_k^p) + \pi_k^f(1-\rho_k^f)$. Combining these terms with the two boundary coefficients and the row conservation result proves the stated source-level decomposition.

C.5 Proof of Theorem 3

For $j \le k$, the cumulative products satisfy

$$s_k^p = s_j^p \prod_{r=j+1}^{k} \rho_r^p \quad \Longrightarrow \quad \prod_{r=j+1}^{k} \rho_r^p = \frac{s_k^p}{s_j^p}.$$

Substitution into the definition of A_{kj}^p gives $A_{kj}^p = s_k^p(1-\rho_j^p)/s_j^p$ on the lower triangle and zero above it. This is a rank-one semiseparable factorization: after fixing k, the k-dependence is entirely contained in s_k^p, whereas the j-dependence is entirely contained in $(1-\rho_j^p)/s_j^p$. Similarly, for $k \le j$,

$$u_j^f = u_k^f \prod_{r=k}^{j-1} \rho_r^f \quad \Longrightarrow \quad A_{kj}^f = \frac{(1-\rho_j^f)u_j^f}{u_k^f}$$

on the upper triangle. Since all ρ's lie in $(0,1)$, the denominators are nonzero. Computing the scans by the two recurrences requires one scalar–vector multiply and one vector addition per location and direction. Thus the transport interaction itself costs $O(Hw)$ arithmetic and $O(Hw)$ working memory. The trainable gate, fuse, residual blocks, normalization and output layers are not part of this scan bound: for fixed d_{in}, w, L they add $O(Hw^2)$ arithmetic. Explicitly materializing A^p or A^f would instead require $O(H^2 w)$ storage (and applying the dense-prefix control used in the experiments has the $O(H^2 d_{\text{in}} w)$ input projection shown in Table 2).

C.6 Proofs of Theorems 4 and 5

With fixed gates, subtracting the two past recurrences gives

$$P_k - P_k' = \rho_k^p (P_{k-1} - P_{k-1}') + (1-\rho_k^p)(z_k - z_k'). \tag{15}$$

Taking norms and using the triangle inequality gives

$$\|P_k - P_k'\| \le \rho_k^p \|P_{k-1} - P_{k-1}'\| + (1-\rho_k^p)\delta. \tag{16}$$

The initial difference is zero because the boundary value is shared. If the right-hand state at $k-1$ is at most δ, then the displayed bound is at most $\rho_k^p \delta + (1-\rho_k^p)\delta = \delta$. Induction proves the first inequality for the past scan. Reversing the induction proves the future inequality. With fixed route weights,

$$\begin{aligned} \|T_k - T_k'\| &\le \pi_k^p \|P_k - P_k'\| + \pi_k^0 \|z_k - z_k'\| + \pi_k^f \|F_k - F_k'\| \\ &\le (\pi_k^p + \pi_k^0 + \pi_k^f)\delta = \delta. \end{aligned} \tag{17}$$

For the signed contrast, the two coefficients enter with opposite signs but the norm removes those signs:

$$\|D_k - D'_k\| \le \pi_k^f \|F_k - F'_k\| + \pi_k^p \|P_k - P'_k\| = q_k \delta. \tag{18}$$

Every entry of K_{real} is nonnegative. Its row sum is the augmented row sum minus the nonnegative boundary masses, hence is at most one. The induced-norm bound then gives $\|K_{\mathrm{real}}\|_\infty \le 1$ and $\rho(K_{\mathrm{real}}) \le 1$.

For the hazard identities, write $\rho_j^p = e^{-\lambda_j^p}$ in the kernel definition. For $j \le k$,

$$A_{kj}^p = (1 - e^{-\lambda_j^p}) \exp\left(- \sum_{r=j+1}^{k} \lambda_r^p \right). \tag{19}$$

The injection hazard appears only in the first factor, so

$$\frac{\partial A_{kj}^p}{\partial \lambda_j^p} = e^{-\lambda_j^p} \exp\left(- \sum_{r=j+1}^{k} \lambda_r^p \right) = \frac{\rho_j^p}{1 - \rho_j^p} A_{kj}^p. \tag{20}$$

For a downstream hazard $t \in \{j+1, \dots, k\}$, differentiating the exponential factor gives

$$\frac{\partial A_{kj}^p}{\partial \lambda_t^p} = -A_{kj}^p. \tag{21}$$

The future identities follow by exchanging $j \le k$ with $j \ge k$ and reversing the index order. If all retentions equal ρ, a contribution that is ℓ steps old survives ℓ times and is then forgotten on the next step, giving $\mathbb{P}(L = \ell) = (1 - \rho)\rho^\ell$. The geometric series identities

$$\sum_{\ell \ge 0} \ell (1 - \rho) \rho^\ell = \frac{\rho}{1 - \rho}, \qquad \sum_{\ell=0}^{H-1} \rho^\ell = \frac{1 - \rho^H}{1 - \rho} \tag{22}$$

prove the remaining memory claims.

