PROBABILISTIC END-TO-END HIERARCHICAL FORECASTING WITH CONDITIONAL FLOW MATCHING

Anonymous authors
Paper under double-blind review

ABSTRACT

Probabilistic forecasting in hierarchical settings requires generating distributions over future trajectories that satisfy strict aggregation constraints while capturing complex uncertainty. We propose Hierarchical Conditional Flow Matching (HCFM), an end-to-end framework in which future trajectories are generated by integrating a continuous-time dynamical system that combines a learnable residual flow with explicit terms for past consistency and hierarchical coherence. The framework supports two complementary instantiations: projected dynamics that preserve sample-wise coherence throughout integration, and soft-guided dynamics with exact endpoint reconciliation. In both cases, HCFM produces individually coherent forecast samples without committing to a fixed parametric output family. Across six public benchmarks, a single configuration (HCFM-UNet) attains the lowest overall scaled Continuous Ranked Probability Score (sCRPS) on five, with an average relative reduction of 14% against the strongest baseline (CLOVER) and competitive point-forecast accuracy.

1 INTRODUCTION

Many applications in industry require forecasts at multiple levels of granularity simultaneously. A retailer plans inventory at the product, category and store level; an energy operator balances load across substations, regions and the national grid; a transport authority allocates capacity from individual routes up to the network level. In all these cases, the quantities of interest are connected through known aggregation relations: the category forecast must equal the sum of its constituent products, the regional load must equal the sum of its substations, and so on. A forecast that violates these relations is called incoherent, and incoherent forecasts can lead to contradictory plans. Hierarchical forecasting addresses this problem: given a collection of time series organized by an aggregation structure, produce forecasts that are coherent; that is, they satisfy aggregation constraints at every forecast horizon (Hyndman et al., 2011; Athanasopoulos et al., 2024).

The classical approach decouples forecasting from coherence. A base model generates independent forecasts for each series, and a reconciliation step projects the result onto the coherent subspace (Hyndman et al., 2011; Wickramasuriya et al., 2019). While this simple and modular approach produces coherent point estimates, it is fundamentally limited when the goal is probabilistic forecasting; that is, when we want to identify the range of plausible futures and quantify their likelihoods. While reconciling a point forecast is a simple linear projection, reconciling a full predictive distribution is intrinsically harder, as projecting individual samples can distort the learned uncertainty structure.

Recent end-to-end probabilistic methods (Rangapuram et al., 2021; Olivares et al., 2024b;c; Utkarsh et al., 2026) enforce coherence within the forecasting model rather than relying on a separate reconciliation stage. However, they often do so through explicit parametric assumptions on the predictive distribution. HierE2E (Rangapuram et al., 2021), for example, draws samples from independent Gaussian marginals and reconciles them through a projection step, while CLOVER (Olivares et al., 2024c) imposes a multivariate Gaussian factor model to obtain coherence by construction. These choices are effective, but they restrict the class of forecast distributions that can be represented and may be poorly matched to settings with heavy tails, multimodality, or complex temporal dependence.

Generative models provide a natural alternative because they represent the predictive law through a learned sampling mechanism rather than a fixed parametric output family (Gouttes et al., 2021; Goodfellow et al., 2020). This gives them greater flexibility, but the resulting distribution is implicit:

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.

moments, quantiles, dependence structure, and scoring-rule estimates must be recovered from generated samples rather than read off from closed-form parameters (Diggle & Gratton, 1984; Cranmer et al., 2020). In hierarchical forecasting, this suggests viewing the task as conditional generation: given the observed past of a multivariate time series, generate future trajectories that follow the conditional predictive distribution and satisfy the aggregation constraints. This viewpoint recasts probabilistic hierarchical forecasting as a structured inverse problem, where past values are the partial observation, future values are the unknown trajectory, and coherence is a hard linear constraint that every generated solution must satisfy.

Continuous-time generative frameworks are well-suited to this formulation. Diffusion-style generators have shown strong performance in forecasting (Su et al., 2025), but they can struggle in limited-data regimes often experienced in hierarchical forecasting, where overfitting and performance deterioration are common (Moon et al., 2022; Zhang et al., 2024). This motivates a different design choice: enforcing structural constraints directly within the generative process rather than after sampling or through restrictive parameterizations. We therefore seek an end-to-end conditional generator that preserves the flexibility of sample-based modeling while enforcing past consistency and hierarchical coherence within the generation dynamics.

We propose Hierarchical Conditional Flow Matching (HCFM), an end-to-end framework for coherent probabilistic hierarchical forecasting. HCFM combines a learned trajectory generator, an ℓ_1 objective for robust training, and explicit past-consistency and coherence mechanisms. This modular formulation supports full-hierarchy generation and reduced HCFM configurations.

Our main contributions are:

1. We introduce HCFM, a flow-matching framework for coherent probabilistic hierarchical forecasting, combining an ℓ_1 objective for robust training with a past–future trajectory formulation anchored to the observed history.
2. We develop soft-guided dynamics with exact endpoint reconciliation and projected-residual dynamics that preserve forecast-horizon coherence throughout sampling.
3. Across six benchmarks, HCFM-UNet attains the lowest overall mean sCRPS on five, with competitive relSE and per-level scores. Several reduced HCFM configurations retain lower mean sCRPS than the previous state of the art in hierarchical forecasting.

2 Related Work

Work on coherent probabilistic hierarchical forecasting can be grouped into several broad families. One line of research aims to produce coherent forecast distributions within a single trainable model. HierE2E (Rangapuram et al., 2021) generates Gaussian forecasts and enforces coherence through a differentiable orthogonal projection at the output layer. CLOVER (Olivares et al., 2024c) instead achieves coherence by construction through a multivariate Gaussian factor model defined at the bottom level and aggregated upward. ProbHardE2E (Utkarsh et al., 2026) extends this perspective to a differentiable layer acting on distribution parameters and can accommodate nonlinear constraints. Related approaches include deep Poisson mixture networks (Olivares et al., 2024b) and top-down probabilistic decompositions (Das et al., 2023). These methods avoid post-hoc reconciliation, but coherence remains tied to a prescribed parametric family, most often Gaussian or a simple mixture model.

A second family follows the classical reconciliation paradigm, in which base forecasts are first produced and then projected onto the coherent subspace (Hyndman et al., 2011; Wickramasuriya et al., 2019; Athanasopoulos et al., 2024). These methods are principled for point forecasting, but they separate forecasting from reconciliation. As a result, the effect of the projection on uncertainty is difficult to characterize, and the framework is naturally better suited to coherent point forecasts than to coherent forecast distributions (Panagiotelis et al., 2023).

A third direction uses generative forecasting models based on diffusion, score matching, and continuous-time flows (Su et al., 2025; Rasul et al., 2021; Shen & Kwok, 2023; Tong et al., 2023; Kollovieh et al., 2025). These models provide flexible conditional generators, but do not by themselves account for hierarchical structure. Related constrained-generation methods therefore impose hard constraints at inference time: diffusion posterior sampling applies gradient-based measurement

corrections (Chung et al., 2022), while PCFM and ECI sampling constrain pretrained flows through continuous corrections or alternating extrapolation, correction, and interpolation steps (Utkarsh et al., 2025; Cheng et al., 2025). This strategy is effective, but assumes a strong pretrained generator and can be problematic in the limited-data regimes common to organisation-, region-, or hierarchy-specific forecasting.

3 Problem Setup

Hierarchical and grouped time series arise when measured quantities aggregate across a known structure. We consider a fixed hierarchy with D series and a total trajectory length $L = T + H$, which combines a past context window of length T and a forecast horizon of length H. Let $\widetilde{\mathbf{Y}}_{\text{past}} \in \mathbb{R}^{D\times T}$ denote the fixed, observed historical data.

We formulate our generative problem over a full reconstructed trajectory state:

$$\mathbf{X} = \left[\mathbf{Y}_{\text{past}}, \mathbf{Y}_{\text{future}}\right] \in \mathbb{R}^{D\times L}, \qquad \mathbf{X}_{1:T} := \mathbf{X}_{[:,1:T]}, \quad \mathbf{X}_{T+1:L} := \mathbf{X}_{[:,T+1:L]}.$$

Here, $\mathbf{Y}_{\text{future}} \in \mathbb{R}^{D\times H}$ represents the generated forecast, and $\mathbf{Y}_{\text{past}} \in \mathbb{R}^{D\times T}$ is an auxiliary reconstruction anchored to the observed history $\widetilde{\mathbf{Y}}_{\text{past}}$ by the past-consistency term. This term corrects model and solver drift. Co-evolving both blocks provides an inductive bias for learning past–future dependence (Appendix F.4). Conditioning and coherence are independent choices: a fixed past block also supports either coherence mechanism. We write $\mathbf{x}_\ell \in \mathbb{R}^D$ for the ℓ-th column of $\mathbf{X}$.