C.7 Proof of Corollary 2

Condition on the recurrent-scan event, which has probability q. Given that event, the scan direction is causal with probability α^p and anti-causal with probability α^f. In the past-directed branch, the lag magnitude has mean $\rho^p/(1 - \rho^p)$ by the geometric-series calculation above. The future-directed branch has the analogous mean $\rho^f/(1 - \rho^f)$. The direct-local branch contributes zero displacement, and the current-state injection within either scan also has lag zero. Therefore the law of total expectation gives

$$\mathbb{E}|\Delta| = q \left(\alpha^p \frac{\rho^p}{1 - \rho^p} + \alpha^f \frac{\rho^f}{1 - \rho^f} \right).$$

Assigning a negative sign to a past displacement and a positive sign to a future displacement gives

$$\mathbb{E}[\Delta] = q \left(\alpha^f \frac{\rho^f}{1 - \rho^f} - \alpha^p \frac{\rho^p}{1 - \rho^p} \right),$$

which proves the two displayed displacement formulas.

C.8 Proofs of Propositions 3 and 4

The representation $T(Z, c) = K(\mathcal{G}(Z, c))Z + K_\partial(\mathcal{G}(Z, c))\xi$ is a composition involving a matrix-valued map and a vector-valued map. For an infinitesimal perturbation $\dot{Z}$, the product rule gives

$$D_Z T[\dot{Z}] = K(\mathcal{G}(Z, c))\dot{Z} + \big(D_Z K[\dot{Z}]\big)Z + \big(D_Z K_\partial[\dot{Z}]\big)\xi. \tag{23}$$

Since this identity holds for every direction $\dot{Z}$, it is the Jacobian decomposition. The first term freezes the gate and is semiseparable by Theorem 3, while the remaining terms are gate responses.

We prove the finite-difference bound. For different retentions, subtracting the past recurrences and adding and subtracting $\rho_k^p P'_{k-1}$ gives

$$\begin{aligned} P_k - P'_k = \rho_k^p (P_{k-1} - P'_{k-1}) + (1 - \rho_k^p)(z_k - z'_k) \\ + (\rho_k^p - \rho_k^{p\prime})(P'_{k-1} - z'_k). \end{aligned} \tag{24}$$

Each scan state remains in the ball of radius B by the convex-combination inequality, starting from a boundary state. Consequently the final term in the displayed recurrence has norm at most $2B|\Delta\rho_k^p|$. If $e_k^p = \max_{i \le k} \|P_i - P'_i\|$, then

$$e_k^p \le \max\{e_{k-1}^p, \delta\} + 2B|\Delta\rho_k^p| \le \delta + 2Bk\,\delta_\rho, \tag{25}$$

where the last inequality follows by induction from $e_0^p = 0$. The backward recurrence gives the same bound for F, proving the stated scan bound. Moreover, writing the two recurrent-scan coordinates as $q_k\alpha_k - q'_k\alpha'_k$ and the local coordinate as $-(q_k - q'_k)$, the triangle inequality gives

$$\|\pi_k - \pi'_k\|_1 \le 2|\Delta q_k| + q_\star \|\Delta\alpha_k\|_1. \tag{26}$$

Set $\bar{\delta} = \delta + 2BH\delta_\rho$. Decompose the difference of two transport summaries into a state part and a weight part:

$$\begin{aligned} T_k - T'_k = \pi_k^p (P_k - P'_k) + \pi_k^0 (z_k - z'_k) + \pi_k^f (F_k - F'_k) \\ + (\pi_k^p - \pi_k^{p\prime}) P'_k + (\pi_k^0 - \pi_k^{0\prime}) z'_k + (\pi_k^f - \pi_k^{f\prime}) F'_k. \end{aligned} \tag{27}$$

The first line is bounded by $\bar{\delta}$, and the second by $B\|\pi_k - \pi'_k\|_1$. Taking maxima and using the displayed weight bound proves the transport bound. Replacing the plus sign in the future term by the minus sign in D_k gives

$$\|D - D'\|_\infty \le q_\star \bar{\delta} + B\big(2\delta_q + q_\star \delta_\alpha\big), \tag{28}$$

which is the stated contrast bound.

D Reproducibility details

Benchmark and data protocol. We evaluate probabilistic forecasts on eight public datasets: Electricity, Exchange, Traffic, M4-Hourly, Wiki2000, Taxi-30m, Wind, and Weather. All methods use the same temporal preprocessing, context length, data splits, forecast origins, and evaluation samples. For each forecast origin, the most recent 96 observations are used as context. Each window is normalized using the median of its context and a robust mean-absolute-deviation scale, with the same normalization applied to the target horizon.

The split uses seed 26072026 and is independent of model randomness. For each training series, rolling origins are generated with stride equal to one quarter of the forecast horizon. The last 10% of the origins form the validation portion, and training targets are required not to overlap validation targets. The training set contains at most 60,000 windows. The validation set is capped at 10,000 windows, of which the first 512 are used for model selection. The test set uses the terminal forecast origin of every eligible test series, up to 10,000 windows. The effective numbers of windows in the campaign are shown below.

Table 4: Temporal windows used in the experiments.

Dataset	Horizon	Training	Validation evaluated	Test
Electricity	24	60,000	512	2,590
Exchange	30	6,088	512	40
Traffic	24	60,000	512	6,741
M4-Hourly	48	21,046	512	414
Wiki2000	30	60,000	512	10,000
Taxi-30m	24	60,000	512	10,000
Wind	60	60,000	512	339
Weather	30	60,000	512	3,010

Random seeds and model selection. For each model–dataset pair, we test a predeclared list of 30 configurations with tuning seed 2025. Candidates are ranked by validation PCRPS, and the five best candidates are repeated with seed 2026. The final configuration is chosen from these five candidates by minimizing the mean PCRPS over the two seeds, subject to its mean ES being no more than 5% larger than the best mean ES among the repeated candidates. Test targets are never used for configuration selection. After the configuration is fixed, the model is retrained and evaluated with formal test seeds $2027, 2028, 2029$, and 2030. The reported standard deviations are sample standard deviations over these four formal seeds. Python, NumPy, and PyTorch/CUDA random generators are initialized from the corresponding model seed.

CoBiST and ablation configurations. Unless noted otherwise, CoBiST-GAN, CoBiST-Flow, CoBiST-Diffusion, and their Dense controls use fixed 30-point candidate lists spanning

$$\text{width} \in \{64, 96, 128, 160, 192\}, \qquad \text{layers} \in \{2, 3, 4\},$$
$$\text{updates} \in \{800, 1200, 1600\}, \qquad \text{learning rate} \in \{10^{-4}, 2 \times 10^{-4}\}.$$

The listed values define the ranges of the deterministic candidate lists. They are not searched as a complete Cartesian product. GAN uses one direct sampling step, Flow uses 8, 16, or 24 integration steps, and Diffusion uses 16, 32, or 48 sampling steps.

The Dense control removes the CoBiST core while retaining the corresponding optimizer, training, and sampling settings. Its hidden width is selected from the same candidate grid to minimize the trainable-parameter difference from the corresponding CoBiST configuration. The realized parameter count, target count, effective width, and relative capacity difference are recorded for every run. The capacity difference between CoBiST and its Dense control is at most 10%. The Neural-q ablation uses a one-hidden-layer MLP (SiLU, hidden width $\max(16, \min(64, W/2))$) that receives local/context features and the two retention probabilities, and predicts a zero-initialized residual around the geometric mean in logit space with scale 0.15 while all original model hyperparameters and training protocols are kept unchanged.

Special case: Wind-GAN stability grid. We found this model–dataset pair to be particularly sensitive and unstable. We therefore use a predeclared 30-point grid with widths $\{64, 96, 128, 160, 192\}$, depths $\{2, 3, 4\}$, and updates $\{1200, 1800, 2400\}$. Learning rates are $\{5 \times 10^{-5}, 10^{-4}, 2 \times 10^{-4}, 3 \times 10^{-4}\}$, while moment, auxiliary-energy, and earliest-checkpoint choices are respectively $\{0.01, 0.02, 0.05\}$, $\{0.1, 0.25, 0.5\}$, and $\{0.15, 0.25, 0.4\}$. It uses one direct sampling step and the same repetition rule.

External baselines. All external baselines are adapted to the common data, tuning, and evaluation protocol. DeepAR uses a 30-point list with widths $\{64, 96, 128, 160, 192\}$, depths $\{2, 3, 4\}$, and updates $\{800, 1200, 1600\}$, with learning rates $\{2.5 \times 10^{-4}, 5 \times 10^{-4}, 10^{-3}\}$. CSDI, TimeDiff, and TSFlow use 30 candidates with updates $800, 840, \ldots, 1960$, learning rates $\{5{\times}10^{-4}, 10^{-3}, 2{\times}10^{-3}\}$, and paired diffusion/inference steps $\{16, 32, 48\}$. TimeGrad uses the same schedule with learning rates $\{10^{-4}, 2 \times 10^{-4}, 4 \times 10^{-4}\}$.

TACTiS-2 uses 30 configurations from eight architectural templates. Flow and copula dimensions are selected from $\{16, 32, 38, 40, 42, 48, 60\}$. Components use one or two layers and heads where applicable. Resolution is 20, batch size 64, and weight decay 10^{-6}. Base phase-1/2 updates range from 80–160 and 120–260. Scaling factors ranging from 1.0 to 1.3 generate candidates, and learning rates use multipliers $0.5, 1, 2$ over the predefined values, yielding $\{2.5 \times 10^{-4}, 5 \times 10^{-4}, 10^{-3}, 2 \times 10^{-3}\}$. Counts are optimization updates, not horizons.

Frozen-gate interpretability protocol. The interpretability analysis is a separate diagnostic campaign that reuses the configuration selected for each of the 8×3 CoBiST dataset–generator cells and uses only formal seed 2027. This is not included in the four-seed uncertainty estimates of the forecasting tables. We use up to the first 64 windows of the deterministic test split (all 40 for Exchange) and four matched generator samples per window. GAN has one transport state. For Flow and Diffusion, we record four uniformly spaced pre-update integration or denoising states from the initial to the final update. The resulting statistics are first averaged over the four matched samples, while the selected generation states and forecast positions remain available to the estimators below.