3.1 Hierarchy and Coherence

Coherence is a linear constraint defined by the hierarchy's aggregation relations. Let $\mathcal{C} \subseteq \mathbb{R}^D$ denote the coherent subspace induced by the hierarchy, and let $\mathbf{P} \in \mathbb{R}^{D\times D}$ be a projector onto $\mathcal{C}$. We write $\mathbf{Q} := \mathbf{I} - \mathbf{P}$ for the complementary projector, so that $\mathbf{Pz}$ is coherent and $\mathbf{Qz}$ is the incoherent residual of any series vector $\mathbf{z} \in \mathbb{R}^D$. A trajectory is coherent if and only if it is coherent at every time step,

$$\mathbf{Q}\mathbf{x}_\ell = \mathbf{0} \quad \forall \ell \in \{1, \dots, L\}, \qquad \text{equivalently} \qquad \mathbf{QX} = \mathbf{0}.$$

In our experiments, $\mathbf{P}$ is the bottom-up projector, which enforces coherence by summing bottom-level series up the hierarchy (Shlifer & Wolff, 1979). The generative dynamics act on per-series normalized trajectories $\mathbf{\Lambda}^{-1}\mathbf{X}$, where $\mathbf{\Lambda} = \operatorname{diag}(s_1, \dots, s_D)$ holds scales estimated from the observed context. In these coordinates we use $\mathbf{P}_{\mathbf{\Lambda}} := \mathbf{\Lambda}^{-1}\mathbf{P}\mathbf{\Lambda}$ and $\mathbf{Q}_{\mathbf{\Lambda}} := \mathbf{I} - \mathbf{P}_{\mathbf{\Lambda}}$; since $\mathbf{Q}_{\mathbf{\Lambda}}(\mathbf{\Lambda}^{-1}\mathbf{X}) = \mathbf{\Lambda}^{-1}\mathbf{QX}$, coherence in normalized coordinates is exactly coherence in original units (Appendix D.4). From Section 4 on, $\mathbf{X}$, $\widetilde{\mathbf{X}}$ and $\widetilde{\mathbf{Y}}^{\text{past}}$ denote normalized quantities.

3.2 Hierarchical Reconstruction as an Inverse Problem

We treat forecasting as trajectory imputation under coherence constraints. The first T columns of $\mathbf{X}$ should agree with the observed context $\widetilde{\mathbf{Y}}_{\text{past}}$, while the final H columns are unobserved and must be generated. To express the observation model over the full trajectory, define

$$\widetilde{\mathbf{X}} = \left[\widetilde{\mathbf{Y}}_{\text{past}}, \mathbf{0}_{D\times H}\right], \qquad \mathbf{M}_{\text{obs}} = \left[\mathbf{1}_{D\times T}, \mathbf{0}_{D\times H}\right] \in \{0,1\}^{D\times L}.$$

The data-consistency term is

$$\left\|\mathbf{M}_{\text{obs}} \odot (\mathbf{X} - \widetilde{\mathbf{X}})\right\|_F^2 = \left\|\mathbf{X}_{1:T} - \widetilde{\mathbf{Y}}_{\text{past}}\right\|_F^2,$$

where $\odot$ denotes entry-wise multiplication. Figure 1 illustrates this formulation.

Combining data consistency with coherence gives the point-estimate reconstruction problem

$$\min_{\mathbf{X}\in\mathbb{R}^{D\times L}} \left\|\mathbf{M}_{\text{obs}} \odot (\mathbf{X} - \widetilde{\mathbf{X}})\right\|_F^2 \qquad \text{s.t.} \qquad \mathbf{QX} = \mathbf{0}. \tag{1}$$

The objective enforces agreement with the observed past, while the constraint enforces exact hierarchical coherence at every time index. In forecasting, exact coherence is typically required on the generated future block; here, it is imposed on the full concatenated trajectory used for generation.

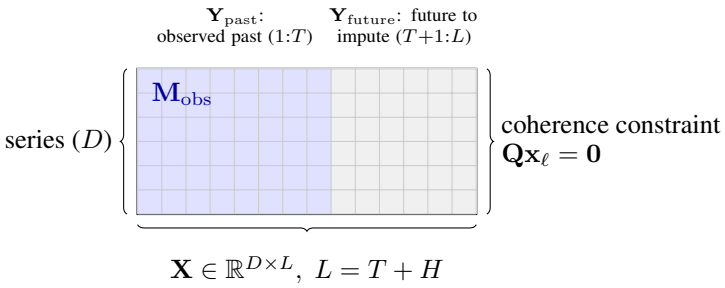


Figure 1: Hierarchical reconstruction task. We observe a past segment (first T columns) and must impute the future (last H columns) while enforcing coherence at every time step.

In the probabilistic setting, we adopt a sample-wise notion of coherence. A forecast distribution π over concatenated trajectories is coherent if

$$\pi\big(\{\mathbf{X} \in \mathbb{R}^{D\times L} : \mathbf{Q}\mathbf{X} = \mathbf{0}\}\big) = 1,$$

so that every generated trajectory is coherent at every time step. The problem in Eq. (1) is ill-posed because many future blocks can be consistent with the same observed context and coherence constraints. Our goal is therefore to sample

$$\mathbf{X} = \big[\mathbf{Y}_{\text{past}}, \mathbf{Y}_{\text{future}}\big] \sim \pi(\mathbf{X} \mid \widetilde{\mathbf{Y}}_{\text{past}}),$$

and use the generated future block $\mathbf{X}_{T+1:L}$ as a draw from the forecast distribution.

4 Method

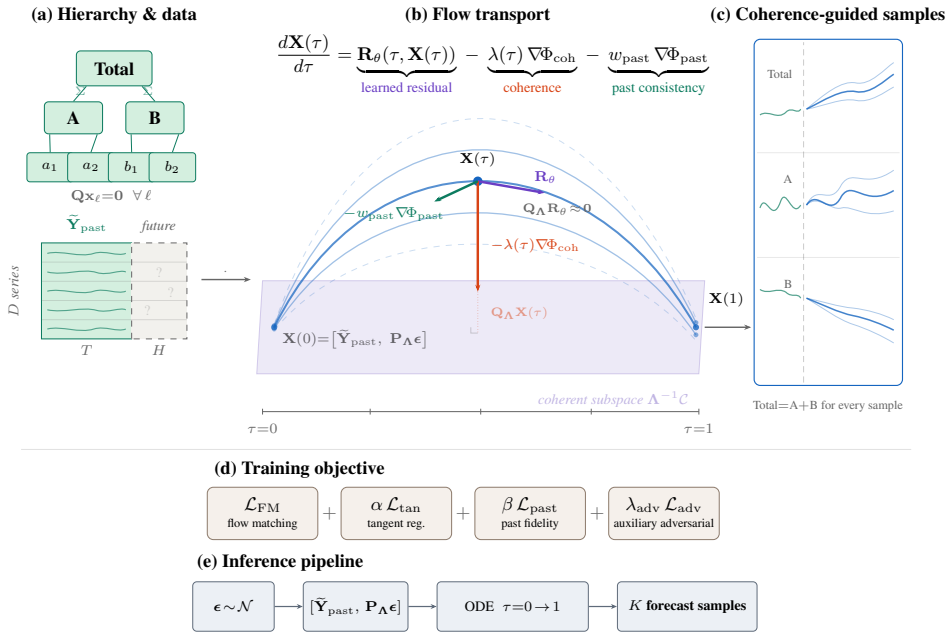


Figure 2: **Hierarchical Conditional Flow Matching (HCFM). (a)** A $D{=}7$ hierarchy (Σ: aggregation) enforces $\mathbf{Q}\mathbf{x}_\ell{=}\mathbf{0}$, with T observed and H forecast steps. **(b)** From $\mathbf{X}(0){=}[\widetilde{\mathbf{Y}}_{\text{past}}, \mathbf{P}_\Lambda\boldsymbol{\epsilon}]$, transport toward the coherent subspace $\boldsymbol{\Lambda}^{-1}\mathcal{C}$ combines a tangent residual $\mathbf{R}_\theta$ with coherence $-\lambda(\tau)\nabla\Phi_{\text{coh}}$ and past-consistency $-w_{\text{past}}\nabla\Phi_{\text{past}}$ gradients. **(c)** Projected residuals or soft-variant reconciliation yield K exactly coherent forecasts. **(d)** Training combines flow matching with tangent, past-fidelity, and optional adversarial regularization. **(e)** Inference integrates the ODE over $\tau \in [0, 1]$.