At a recorded state, let o_k denote the scalar output of the complete position-wise generator head after transport, fusion, residual blocks, and normalization. We evaluate the gate once and keep $(\alpha, \rho, \boldsymbol{\pi})$, the generator state, the context, and the direct local coordinate fixed. For a real source position j, we then remove only its contribution $K_{kj} z_j$ from the conservative memory coordinate and the corresponding coefficient from the signed-contrast coordinate, and pass the modified coordinates through the same nonlinear head. The empirical position-level influence is

$$I_{kj} = \left| o_k - o_k^{(-j)} \right|.$$

For the grouped source analysis, all $j < k$, $j = k$, or $j > k$ contributions are removed jointly, giving (I_k^-, I_k^0, I_k^+). Boundary mass is excluded before comparing the three real-source proportions:

$$p_k^d = \frac{\mu_k^d}{\mu_k^- + \mu_k^0 + \mu_k^+}, \qquad r_k^d = \frac{I_k^d}{I_k^- + I_k^0 + I_k^+}, \qquad d \in \{-, 0, +\}.$$

Mass–response alignment is summarized by the pooled Spearman correlation, mean Jensen–Shannon divergence between p_k and r_k, and agreement of their largest coordinates. Cells with a numerically zero total response are excluded from these three calculations.

To validate source-position ranking, we delete the largest 10%, 25%, 50%, 75%, and 100% of the K_{kj} values and compare the resulting readout change with a fixed seed-2027 random ordering and the reverse-mass ordering. Each curve is normalized by its full-deletion response, and deletion gain is the area between the CoBiST and random curves. The model- and intervention-based memory distances and signed displacements are

$$M_{K,k} = \frac{\sum_j |j-k| K_{kj}}{\sum_j K_{kj}}, \qquad M_{I,k} = \frac{\sum_j |j-k| I_{kj}}{\sum_j I_{kj}},$$

$$D_{K,k} = \frac{\sum_j (j-k) K_{kj}}{\sum_j K_{kj}}, \qquad D_{I,k} = \frac{\sum_j (j-k) I_{kj}}{\sum_j I_{kj}}.$$

The Direction Spearman statistic is the within-cell Spearman correlation between the sequences D_K and D_I, while Sign agreement is the fraction of nonzero pairs with matching signs. These finite-window quantities condition on real-node mass, and the two virtual boundary columns are evaluated separately. Since their implemented values are zero, deletion would have no numerical effect. We instead inject the fixed unit vector $v = \mathbf{1}/\sqrt{w}$ into both virtual boundaries and compare the absolute readout response with $\mu_k^\partial = \pi_k^p \beta_k^p + \pi_k^f \beta_k^f$. This test validates virtual-sink sensitivity rather than the usefulness of observed information.

All generator-level entries in Table 13 are macro averages of the eight dataset-level statistics. Their intervals are the 2.5th and 97.5th percentiles from 2,000 bootstrap resamples of the eight datasets. They therefore quantify dataset heterogeneity under seed 2027, not initialization or multi-seed uncertainty. The shaded bands in Figure 2(a) are 1.96 standard errors over the eight datasets.

Evaluation metrics. Every formal evaluation uses 128 forecast samples per origin. Let $\widehat{\boldsymbol{y}}^{(s)}$ denote the s-th forecast sample and $\boldsymbol{y}$ the observed target.

MCRPS is the average of the fair univariate CRPS over forecast dimensions:

$$\mathrm{CRPS} = \frac{1}{S} \sum_{s=1}^{S} |\widehat{y}^{(s)} - y| - \frac{1}{2S(S-1)} \sum_{s \neq t} |\widehat{y}^{(s)} - \widehat{y}^{(t)}|, \qquad S = 128.$$

PCRPS is computed by projecting each multivariate forecast onto 128 fixed random unit directions. The directions are generated once with seed 74021. The fair univariate CRPS is averaged over directions and evaluation origins.

ES is computed using the multivariate Euclidean distance:

$$\mathrm{ES} = \frac{1}{S} \sum_{s=1}^{S} \|\widehat{\boldsymbol{y}}^{(s)} - \boldsymbol{y}\|_2 - \frac{1}{2S(S-1)} \sum_{s \neq t} \|\widehat{\boldsymbol{y}}^{(s)} - \widehat{\boldsymbol{y}}^{(t)}\|_2.$$

VS compares pairwise increments of a sampled path:

$$\mathrm{VS}_p = \sum_{i,j} w_{ij} \left(|y_i - y_j|^p - \frac{1}{S} \sum_{s=1}^{S} |\widehat{y}_i^{(s)} - \widehat{y}_j^{(s)}|^p \right)^2 .$$

The same fixed exponent p and nonnegative pair weights w_{ij} are used for every model and dataset. VS supplements PCRPS, MCRPS, and ES with a dependence-sensitive diagnostic. Lower VS values are better. The released evaluation code contains the fixed implementation choices. For every metric, lower values indicate better probabilistic forecasts. Tables report the mean and sample standard deviation across the four formal test seeds. Ablation improvements are computed as

$$100 \times \frac{\mathrm{Metric}_{\mathrm{control}} - \mathrm{Metric}_{\mathrm{CoBiST}}}{\mathrm{Metric}_{\mathrm{control}}}.$$

Fair comparison. All models use the same normalized windows, forecast horizons, validation budget, candidate budget, selection rule, number of forecast samples, and formal test seeds. Architecture-specific hyperparameters stay within the predeclared ranges. Model selection is separate from final evaluation, and each Dense comparison changes only the designated structural component while remaining within the capacity constraint.

The complete implementation is public at `https://anonymous.4open.science/r/CoBiST-0800`. All experimental results can be exactly reproduced under the default settings.

E Additional experimental results

E.1 Ablation results

Table 5: PCRPS ablation improvement rates (%), mean (standard deviation).

Model vs Dense Control	Elec.	Exch.	Traffic	M4	Wiki	Taxi	Wind	Weather
CoBiST-GAN vs Dense	3.41 (4.03)	4.62 (11.32)	4.45 (3.82)	1.31 (5.25)	7.53 (6.51)	13.14 (3.53)	13.91 (3.60)	5.42 (5.10)
CoBiST-Flow vs Dense	17.05 (2.95)	23.52 (1.22)	7.42 (2.05)	15.18 (1.18)	15.31 (1.01)	4.59 (1.04)	63.07 (0.99)	25.70 (0.91)
CoBiST-Diffusion vs Dense	15.92 (0.30)	21.56 (2.36)	10.24 (0.41)	16.70 (1.69)	13.40 (0.22)	4.17 (0.45)	60.50 (1.05)	23.78 (1.97)
CoBiST-GAN vs Neural-q	-0.63 (2.14)	-2.55 (8.80)	-0.67 (1.03)	3.45 (4.41)	2.50 (3.70)	1.80 (4.55)	0.70 (4.12)	2.56 (6.71)
CoBiST-Flow vs Neural-q	-0.05 (0.13)	0.30 (3.20)	0.05 (0.20)	0.22 (0.34)	-0.39 (1.60)	-0.14 (0.09)	-0.85 (0.55)	0.88 (2.32)
CoBiST-Diffusion vs Neural-q	-0.11 (0.16)	-0.02 (0.25)	0.77 (0.13)	1.20 (2.68)	0.37 (0.27)	0.01 (0.17)	-3.38 (2.80)	1.47 (3.48)

Table 6: MCRPS ablation improvement rates (%), mean (standard deviation).

Model vs Dense Control	Elec.	Exch.	Traffic	M4	Wiki	Taxi	Wind	Weather
CoBiST-GAN vs Dense	3.06 (4.47)	3.70 (12.96)	3.80 (4.80)	-0.89 (4.64)	12.51 (11.48)	13.72 (4.27)	11.18 (3.75)	5.91 (3.89)
CoBiST-Flow vs Dense	17.82 (2.74)	16.85 (1.56)	13.54 (2.80)	15.43 (1.32)	26.39 (1.83)	7.36 (1.04)	63.21 (0.95)	33.31 (0.92)
CoBiST-Diffusion vs Dense	15.72 (0.31)	15.08 (2.61)	15.81 (0.60)	16.75 (1.60)	23.87 (0.36)	6.83 (0.42)	60.70 (0.97)	31.62 (1.50)
CoBiST-GAN vs Neural-q	-0.96 (2.25)	-3.79 (11.03)	-0.57 (0.96)	2.01 (2.37)	4.61 (6.71)	1.98 (5.24)	0.85 (4.98)	2.71 (6.61)
CoBiST-Flow vs Neural-q	-0.09 (0.13)	0.30 (3.51)	-0.05 (0.22)	0.18 (0.35)	-0.76 (3.65)	-0.16 (0.10)	-0.93 (0.24)	1.50 (3.54)
CoBiST-Diffusion vs Neural-q	-0.17 (0.16)	-0.03 (0.23)	1.08 (0.36)	1.23 (2.75)	0.60 (0.80)	0.01 (0.20)	-4.00 (3.21)	1.91 (2.80)

Table 7: ES ablation improvement rates (%), mean (standard deviation).

Model vs Dense Control	Elec.	Exch.	Traffic	M4	Wiki	Taxi	Wind	Weather
CoBiST-GAN vs Dense	4.13 (4.29)	4.98 (10.88)	4.12 (3.56)	0.76 (4.32)	7.39 (6.59)	13.51 (3.42)	13.02 (3.27)	5.40 (5.10)
CoBiST-Flow vs Dense	17.16 (2.61)	23.01 (1.16)	6.33 (2.26)	14.78 (1.10)	15.30 (1.02)	4.58 (1.04)	63.67 (0.93)	25.45 (0.88)
CoBiST-Diffusion vs Dense	15.57 (0.31)	21.09 (2.36)	9.16 (0.42)	16.81 (1.52)	13.38 (0.20)	4.34 (0.41)	61.16 (0.92)	23.44 (1.92)
CoBiST-GAN vs Neural-q	-0.63 (1.95)	-2.64 (8.50)	-0.25 (0.94)	2.83 (4.34)	2.40 (3.72)	1.75 (4.74)	0.47 (4.18)	2.52 (6.52)
CoBiST-Flow vs Neural-q	-0.06 (0.13)	0.32 (3.08)	0.06 (0.20)	0.20 (0.30)	-0.38 (1.64)	-0.14 (0.08)	-0.71 (0.50)	0.92 (2.36)
CoBiST-Diffusion vs Neural-q	-0.17 (0.19)	-0.02 (0.23)	0.81 (0.12)	1.13 (2.63)	0.36 (0.29)	0.01 (0.18)	-3.05 (2.45)	1.39 (3.36)

Table 8: VS ablation improvement rates (%), mean (standard deviation).