4.1 Vector Field Dynamics and Soft Coherence

The generative vector field is decomposed into explicit structural forces and a learned residual field $\mathbf{R}_\theta$ (Figure 2). The structural terms enforce data consistency with the observed past and hierarchical coherence, while $\mathbf{R}_\theta$ captures the remaining distributional correction needed to match the data. This decomposition is inspired by unrolled and plug-and-play methods for inverse problems (Romano et al., 2017; Aggarwal et al., 2019; Hammernik et al., 2018), where alternating consistency updates and learned refinements admit the continuous-time form

$$\frac{d\mathbf{X}(\tau)}{d\tau} = \mathbf{R}_\theta(\tau, \mathbf{X}(\tau)) - \nabla_{\mathbf{X}}\Phi(\mathbf{X}(\tau)),$$

$\tau \in [0,1]$ is the flow-matching integration time, a continuous variable that parameterizes the transport from the initial state $\mathbf{X}(0)$ to the generated endpoint $\mathbf{X}(1)$, and $\mathbf{X}(\tau) \in \mathbb{R}^{D\times L}$ is the flow state at arbitrary time τ. $\mathbf{R}_\theta : [0,1] \times \mathbb{R}^{D\times L} \to \mathbb{R}^{D\times L}$ is a learnable vector field, parameterized by θ, that maps the flow time τ and the current trajectory state $\mathbf{X}(\tau)$ to a velocity correction. The full derivation is given in Appendix C. We encode the two structural requirements as quadratic penalties with closed-form linear gradients. Recalling that $\mathbf{x}_\ell$ denotes the ℓ-th column of a generic trajectory matrix $\mathbf{X}$, hierarchical coherence is enforced by

$$\Phi_{\text{coh}}(\mathbf{X}) \triangleq \frac{1}{2}\sum_{\ell=1}^{L} \left\|\mathbf{Q}_{\mathbf{\Lambda}}\mathbf{x}_\ell\right\|_2^2, \qquad \nabla_{\mathbf{X}}\Phi_{\text{coh}}(\mathbf{X}) = \mathbf{Q}_{\mathbf{\Lambda}}^\top\mathbf{Q}_{\mathbf{\Lambda}}\mathbf{X}. \tag{2}$$

Agreement with the observed past is enforced by

$$\Phi_{\text{past}}(\mathbf{X}) \triangleq \frac{1}{2}\left\|\mathbf{M}_{\text{obs}} \odot (\mathbf{X} - \widetilde{\mathbf{X}})\right\|_F^2, \qquad \nabla_{\mathbf{X}}\Phi_{\text{past}}(\mathbf{X}) = \mathbf{M}_{\text{obs}} \odot (\mathbf{X} - \widetilde{\mathbf{X}}), \tag{3}$$

where $\mathbf{M}_{\text{obs}} \in \{0,1\}^{D\times L}$ selects the observed past block and $\widetilde{\mathbf{X}} = [\widetilde{\mathbf{Y}}_{\text{past}}, \mathbf{0}_{D\times H}]$ is the padded observation trajectory.

Putting these pieces together, we define the soft-guided HCFM dynamics as

$$\mathbf{v}_\theta(\tau, \mathbf{X}(\tau)) = \mathbf{R}_\theta(\tau, \mathbf{X}(\tau)) - \lambda(\tau)\nabla_{\mathbf{X}}\Phi_{\text{coh}}(\mathbf{X}(\tau)) - w_{\text{past}}\nabla_{\mathbf{X}}\Phi_{\text{past}}(\mathbf{X}(\tau)), \tag{4}$$

where $\lambda(\tau) \geq 0$ is a time-dependent coherence-guidance weight and $w_{\text{past}} \geq 0$ is the past-consistency weight. We keep w_{past} fixed and parameterize $\lambda(\tau)$ as a learned positive piecewise-constant schedule over flow time; details are given in Appendix D.

The method is modular with respect to the parameterization of $\mathbf{R}_\theta$. We study two choices: HCFM-MQCNN uses a one-dimensional temporal CNN applied independently to each series, following the MQCNN backbone of Olivares et al. (2024c). HCFM-UNet uses a two-dimensional U-Net that operates jointly on the $D \times L$ series-by-time grid, using multiscale convolutions to capture interactions across both series and time. Architectural details are presented in Appendix D.

4.2 Projected Residual

The field in Eq. (4) enforces coherence softly, through the guidance gradient. We also introduce a projected residual variant that enforces exact coherence on the generated future block by construction:

$$\mathbf{v}_\theta^{\text{proj}}(\tau, \mathbf{X}(\tau)) = \mathbf{P}_{\mathbf{\Lambda}}\mathbf{R}_\theta(\tau, \mathbf{X}(\tau)) - w_{\text{past}}\nabla_{\mathbf{X}}\Phi_{\text{past}}(\mathbf{X}(\tau)). \tag{5}$$

On any future column $\ell > T$ the past gradient vanishes, so $d\mathbf{x}_\ell/d\tau = (\mathbf{P}_{\mathbf{\Lambda}}\mathbf{R}_\theta)_\ell \in \text{range}(\mathbf{P}_{\mathbf{\Lambda}})$. Since the future is initialized with $\mathbf{Q}_{\mathbf{\Lambda}}\mathbf{x}_\ell(0) = \mathbf{0}$ (Section 4.3), $\mathbf{Q}_{\mathbf{\Lambda}}\mathbf{x}_\ell(\tau) = \mathbf{0}$ for all $\tau \in [0,1]$: forecasts are exactly coherent in original units at each integration step, at the cost of one application of $\mathbf{P}_{\mathbf{\Lambda}}$ per residual evaluation. The soft-guided variant (Eq. (4)) uses the schedule $\lambda(\tau)$ to control coherence strength (Appendix B). Appendix A bounds the curvature-dependent deviations introduced by ECI-style extrapolation–projection–interpolation corrections, motivating direct tangent integration.

4.3 Training and Inference

To learn the generative model, we build on conditional flow matching (CFM) (Lipman et al., 2023; Tong et al., 2023), which trains a continuous-time conditional generator by regressing a vector field to

a target transport velocity along an interpolant between sampled endpoints (Appendix J). For training, as illustrated in Figure 2(d), we construct paired flow endpoints over the concatenated state

$$\mathbf{X}^{\mathrm{data}} = \big[\widetilde{\mathbf{Y}}_{\mathrm{past}}, \mathbf{Y}_{\mathrm{future}}\big] \in \mathbb{R}^{D\times L},$$

using the observed past as the data-consistency block and coherent noise on the unobserved future block:

$$\mathbf{X}(1) = \mathbf{X}^{\mathrm{data}}, \qquad \mathbf{X}(0) = \mathbf{M}_{\mathrm{obs}} \odot \widetilde{\mathbf{X}} + (\mathbf{1}_{D\times L} - \mathbf{M}_{\mathrm{obs}}) \odot \mathbf{P}_{\mathbf{\Lambda}}\boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \quad (6)$$

where $\mathbf{M}_{\mathrm{obs}} \in \{0,1\}^{D\times L}$ is the full observation selector, $\widetilde{\mathbf{X}} = [\widetilde{\mathbf{Y}}_{\mathrm{past}}, \mathbf{0}_{D\times H}]$ is the masked observed trajectory, $\boldsymbol{\epsilon} \in \mathbb{R}^{D\times L}$ has i.i.d. standard normal entries, and $\mathbf{P}_{\mathbf{\Lambda}}\boldsymbol{\epsilon}$ applies $\mathbf{P}_{\mathbf{\Lambda}}$ column-wise. We then draw $\tau \sim \mathcal{U}([0,1])$ and interpolate $\mathbf{X}(\tau) = (1-\tau)\mathbf{X}(0) + \tau\mathbf{X}(1)$, using the flow-matching target $\mathbf{X}(1) - \mathbf{X}(0)$.

Both endpoints share the observed history, so $\mathbf{X}_{1:T}(\tau) = \widetilde{\mathbf{Y}}_{\mathrm{past}}$ along the training interpolant and the past-block target velocity is zero. The paired full-trajectory objective trains $\mathbf{R}_\theta$ to use the observed history when generating the future. During sampling, the past-consistency force anchors the auxiliary history as both blocks evolve.