Model vs Dense Control	Elec.	Exch.	Traffic	M4	Wiki	Taxi	Wind	Weather
CoBiST-GAN vs Dense	11.04 (6.70)	26.85 (10.43)	6.49 (2.20)	2.73 (3.28)	8.43 (4.53)	20.50 (2.57)	-44.92 (6.91)	8.41 (6.80)
CoBiST-Flow vs Dense	45.27 (6.37)	89.56 (0.40)	38.64 (2.95)	41.07 (0.31)	58.78 (0.65)	29.33 (0.77)	95.50 (0.08)	74.44 (0.54)
CoBiST-Diffusion vs Dense	47.86 (0.38)	89.13 (0.12)	47.91 (0.51)	39.41 (0.67)	55.95 (0.11)	26.89 (0.11)	96.30 (0.17)	72.47 (0.46)
CoBiST-GAN vs Neural-q	-0.18 (4.39)	1.58 (8.11)	-0.49 (3.09)	0.21 (5.85)	-1.13 (1.89)	1.99 (4.82)	-4.91 (7.53)	2.59 (13.93)
CoBiST-Flow vs Neural-q	0.10 (0.12)	-0.27 (3.41)	-0.11 (0.29)	0.03 (0.15)	-0.23 (1.76)	-0.12 (0.08)	1.47 (3.18)	-0.01 (2.43)
CoBiST-Diffusion vs Neural-q	0.01 (0.98)	-0.06 (0.52)	0.68 (0.75)	0.28 (1.19)	0.36 (0.21)	0.03 (0.12)	2.73 (9.65)	-0.94 (1.54)

E.2 External comparison results

Table 9: Average performance ranks across eight datasets (1 is best, best two shown in bold).

Metric	DeepAR	CSDI	TimeGrad	TimeDiff	TSFlow	TACTiS-2	CoBiST-GAN	CoBiST-Flow	CoBiST-Diffusion
PCRPS	3.00	3.75	7.38	6.25	8.88	6.50	5.00	**2.25**	**2.00**
MCRPS	2.75	3.00	7.50	6.38	8.75	6.12	5.62	**2.50**	**2.38**
ES	2.88	3.62	7.38	6.25	8.88	6.62	5.12	**2.25**	**2.00**
VS	**2.25**	3.50	7.38	4.75	7.75	7.38	5.50	3.88	**2.62**

Table 10: MCRPS, mean (standard deviation), best two shown in bold.

Model	Elec.	Exch.	Traffic	M4	Wiki	Taxi	Wind	Weather
DeepAR	**0.3733 (0.0019)**	**0.1440 (0.0056)**	**0.3327 (0.0051)**	0.3232 (0.0012)	1.1693 (0.0050)	**0.5717 (0.0025)**	0.6196 (0.0276)	0.8015 (0.0152)
CSDI	**0.3922 (0.0103)**	0.1518 (0.0115)	**0.3975 (0.0028)**	0.3151 (0.0140)	1.1622 (0.0383)	0.6032 (0.0137)	0.5964 (0.0130)	**0.7528 (0.0160)**
TimeGrad	0.7959 (0.0005)	0.2967 (0.0017)	0.7654 (0.0004)	0.7149 (0.0014)	1.2113 (0.0016)	0.8178 (0.0005)	0.7897 (0.0023)	0.8594 (0.0036)
TimeDiff	0.4432 (0.0164)	0.2269 (0.0030)	0.4419 (0.0090)	0.3330 (0.0072)	1.3426 (0.0822)	0.6600 (0.0055)	0.6792 (0.0388)	0.9356 (0.0237)
TSFlow	0.9999 (0.0187)	0.3815 (0.0038)	0.9732 (0.0379)	1.0213 (0.0075)	1.2835 (0.0172)	1.1022 (0.0036)	0.8441 (0.0067)	0.9226 (0.0132)
TACTiS-2	0.6966 (0.1205)	0.1605 (0.0125)	0.4545 (0.0121)	0.3960 (0.0325)	**1.1522 (0.0097)**	0.6658 (0.0157)	0.6953 (0.0122)	0.8928 (0.0345)
CoBiST-GAN	0.4475 (0.0120)	0.1589 (0.0165)	0.4474 (0.0044)	0.3554 (0.0089)	1.2731 (0.0538)	0.6824 (0.0211)	0.5559 (0.0390)	0.8129 (0.0075)
CoBiST-Flow	0.3980 (0.0014)	0.1506 (0.0032)	0.4056 (0.0068)	**0.2956 (0.0039)**	1.1571 (0.0282)	**0.5936 (0.0047)**	**0.4292 (0.0112)**	0.7955 (0.0099)
CoBiST-Diffusion	0.4021 (0.0014)	**0.1471 (0.0033)**	0.4021 (0.0040)	**0.2967 (0.0063)**	**1.1420 (0.0024)**	0.6026 (0.0024)	**0.4301 (0.0110)**	**0.7782 (0.0151)**

Table 11: ES, mean (standard deviation), best two shown in bold.