We regress the vector field with an entrywise ℓ_1 loss for robustness (Appendix G):

$$\mathcal{L}_{\mathrm{FM}}(\theta) = \mathbb{E}_{(\mathbf{X}(0),\mathbf{X}(1)),\tau}\Big[\big\|\mathbf{v}_\theta(\tau, \mathbf{X}(\tau)) - (\mathbf{X}(1) - \mathbf{X}(0))\big\|_1\Big]. \quad (7)$$

Two residual regularizers align $\mathbf{R}_\theta(\tau, \mathbf{X}(\tau))$ with the structural forces: a tangency penalty that discourages incoherent residual motion, and a past penalty that suppresses residual motion on the observed context:

$$\mathcal{L}_{\tan}(\theta) = \mathbb{E}\big[\|\mathbf{Q}_{\mathbf{\Lambda}}\mathbf{R}_\theta(\tau, \mathbf{X}(\tau))\|_F^2\big], \qquad \mathcal{L}_{\mathrm{past}}(\theta) = \mathbb{E}\big[\|\mathbf{M}_{\mathrm{obs}} \odot \mathbf{R}_\theta(\tau, \mathbf{X}(\tau))\|_F^2\big]. \quad (8)$$

We use squared Frobenius penalties here to provide a low-variance training signal, while keeping the main regression Eq. (7) robust under ℓ_1. The base objective is

$$\mathcal{L}_{\mathrm{base}}(\theta) = \mathcal{L}_{\mathrm{FM}}(\theta) + \alpha\mathcal{L}_{\tan}(\theta) + \beta\mathcal{L}_{\mathrm{past}}(\theta), \quad (9)$$

with weights $\alpha, \beta \geq 0$.

For any idempotent $\mathbf{P}$, Appendix B bounds future incoherence by an exponentially damped initial error plus accumulated residual leakage, motivating the tangency regularizer in Eq. (8).

Entrywise ℓ_1 regression can induce within-mode contraction (Appendix G). We include an optional auxiliary adversarial regularizer to mitigate this underdispersion. A hinge-loss discriminator d_ψ scores rectified endpoints $\widehat{\mathbf{X}}_1(\tau) = \mathbf{X}(\tau) + (1-\tau)\mathbf{v}_\theta(\tau, \mathbf{X}(\tau))$, with generator loss $\mathcal{L}_{\mathrm{adv}}(\theta) = -\mathbb{E}[d_\psi(\widehat{\mathbf{X}}_1(\tau))]$. The objective is

$$\mathcal{L}(\theta) = \mathcal{L}_{\mathrm{base}}(\theta) + \lambda_{\mathrm{adv}}\mathcal{L}_{\mathrm{adv}}(\theta) \quad (10)$$

The default configuration uses $\lambda_{\mathrm{adv}} = 0.1$; setting $\lambda_{\mathrm{adv}} = 0$ recovers the base objective. Calibration and dispersion ablations appear in Appendix F.2, Table 6.

At inference time (Figure 2), we initialize $\mathbf{X}(0) = [\widetilde{\mathbf{Y}}_{\mathrm{past}}, \mathbf{P}_{\mathbf{\Lambda}}\boldsymbol{\epsilon}_{\mathrm{future}}]$ with fresh noise $\boldsymbol{\epsilon}_{\mathrm{future}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and integrate $d\mathbf{X}(\tau)/d\tau = \mathbf{v}_\theta(\tau, \mathbf{X}(\tau))$ from $\tau = 0$ to $\tau = 1$ using 20 Heun steps (Appendix F.3). The forecast is the future block of $\mathbf{\Lambda}\mathbf{X}(1)$. Projected residuals preserve forecast-horizon coherence throughout integration; soft guidance applies $\mathbf{P}$ to the unscaled forecast. Raw-endpoint coherence errors are reported in Table 5.

5 Experiments

We evaluate HCFM against coherent forecasters and examine internal HCFM ablations, covering forecast accuracy, hierarchical context, past evolution, coherence mechanisms, multivariate quality, loss terms, and sampling cost.

Table 1: Hierarchical forecasting datasets. Items is the number of independent hierarchical instances (each with the same hierarchy template). D denotes the total number of series per instance, consistent with the notation in Section 4. H denotes the forecast horizon.

Dataset	Items	Bottom-level series	Levels	Total series (D)	Time range	Frequency	Horizon (H)
`Labour`	1	32	4	57	2/1978–12/2019	Monthly	12
`Traffic`	1	200	4	207	1/2008–3/2009	Daily	1
`Tourism-S`	1	56	4	89	1998–2006	Quarterly	4
`Tourism-L`	1	304	4/5	555	1998–2016	Monthly	12
`Wiki`	1	150	5	199	1/2016–12/2016	Daily	7
`Favorita`	4036	54	4	93	1/2013–8/2017	Daily	34

5.1 Datasets

We evaluate HCFM on six public hierarchical forecasting datasets (Table 1). `Favorita` contains 4,036 product-level instances that share a common four-level hierarchy with daily sales series. The remaining datasets each contain a single hierarchical instance: `Tourism-S` and `Tourism-L` report Australian visitor counts grouped by geography and purpose. `Traffic` contains daily San Francisco Bay Area highway occupancy rates. `Labour` tracks monthly Australian employment, and `Wiki` summarizes daily page views by country, topic, and access type. More details are provided in Appendix E.

5.2 Baselines

We compare HCFM with four state-of-the-art coherent probabilistic forecasters: CLOVER (Olivares et al., 2024c), HierE2E (Rangapuram et al., 2021), ProbHardE2E (Utkarsh et al., 2026), Hier-Transformer-CNF (Wang et al., 2022), and the classical reconciliation baseline PERMBU-MinT (Wickramasuriya et al., 2019; Ben Taieb et al., 2017). For PERMBU-MinT, the unreconciled base forecasts are produced by ARIMA, using MinT reconciliation. HCFM-MQCNN and HCFM-UNet use bottom-up reconciliation and soft guidance. Projected residuals are evaluated in Appendix F.1. Appendix I.2 adds same-backbone leaf-only and N-HiTS (with bottom-up reconciliation) (Challu et al., 2023) baselines. CLOVER is reproduced from its paper and re-evaluated under our rolling-window protocol.

5.3 Hyperparameters

All loss weights ($\alpha = \beta = 0.1$, $w_{\text{past}} = 20$, $\lambda_{\text{adv}} = 0.1$), the guidance-schedule parameterization and the sampler (Heun, 20 steps) were set once and are shared by all six datasets and both backbones. None was tuned per dataset. The only dataset-dependent choices are the U-Net width and dropout, selected on validation sCRPS between two presets, and the MQCNN dilation schedule. Checkpoints are selected by validation sCRPS. Appendix D provides more details.

5.4 Evaluation Protocol

Prior work in coherent probabilistic hierarchical forecasting typically reports results on a single held-out test window, averaging over random seeds on that fixed split (Rangapuram et al., 2021; Olivares et al., 2024c; Utkarsh et al., 2026). This captures optimisation variance but not whether performance is stable across time, where temporal drift or a favorable test window can overstate gains. We therefore evaluate all methods on three non-overlapping chained test windows, ordered chronologically, each with its own training, validation, and test segments. The reported mean and standard deviation thus reflect variation across historical periods.

We report several complementary metrics. The primary metric is the scaled Continuous Ranked Probability Score (sCRPS), which measures distributional accuracy at each level of the hierarchy. The Continuous Ranked Probability Score for a target y and forecast distribution $\widetilde{Y}$ is $\mathrm{CRPS}(y, \widetilde{Y}) = \mathbb{E}|\widetilde{Y} - y| - \frac{1}{2}\mathbb{E}|\widetilde{Y} - \widetilde{Y}'|$, where $\widetilde{Y}'$ is an independent copy of $\widetilde{Y}$. It is a strictly proper scoring rule (Gneiting & Raftery, 2007). Let $\mathcal{G}$ be the set of hierarchical levels, and for a level $g \in \mathcal{G}$, let $\mathcal{I}_g$ be the set of series indices at that level. For a test window with forecast origin index ℓ_0 and horizon H,

the level-g sCRPS is

$$\mathrm{sCRPS}_g = \frac{\sum_{i\in\mathcal{I}_g}\sum_{\eta=1}^{H}\mathrm{CRPS}\big(y_{i,\ell_0+\eta},\,\widetilde{Y}_{i,\ell_0+\eta}\big)}{\sum_{i\in\mathcal{I}_g}\sum_{\eta=1}^{H}|y_{i,\ell_0+\eta}|}, \tag{11}$$

where $y_{i,\ell_0+\eta}$ is the observed value of series i at horizon step η and $\widetilde{Y}_{i,\ell_0+\eta}$ is the corresponding forecast distribution. The denominator normalises by the total absolute target magnitude at level g, making sCRPS scale-invariant across levels.

We also report the relative Squared Error (relSE), which compares the mean forecast $\bar{Y}_{i,\ell_0+\eta} = \mathbb{E}[\widetilde{Y}_{i,\ell_0+\eta}]$ to a naive last-value baseline that repeats the final observed value y_{i,ℓ_0} across the horizon:

$$\mathrm{relSE}_g = \frac{\sum_{i\in\mathcal{I}_g}\sum_{\eta=1}^{H}\big(y_{i,\ell_0+\eta}-\bar{Y}_{i,\ell_0+\eta}\big)^2}{\sum_{i\in\mathcal{I}_g}\sum_{\eta=1}^{H}\big(y_{i,\ell_0+\eta}-y_{i,\ell_0}\big)^2}. \tag{12}$$

A value of $\mathrm{relSE}_g < 1$ indicates the mean forecast improves on the naive baseline at level g. For both metrics, lower is better. We select all final models according to their validation performance.

Appendix I.1 reports relSE, Appendix I.3 evaluates multivariate dependence using energy and spatial variogram scores, and Appendix I.4 reports statistical tests and their limitations. Forecast visualizations appear in Appendix H.

5.5 Results

Forecast accuracy. HCFM-UNet attains the lowest overall mean sCRPS on five of six datasets (Table 2), with a 13.6% average reduction against CLOVER, the strongest baseline on each dataset (datasets weighted equally). Gains are largest on `Traffic`, `Wiki`, and `Tourism-L`. HCFM-UNet outperforms HCFM-MQCNN on five datasets. Sparse quarterly data and a small hierarchy favor MQCNN on `Tourism-S`, where CLOVER leads at the top level.

Point accuracy. HCFM's main gains concern distributional accuracy. CLOVER has the lowest relSE on `Traffic`, `Tourism-S`, and `Favorita`. HCFM-UNet leads on `Wiki` and `Tourism-L`, and PERMBU-MinT is competitive on `Labour` (Table 12).

Multivariate quality. With hierarchy levels pooled, an HCFM variant achieves the lowest mean for each of the energy (ES) and spatial variogram (VS) scores on all five datasets (Tables 14 and 15). CLOVER leads leaf-level ES on `Tourism-S`.

6 Analysis and Discussion

Shared forecasting formulation and hierarchical context. Leaves Only, Past-Frozen, and HCFM without the auxiliary adversarial loss retain lower mean overall sCRPS than CLOVER on `Traffic` and `Wiki` (Tables 2, 6, 10 and 13). Leaves Only retains our ℓ_1 trajectory forecaster and past consistency, with bottom-up aggregation ensuring coherence. Full-hierarchy Projected Residual lowers `Traffic` sCRPS from 0.0324 ± 0.0089 to 0.0209 ± 0.0049 (Tables 4 and 13). Both multivariate scores improve on `Traffic` and `Tourism-L`, with comparable results on `Labour`, `Tourism-S`, and `Wiki` (Table 11). Bottom-up aggregation preserves leaf values, so these gains arise during generation. HCFM-MQCNN processes each series independently within $\mathbf{R}_\theta$, while HCFM-UNet models multiscale interactions across the full $D \times L$ series–time grid, consistent with cross-series MLP gains (Olivares et al., 2024c). Smaller backbone gaps on `Tourism-S` and `Labour` suggest that past-consistency and coherence terms already capture more of the relevant structure.

Past evolution and coherence mechanisms. For the trained HCFM sampler, co-evolution lowers mean sCRPS, ES, and VS on all five datasets versus inference-time past freezing (Tables 10 and 11). Soft-guided and projected-residual dynamics show dataset-dependent accuracy tradeoffs (Tables 4 and 5). Both yield exactly coherent endpoints: soft guidance uses final reconciliation, while projected residuals preserve forecast-horizon coherence throughout integration.

Regression loss and predictive dispersion. The ℓ_1 objective lowers mean overall sCRPS and relSE for both backbones on `Wiki` (Table 9). The auxiliary adversarial term lowers CCE and widens

Table 2: Empirical evaluation of coherent probabilistic forecasts. Each cell reports mean scaled CRPS (sCRPS) $\pm$ standard deviation across 3 rolling chained cross-validated windows. Lower is better. HCFM entries use the default soft-guided dynamics in Eq. (4). Best and second-best results are determined from the unrounded mean; the reported standard deviations are descriptive and are not used as a tie-breaker.