Model	Elec.	Exch.	Traffic	M4	Wiki	Taxi	Wind	Weather
DeepAR	**2.3806** **(0.0140)**	**0.9003** **(0.0326)**	**2.5500** **(0.0190)**	2.9332 (0.0111)	12.6187 (0.0264)	**3.6745** **(0.0180)**	5.6236 (0.4032)	6.7097 (0.0846)
CSDI	2.4737 (0.0689)	0.9519 (0.0737)	2.9512 (0.0265)	2.8688 (0.1304)	12.6231 (0.2037)	3.8442 (0.0779)	5.2853 (0.0900)	**6.3321** **(0.2048)**
TimeGrad	4.8343 (0.0022)	1.7896 (0.0084)	4.8802 (0.0020)	6.1333 (0.0109)	12.9304 (0.0112)	4.9854 (0.0024)	6.7687 (0.0047)	6.5268 (0.0143)
TimeDiff	2.7494 (0.0835)	1.3739 (0.0168)	3.2620 (0.0379)	2.9646 (0.0497)	13.8182 (0.4753)	4.1331 (0.0298)	6.0184 (0.2887)	6.5777 (0.2535)
TSFlow	6.0735 (0.1222)	2.1911 (0.0206)	6.0571 (0.2018)	8.3762 (0.0689)	13.5340 (0.1096)	6.4827 (0.0781)	7.1962 (0.0536)	7.1400 (0.0906)
TACTiS-2	4.2502 (0.6633)	1.0377 (0.0759)	3.5080 (0.0618)	3.5591 (0.2869)	12.6338 (0.0561)	4.2102 (0.0863)	6.5153 (0.8989)	6.7734 (0.1068)
CoBiST-GAN	2.7229 (0.0694)	0.9810 (0.0861)	3.0150 (0.0309)	3.1346 (0.0732)	13.1695 (0.3255)	4.2539 (0.1138)	5.2255 (0.3238)	6.3566 (0.1288)
CoBiST-Flow	**2.4425** **(0.0078)**	0.9414 (0.0166)	2.8546 (0.0355)	**2.6443** **(0.0288)**	**12.5031** **(0.1481)**	**3.7765** **(0.0279)**	**4.0513** **(0.1032)**	6.3781 (0.0703)
CoBiST-Diffusion	2.4601 (0.0067)	**0.9208** **(0.0210)**	**2.8249** **(0.0161)**	**2.6453** **(0.0530)**	**12.4364** **(0.0074)**	3.8326 (0.0115)	**4.0693** **(0.0983)**	**6.2860** **(0.1402)**

Table 12: VS, mean (standard deviation), best two shown in bold.

Model	Elec.	Exch.	Traffic	M4	Wiki	Taxi	Wind	Weather
DeepAR	**0.1229** **(0.0014)**	**0.0182** **(0.0004)**	**0.1192** **(0.0023)**	0.0867 (0.0009)	**0.7662** **(0.0018)**	**0.2313** **(0.0006)**	0.2079 (0.1268)	**0.4150** **(0.0014)**
CSDI	0.1317 (0.0043)	0.0203 (0.0010)	0.1470 (0.0026)	0.0832 (0.0065)	0.7896 (0.0014)	**0.2396** **(0.0035)**	**0.0974** **(0.0074)**	0.4455 (0.0445)
TimeGrad	0.3336 (0.0023)	0.1114 (0.0013)	0.3001 (0.0006)	0.2720 (0.0015)	0.8385 (0.0022)	0.3203 (0.0008)	0.1884 (0.0081)	0.4443 (0.0053)
TimeDiff	**0.1280** **(0.0025)**	0.0279 (0.0011)	0.1416 (0.0020)	**0.0767** **(0.0018)**	0.9076 (0.0452)	0.2689 (0.0039)	**0.1102** **(0.0039)**	0.5294 (0.0440)
TSFlow	0.5143 (0.0246)	0.0322 (0.0013)	0.5671 (0.0445)	0.6176 (0.0186)	0.8643 (0.0186)	0.7210 (0.0138)	0.1148 (0.0059)	0.4613 (0.0111)
TACTiS-2	0.2764 (0.0479)	0.0573 (0.0106)	0.1797 (0.0061)	0.1263 (0.0166)	0.8299 (0.0062)	0.2747 (0.0023)	0.2114 (0.0506)	0.6944 (0.0855)
CoBiST-GAN	0.1467 (0.0038)	0.0220 (0.0032)	0.1418 (0.0038)	0.0900 (0.0023)	0.8154 (0.0071)	0.2568 (0.0054)	0.2971 (0.0205)	0.4241 (0.0258)
CoBiST-Flow	0.1382 (0.0013)	0.0205 (0.0007)	0.1369 (0.0018)	0.0787 (0.0004)	0.7967 (0.0123)	0.2402 (0.0018)	0.1462 (0.0027)	0.4286 (0.0088)
CoBiST-Diffusion	0.1370 (0.0011)	**0.0191** **(0.0002)**	**0.1349** **(0.0013)**	**0.0777** **(0.0009)**	**0.7859** **(0.0020)**	0.2416 (0.0004)	0.1107 (0.0051)	**0.4184** **(0.0071)**