Dataset	Level / Hierarchy	HCFM-MQCNN	HCFM-UNet	CLOVER	HierE2E	ProbHardE2E	Hier-Transformer-CNF	PERMBU-MinT*
Traffic	Overall	0.0936 ± 0.0142	$\mathbf{0.0245 \pm 0.0122}$	0.0419 ± 0.0229	0.0727 ± 0.0100	0.0777 ± 0.0116	0.1148 ± 0.0132	0.1351 ± 0.0848
	Level1	0.0611 ± 0.0119	$\mathbf{0.0125 \pm 0.0050}$	0.0181 ± 0.0093	0.0365 ± 0.0186	0.0284 ± 0.0152	0.0974 ± 0.0195	0.0982 ± 0.1027
	Level2	0.0594 ± 0.0106	$\mathbf{0.0128 \pm 0.0049}$	0.0185 ± 0.0086	0.0382 ± 0.0178	0.0344 ± 0.0113	0.0978 ± 0.0175	0.1011 ± 0.0934
	Level3	0.0615 ± 0.0043	$\mathbf{0.0172 \pm 0.0085}$	0.0233 ± 0.0156	0.0455 ± 0.0188	0.0430 ± 0.0082	0.0980 ± 0.0155	0.1001 ± 0.0871
	Level4	0.1923 ± 0.0349	$\mathbf{0.0554 \pm 0.0356}$	0.1075 ± 0.0621	0.1708 ± 0.0140	0.2051 ± 0.0268	0.1659 ± 0.0082	0.2408 ± 0.0682
Tourism-S	Overall	$\mathbf{0.0752 \pm 0.0183}$	0.0808 ± 0.0117	0.0769 ± 0.0092	0.0892 ± 0.0140	0.0941 ± 0.0476	0.1120 ± 0.0258	0.1160 ± 0.0197
	Country	0.0403 ± 0.0341	0.0403 ± 0.0198	$\mathbf{0.0392 \pm 0.0212}$	0.0444 ± 0.0314	0.0527 ± 0.0519	0.0675 ± 0.0366	0.0805 ± 0.0331
	Country/Purpose	0.0601 ± 0.0201	$\mathbf{0.0592 \pm 0.0143}$	0.0604 ± 0.0071	0.0771 ± 0.0133	0.0800 ± 0.0533	0.0977 ± 0.0272	0.0980 ± 0.0260
	Country/Purpose/State	$\mathbf{0.0899 \pm 0.0126}$	0.1013 ± 0.0100	0.0935 ± 0.0047	0.1078 ± 0.0056	0.1109 ± 0.0443	0.1325 ± 0.0207	0.1342 ± 0.0189
	Country/Purpose/State/CityNonCity	$\mathbf{0.1105 \pm 0.0104}$	0.1223 ± 0.0095	0.1144 ± 0.0103	0.1276 ± 0.0082	0.1328 ± 0.0429	0.1502 ± 0.0212	0.1513 ± 0.0136
Tourism-L	Overall	0.1466 ± 0.0165	$\mathbf{0.1366 \pm 0.0069}$	0.1489 ± 0.0271	0.2872 ± 0.2168	0.2292 ± 0.0830	0.2216 ± 0.0380	–
	Country	0.0667 ± 0.0186	$\mathbf{0.0529 \pm 0.0066}$	0.0766 ± 0.0415	0.2338 ± 0.2625	0.1188 ± 0.0398	0.1733 ± 0.0597	–
	Country/State	0.0853 ± 0.0112	$\mathbf{0.0787 \pm 0.0100}$	0.0951 ± 0.0300	0.2464 ± 0.2392	0.1522 ± 0.0497	0.1696 ± 0.0450	–
	Country/Purpose	0.0984 ± 0.0270	$\mathbf{0.0825 \pm 0.0117}$	0.0984 ± 0.0390	0.2602 ± 0.2342	0.1884 ± 0.0841	0.2011 ± 0.0572	–
	Country/State/Zone	0.1300 ± 0.0136	$\mathbf{0.1205 \pm 0.0078}$	0.1296 ± 0.0234	0.2669 ± 0.2185	0.2019 ± 0.0748	0.1926 ± 0.0347	–
	Country/State/Purpose	0.1380 ± 0.0196	$\mathbf{0.1269 \pm 0.0107}$	0.1404 ± 0.0283	0.2831 ± 0.2161	0.2382 ± 0.0978	0.2167 ± 0.0389	–
	Country/State/Zone/Region	0.1721 ± 0.0127	$\mathbf{0.1638 \pm 0.0053}$	0.1701 ± 0.0198	0.2920 ± 0.1936	0.2534 ± 0.0925	0.2228 ± 0.0279	–
	Country/State/Zone/Purpose	0.2077 ± 0.0168	$\mathbf{0.1983 \pm 0.0084}$	0.2056 ± 0.0200	0.3301 ± 0.1947	0.3026 ± 0.1065	0.2673 ± 0.0274	–
	Country/State/Zone/Region/Purpose	0.2745 ± 0.0140	$\mathbf{0.2695 \pm 0.0055}$	0.2754 ± 0.0157	0.3849 ± 0.1759	0.3784 ± 0.1206	0.3295 ± 0.0185	–
Labour	Overall	0.0096 ± 0.0013	$\mathbf{0.0093 \pm 0.0003}$	0.0095 ± 0.0005	0.0095 ± 0.0011	0.0165 ± 0.0031	0.0136 ± 0.0014	0.0218 ± 0.0065
	Country	0.0059 ± 0.0029	0.0049 ± 0.0012	0.0063 ± 0.0002	$\mathbf{0.0026 \pm 0.0002}$	0.0111 ± 0.0024	0.0086 ± 0.0030	0.0203 ± 0.0104
	Country/Region	$\mathbf{0.0078 \pm 0.0013}$	0.0081 ± 0.0001	0.0082 ± 0.0007	0.0095 ± 0.0012	0.0162 ± 0.0040	0.0126 ± 0.0016	0.0215 ± 0.0067
	Country/Gender/Region	0.0097 ± 0.0005	0.0095 ± 0.0004	$\mathbf{0.0090 \pm 0.0008}$	0.0101 ± 0.0014	0.0168 ± 0.0034	0.0136 ± 0.0012	0.0211 ± 0.0056
	Country/Employment/Gender/Region	0.0148 ± 0.0008	0.0145 ± 0.0005	$\mathbf{0.0144 \pm 0.0009}$	0.0159 ± 0.0023	0.0218 ± 0.0029	0.0197 ± 0.0005	0.0245 ± 0.0042
Wiki	Overall	0.1907 ± 0.0997	$\mathbf{0.1838 \pm 0.0981}$	0.2753 ± 0.0079	0.3285 ± 0.0018	0.3654 ± 0.1000	0.4431 ± 0.1630	0.4824 ± 0.1436
	Views	0.1163 ± 0.1210	$\mathbf{0.1121 \pm 0.1115}$	0.1283 ± 0.0316	0.1741 ± 0.0683	0.2353 ± 0.0753	0.3559 ± 0.1780	0.1835 ± 0.1264
	Views/Country	0.1597 ± 0.1038	$\mathbf{0.1522 \pm 0.1058}$	0.2290 ± 0.0193	0.2521 ± 0.0510	0.2836 ± 0.0697	0.3964 ± 0.1902	0.3389 ± 0.1595
	Views/Country/Access	0.1979 ± 0.0966	$\mathbf{0.1892 \pm 0.0957}$	0.2797 ± 0.0119	0.3126 ± 0.0141	0.3272 ± 0.0695	0.4397 ± 0.1461	0.4766 ± 0.1097
	Views/Country/Access/Agent	0.2015 ± 0.0980	$\mathbf{0.1925 \pm 0.0969}$	0.2959 ± 0.0253	0.3362 ± 0.0238	0.3495 ± 0.0596	0.4448 ± 0.1460	0.4913 ± 0.1104
	Views/Country/Access/Agent/Topic	0.2779 ± 0.0808	$\mathbf{0.2728 \pm 0.0807}$	0.4433 ± 0.0230	0.5675 ± 0.1427	0.6312 ± 0.2289	0.5787 ± 0.1560	0.9215 ± 0.2191
Favorita	Overall	0.3437 ± 0.0048	$\mathbf{0.3310 \pm 0.0214}$	0.3354 ± 0.0103	0.4788 ± 0.0090	0.4871 ± 0.0216	0.7857 ± 0.0324	0.4896 ± 0.0501
	Country	0.2588 ± 0.0094	0.2554 ± 0.0192	$\mathbf{0.2312 \pm 0.0114}$	0.3821 ± 0.0158	0.3880 ± 0.0281	0.7254 ± 0.2647	0.3809 ± 0.0583
	Country/State	0.3245 ± 0.0047	$\mathbf{0.3146 \pm 0.0166}$	0.3156 ± 0.0092	0.4560 ± 0.0087	0.4638 ± 0.0227	0.7627 ± 0.0638	0.4791 ± 0.0490
	Country/State/City	0.3436 ± 0.0043	$\mathbf{0.3319 \pm 0.0124}$	0.3394 ± 0.0115	0.4880 ± 0.0080	0.4969 ± 0.0210	0.7798 ± 0.0282	0.4996 ± 0.0504
	Country/State/City/Store	0.4479 ± 0.0038	$\mathbf{0.4221 \pm 0.0149}$	0.4551 ± 0.0100	0.5889 ± 0.0075	0.5997 ± 0.0173	0.8749 ± 0.2309	0.5987 ± 0.0446

*PERMBU-MinT requires a strict hierarchy and does not support grouped hierarchies like Tourism-L.

intervals on four of five datasets, consistent with mitigating the contraction discussed in Appendix G. On `Labour`, CCE increases and intervals narrow (Table 6).

Practical tradeoffs. Sampling uses 20 Heun steps by default. Latency grows with neural function evaluations. The MQCNN solver study reports the accuracy–latency tradeoff (Tables 7 and 8). The implementation assumes a fixed hierarchy and series ordering.

7 Conclusion

We introduce HCFM, a conditional flow-matching framework for probabilistic hierarchical forecasting that builds past consistency and hierarchical coherence directly into continuous-time generative dynamics. The projected-residual variant is exact by construction on the forecast horizon, while the default soft-guided variant enforces coherence continuously through guidance and tangency regularization, followed by exact endpoint reconciliation. The framework is modular with respect to the residual-field architecture.

HCFM-UNet attains the lowest overall mean sCRPS on five of six datasets (Table 2) and reduces overall sCRPS by 13.6% on average relative to CLOVER, weighting datasets equally. It is usually competitive at the per-level granularity, with `Tourism-S` favoring HCFM-MQCNN. The largest gains are on `Traffic` (42%), `Wiki` (33%), and `Tourism-L` (8%), highlighting the value of joint modeling over the series-by-time grid when the hierarchy is sufficiently structured. These results support incorporating known linear structure directly into generative dynamics in small-data hierarchical settings.

Several directions for improvement remain. HCFM's ODE sampling cost scales with neural function evaluations, which may limit large-scale or high-frequency use (Appendix F.3). The fixed hierarchy, reconciliation, and series ordering also reduce adaptability under structural change. Future work could pair constraint-aware dynamics with few-step/distilled samplers and permutation-equivariant or graph-based residuals, and support variable horizons via rollout or native variable-horizon forecasting.

AI USE STATEMENT

We used generative AI tools to assist with coding and manuscript editing, including copyediting, improving readability, and standardizing mathematical notation. The core logical structure and steps of the proofs were conceived entirely by the authors. AI assistance supported the written presentation of these proofs through editorial and notation refinements. Generative AI also assisted with reviewing the interpretation and presentation of experimental results. The authors take full responsibility for the final manuscript and accompanying code, including all mathematical claims, experimental results, and AI-assisted content.

REFERENCES

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Rafal Jozefowicz, Yangqing Jia, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Mike Schuster, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow, Large-scale machine learning on heterogeneous systems, November 2015.

Coen Adler, Yuxin Chang, Samar Abdi, Felix Draxler, and Padhraic Smyth. Beyond accuracy: Are time series foundation models well-calibrated? In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=nGBN7UjHcy.