E.3 Intervention-based interpretability diagnostics

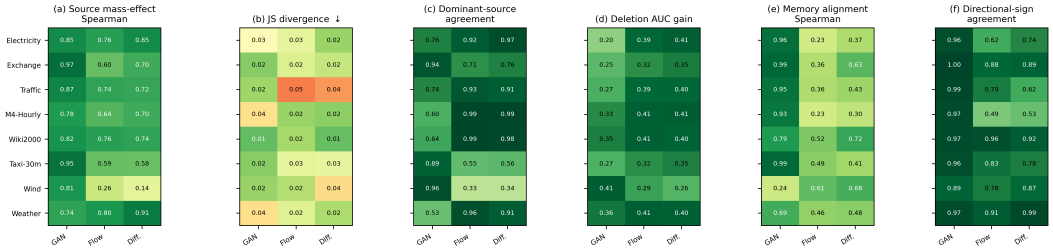

Figure 3: Dataset-level frozen-gate interpretability diagnostics. Columns in each panel correspond to GAN, Flow, and Diffusion, and all panels follow the dataset ordering printed in (a). Color ranges are fixed at $[-1, 1]$ for correlations, $[0, 0.06]$ for JS divergence, $[0, 1]$ for agreement, and $[0, 0.45]$ for deletion gain. The printed entries are the statistics used in the analysis.

The boundary columns complete the augmented row-stochastic accounting but carry zero-valued virtual states in the implementation. Hence a large μ_k^{∂} does not indicate the importance of observed-information: it indicates that part of the recurrent route budget survives to a finite-window sink. Under the standardized boundary injection described in Appendix D, boundary mass has dataset-macro Spearman correlations 0.83, 0.94, and 0.90 with the nonlinear readout response for GAN, Flow,

Table 13: Generator-level semantic validation. Entries are dataset-macro means with 95% dataset-bootstrap intervals in brackets. Up arrows indicate larger values are better and the down arrow indicates smaller values are better.

Model	Source Spearman ↑	JS ↓	Top-1 ↑	Deletion gain ↑	Memory Spearman ↑	Direction Spearman ↑	Sign agreement ↑	Boundary Spearman ↑
GAN	0.85 [0.80,0.90]	0.02 [0.02,0.03]	0.76 [0.66,0.86]	0.30 [0.26,0.35]	0.82 [0.63,0.95]	0.90 [0.78,0.97]	0.96 [0.94,0.98]	0.87 [0.75,0.95]
Flow	0.64 [0.52,0.74]	0.03 [0.02,0.03]	0.80 [0.63,0.94]	0.37 [0.34,0.39]	0.41 [0.32,0.50]	0.68 [0.55,0.81]	0.78 [0.68,0.88]	0.94 [0.90,0.97]
Diffusion	0.67 [0.50,0.79]	0.02 [0.02,0.03]	0.80 [0.64,0.94]	0.37 [0.34,0.40]	0.50 [0.40,0.60]	0.63 [0.47,0.78]	0.79 [0.69,0.88]	0.89 [0.82,0.95]

and Diffusion. Figure 4 shows both the within-cell rank relationship and the individual cells. This supports boundary mass as a truncation and virtual-sink sensitivity diagnostic, not as evidence that larger boundary mass improves forecasting.

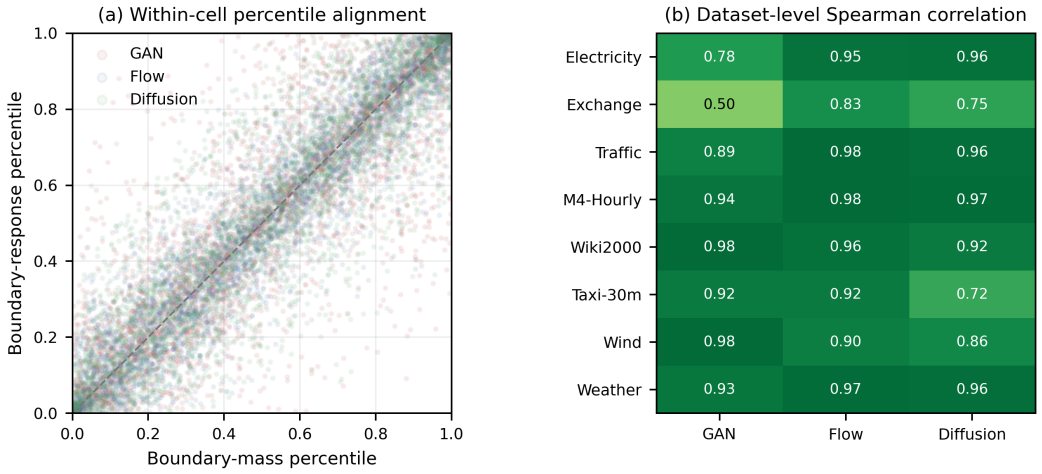

Figure 4: Validation of virtual-boundary semantics. (a) Boundary mass and the response to a standardized virtual-boundary injection are converted to within-cell percentiles before pooling, which removes generator-specific output scale. (b) Spearman correlations before pooling. The test concerns potential sensitivity to a virtual sink and does not treat boundary mass as an observed source contribution.