Hemant Kumar Aggarwal, Merry P. Mani, and Mathews Jacob. Modl: Model-based deep learning architecture for inverse problems. *IEEE Transactions on Medical Imaging*, 38(2):394–405, 2019. doi: 10.1109/TMI.2018.2865356. arXiv:1712.02862.

Oren Anava, Vitaly Kuznetsov, and Google Inc. Web Traffic Time Series Forecasting. Kaggle Competition, 2018. URL https://www.kaggle.com/c/web-traffic-time-series-forecasting/.

Jason Ansel, Edward Yang, Horace He, Natalia Gimelshein, Animesh Jain, Michael Voznesensky, Bin Bao, Peter Bell, David Berard, Evgeni Burovski, Geeta Chauhan, Anjali Chourdia, Will Constable, Alban Desmaison, Zachary DeVito, Elias Ellison, Will Feng, Jiong Gong, Michael Gschwind, Brian Hirsh, Sherlock Huang, Kshiteej Kalambarkar, Laurent Kirsch, Michael Lazos, Mario Lezcano, Yanbo Liang, Jason Liang, Yinghai Lu, CK Luk, Bert Maher, Yunjie Pan, Christian Puhrsch, Matthias Reso, Mark Saroufim, Marcos Yukio Siraichi, Helen Suk, Michael Suo, Phil Tillet, Eikan Wang, Xiaodong Wang, William Wen, Shunting Zhang, Xu Zhao, Keren Zhou, Richard Zou, Ajit Mathews, Gregory Chanan, Peng Wu, and Soumith Chintala. PyTorch 2: Faster Machine Learning Through Dynamic Python Bytecode Transformation and Graph Compilation. In *29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2 (ASPLOS '24)*. ACM, April 2024. doi: 10.1145/3620665.3640366. URL https://docs.pytorch.org/assets/pytorch2-2.pdf.

Uri M. Ascher and Linda R. Petzold. *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. Society for Industrial and Applied Mathematics, Philadelphia, 1998. ISBN 978-0-89871-412-8.

George Athanasopoulos, Rob J. Hyndman, Nikolaos Kourentzes, and Anastasios Panagiotelis. Forecast reconciliation: A review. *International Journal of Forecasting*, 40(2):430–456, 2024. doi: 10.1016/j.ijforecast.2023.10.010.

Australian Bureau of Statistics. Labour Force, Australia. Accessed Online, 2019. URL https://www.abs.gov.au/AUSSTATS/abs@.nsf/DetailsPage/6202.0Dec%202019?OpenDocument.

Souhaib Ben Taieb and Bonsoo Koo. Regularized regression for hierarchical forecasting without unbiasedness conditions. In *Proceedings of the 25th ACM SIGKDD International Conference*

on Knowledge Discovery & Data Mining, KDD '19, pp. 1337–1347, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450362016. doi: 10.1145/3292500.3330976. URL https://doi.org/10.1145/3292500.3330976.

Souhaib Ben Taieb, James W. Taylor, and Rob J. Hyndman. Coherent probabilistic forecasts for hierarchical time series. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3348–3357. PMLR, 2017. URL https://proceedings.mlr.press/v70/taieb17a.html.

John C. Butcher. *Numerical Methods for Ordinary Differential Equations*. John Wiley & Sons, New York, 2003. ISBN 978-0-471-96758-3.

C. Challu, K. G. Olivares, B. N. Oreshkin, F. Garza Ramirez, M. Mergenthaler Canseco, and A. Dubrawski. NHITS: Neural Hierarchical Interpolation for Time Series Forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(6):6989–6997, 2023.

Chaoran Cheng, Boran Han, Danielle C. Maddix, Abdul Fatir Ansari, Andrew Stuart, Michael W. Mahoney, and Bernie Wang. Gradient-free generation for hard-constrained systems. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=teE4pl9ftK.

Hyungjin Chung, Jeongsol Kim, Michael T. McCann, Marc L. Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. *arXiv preprint arXiv:2209.14687*, 2022. doi: 10.48550/arXiv.2209.14687. URL https://arxiv.org/abs/2209.14687.

Corporación Favorita, inversion, Julia Elliott, and Mark McDonald. Corporación Favorita Grocery Sales Forecasting. Kaggle Competition, 2017. URL https://kaggle.com/competitions/favorita-grocery-sales-forecasting.

Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, 2020. doi: 10.1073/pnas.1912789117.

Casper da Costa-Luis, Stephen Karl Larroque, Kyle Altendorf, Hadrien Mary, richardsheridan, Mikhail Korobov, Noam Raphael, Ivan Ivanov, Marcel Bargull, Nishant Rodrigues, Shawn, Mikhail Dektiarev, Michał Górny, mjstevens777, Matthew D. Pagel, Martin Zugnoni, CrazyPython, Charles Newey, Antony Lee, pgajdos, Todd, Staffan Malmgren, redbug312, Orivej Desh, Nikolai Nechaev, Mike Boyle, Max Nordlund, MapleCCC, Jack McCracken, and Hugo van Kemenade. tqdm: A fast, extensible progress bar for python and cli, February 2026. URL https://doi.org/10.5281/zenodo.18473238.

A. Das, W. Kong, B. Paria, and R. Sen. Dirichlet proportions model for hierarchically coherent probabilistic forecasting. In *Proceedings of the Thirty-Ninth Conference on Uncertainty in Artificial Intelligence*, volume 216 of *Proceedings of Machine Learning Research*, pp. 518–528. PMLR, 2023. URL https://proceedings.mlr.press/v216/das23b.html.

Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7(1):1–30, 2006. URL http://jmlr.org/papers/v7/demsar06a.html.

Peter J. Diggle and Richard J. Gratton. Monte carlo methods of inference for implicit statistical models. *Journal of the Royal Statistical Society: Series B (Methodological)*, 46(2):193–212, 1984. doi: 10.1111/j.2517-6161.1984.tb01290.x.

William Falcon and The PyTorch Lightning team. PyTorch Lightning, March 2019. URL https://github.com/Lightning-AI/lightning.

Erwin Fehlberg. Low-Order Classical Runge-Kutta Formulas with Stepsize Control and Their Application to Some Heat Transfer Problems. NASA Technical Report R-315, National Aeronautics and Space Administration, 1969.

Tilmann Gneiting and Adrian E. Raftery. Strictly Proper Scoring Rules, Prediction, and Estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. doi: 10.1145/3422622.

Adèle Gouttes, Kashif Rasul, Mateusz Koren, Johannes Stephan, and Tofigh Naghibi. Probabilistic time series forecasting with implicit quantile networks, 2021. URL https://arxiv.org/abs/2107.03743.

Kerstin Hammernik, Teresa Klatzer, Erich Kobler, Michael P. Recht, Daniel K. Sodickson, Thomas Pock, and Florian Knoll. Learning a variational network for reconstruction of accelerated MRI data. *Magnetic Resonance in Medicine*, 79(6):3055–3071, 2018. doi: 10.1002/mrm.26977. URL https://doi.org/10.1002/mrm.26977.

Charles R. Harris, K. Jarrod Millman, Stéfan J van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585:357–362, 2020. doi: 10.1038/s41586-020-2649-2.

Rob J. Hyndman, Roman A. Ahmed, George Athanasopoulos, and Han Lin Shang. Optimal combination forecasts for hierarchical time series. *Computational Statistics & Data Analysis*, 55(9): 2579–2589, 2011. doi: 10.1016/j.csda.2011.03.006. URL https://robjhyndman.com/publications/hierarchical/.

Marcel Kollovieh, Marten Lienen, David Lüdke, Leo Schwinn, and Stephan Günnemann. Flow matching with gaussian process priors for probabilistic time series forecasting. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=uxVBbSlKQ4.

Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2noise: Learning image restoration without clean data, 2018. URL https://arxiv.org/abs/1803.04189.

Shanchuan Lin, Ceyuan Yang, Zhijie Lin, Hao Chen, and Haoqi Fan. Adversarial flow models. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=3GPPUHlJU9.

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

Étienne Marcotte, Valentina Zantedeschi, Alexandre Drouin, and Nicolas Chapados. Regions of reliability in the evaluation of multivariate probabilistic forecasts. In *Proceedings of the 40th International Conference on Machine Learning*, 2023.

Taehong Moon, Moonseok Choi, Gayoung Lee, Jung-Woo Ha, and Juho Lee. Fine-tuning diffusion models with limited data. In *NeurIPS 2022 Workshop on Score-Based Methods*, 2022. URL https://openreview.net/forum?id=0J6afk9DqrR.

Nixtla. DatasetsForecast: Large scale time series forecasting datasets, 2023. URL https://github.com/Nixtla/datasetsforecast.

Kin G. Olivares, Azul Garza, David Luo, Cristian Challú, Max Mergenthaler, Souhaib Ben Taieb, Shanika L. Wickramasuriya, and Artur Dubrawski. Hierarchicalforecast: A reference framework for hierarchical forecasting in python. *arXiv*, 2024a. URL https://arxiv.org/abs/2207.03517.

Kin G. Olivares, O. Nganba Meetei, Ruijun Ma, Rohan Reddy, Mengfei Cao, and Lee Dicker. Probabilistic hierarchical forecasting with deep Poisson mixtures. *International Journal of Forecasting*, 40(2):470–489, 2024b. doi: 10.1016/j.ijforecast.2023.04.007. URL https://doi.org/10.1016/j.ijforecast.2023.04.007.

Kin G. Olivares, Geoffrey Négiar, Ruijun Ma, Oinam Nganba Meetei, Mengfei Cao, and Michael W. Mahoney. CLOVER: Probabilistic forecasting with coherent learning objective reparameterization. *Transactions on Machine Learning Research*, 2024c. ISSN 2835-8856. URL https://openreview.net/forum?id=q7YXEbFOAt.

Anastasios Panagiotelis, Puwasala Gamakumara, George Athanasopoulos, and Rob J. Hyndman. Probabilistic forecast reconciliation: Properties, evaluation and score optimisation. *European Journal of Operational Research*, 306(2):693–706, 2023. ISSN 0377-2217. doi: https://doi.org/10.1016/j.ejor.2022.07.040. URL https://www.sciencedirect.com/science/article/pii/S0377221722006087.

Syama Sundar Rangapuram, Lucien D. Werner, Konstantinos Benidis, Pedro Mercado, Jan Gasthaus, and Tim Januschowski. End-to-end learning of coherent probabilistic forecasts for hierarchical time series. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8832–8843. PMLR, 2021. URL https://proceedings.mlr.press/v139/rangapuram21a.html.

Kashif Rasul, Calvin Seward, Ingmar Schuster, and Roland Vollgraf. Autoregressive denoising diffusion models for multivariate probabilistic time series forecasting. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8857–8868. PMLR, 2021. URL https://proceedings.mlr.press/v139/rasul21a.html.

Giampaolo Rodolà. psutil: Cross-platform library for process and system monitoring in Python, 2024. URL https://github.com/giampaolo/psutil.

Yaniv Romano, Michael Elad, and Peyman Milanfar. The little engine that could: Regularization by denoising (RED). *SIAM Journal on Imaging Sciences*, 10(4):1804–1844, 2017. doi: 10.1137/16M1102884. URL https://doi.org/10.1137/16M1102884.

Michael Scheuerer and Thomas M. Hamill. Variogram-Based Proper Scoring Rules for Probabilistic Forecasts of Multivariate Quantities. *Monthly Weather Review*, 143(4):1321–1334, 2015.

Lifeng Shen and James Kwok. Non-autoregressive conditional diffusion models for time series prediction. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 31016–31029. PMLR, 2023. URL https://proceedings.mlr.press/v202/shen23d.html.

E. Shlifer and R. W. Wolff. Aggregation and Proration in Forecasting. *Management Science*, 25 (6):594–603, June 1979. doi: 10.1287/mnsc.25.6.594. URL https://www.jstor.org/stable/2630330.

Kirill Simonov, Ingy döt Net, and The YAML and Python Communities. PyYAML: YAML parser and emitter for Python, 2024. URL https://github.com/yaml/pyyaml.

Weixi Song, Zhetao Chen, Tao Xu, Xianchao Zeng, Xinyu Zhou, Lixin Yang, Donglin Wang, Cewu Lu, and Yong-Lu Li. L1 sample flow for efficient visuomotor learning, 2025. URL https://arxiv.org/abs/2511.17898.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=PxTIG12RRHS.

Chen Su, Zhengzhou Cai, Yuanhe Tian, Zhuochao Chang, Zihong Zheng, and Yan Song. Diffusion models for time series forecasting: A survey, 2025. URL https://arxiv.org/abs/2507.14507.

The pandas development team. pandas-dev/pandas: Pandas, 2010. URL https://github.com/pandas-dev/pandas.

Alexander Tong, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Kilian Fatras, Guy Wolf, and Yoshua Bengio. Conditional flow matching: Simulation-free dynamic optimal transport. *arXiv preprint arXiv:2302.00482*, 2023. doi: 10.48550/ARXIV.2302.00482. URL https://arxiv.org/abs/2302.00482.

Tourism Australia, Canberra. Tourism Research Australia (2005), Travel by Australians. Kaggle, 2005. URL https://www.kaggle.com/luisblanche/quarterly-tourism-in-australia/.

Utkarsh Utkarsh, Pengfei Cai, Alan Edelman, Rafael Gomez-Bombarelli, and Christopher Vincent Rackauckas. Physics-constrained flow matching: Sampling generative models with hard constraints. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=cf4etwjY7n.

Utkarsh Utkarsh, Danielle C. Maddix, Ruijun Ma, Michael W. Mahoney, and Bernie Wang. End-to-end probabilistic framework for learning with hard constraints. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=RPowYXiRmW.

Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2. URL https://doi.org/10.1038/s41592-019-0686-2.

Shiyu Wang, Fan Zhou, Yinbo Sun, Lintao Ma, James Zhang, and Yangfei Zheng. End-to-end modeling of hierarchical time series using autoregressive transformer and conditional normalizing flow-based reconciliation. In *2022 IEEE International Conference on Data Mining Workshops (ICDMW)*, pp. 1087–1094, 2022. doi: 10.1109/ICDMW58026.2022.00141. URL https://doi.org/10.1109/ICDMW58026.2022.00141.

Shanika L. Wickramasuriya, George Athanasopoulos, and Rob J. Hyndman. Optimal forecast reconciliation for hierarchical and grouped time series through trace minimization. *Journal of the American Statistical Association*, 114(526):804–819, 2019. doi: 10.1080/01621459.2018.1448825. URL https://robjhyndman.com/publications/mint/.

Omry Yadan. Hydra - a framework for elegantly configuring complex applications. Github, 2019. URL https://github.com/facebookresearch/hydra.

Omry Yadan, Jasha Sommer-Simpson, and Olivier Delalleau. omegaconf, November 2019. URL https://github.com/omry/omegaconf.

Hengrui Zhang, Jiani Zhang, Zhengyuan Shen, Balasubramaniam Srinivasan, Xiao Qin, Christos Faloutsos, Huzefa Rangwala, and George Karypis. Mixed-type tabular data synthesis with score-based diffusion in latent space. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=4Ay23yeuz0.

A TANGENT FLOWS VS. PROJECTION CORRECTIONS

In Appendices A and B, $\mathbf{P}$ and $\mathbf{Q}$ denote the normalized-coordinate operators $\mathbf{P}_{\mathbf{\Lambda}}$ and $\mathbf{Q}_{\mathbf{\Lambda}}$ of Appendix D.4; only idempotence is used.

The hard constraint set for a trajectory is the linear subspace where coherence is satisfied at every time step,

$$\mathcal{C}_L := \{\mathbf{X} \in \mathbb{R}^{D\times L} : \mathbf{Q}\mathbf{X} = \mathbf{0}\}.$$

We denote by $\mathbf{P}$ an idempotent projection onto $\mathcal{C}_L$, with $\mathbf{Q} = \mathbf{I} - \mathbf{P}$.

Consider an ideal constrained ODE

$$\frac{d\mathbf{X}^\star_\tau}{d\tau} = \mathbf{v}^\star(\tau, \mathbf{X}^\star_\tau), \qquad \mathbf{X}^\star_\tau \in \mathcal{C}_L \ \ \forall \tau \in [0,1]. \tag{13}$$

Since $\mathbf{Q}$ is a linear operator, feasibility along the trajectory is equivalent to the tangency condition

$$\mathbf{Q}\mathbf{v}^\star(\tau, \mathbf{X}^\star_\tau) = \mathbf{0} \quad \text{for all } \tau. \tag{14}$$

A.1 ECI CORRECTION MAP AND THE INDUCED DEVIATION

An ECI correction at time $\tau \in [0, 1)$ replaces the current state $\mathbf{X}_\tau$ by

$$\mathcal{T}_\tau(\mathbf{X}_\tau) := (1-\tau)\,\mathbf{X}_0 + \tau\,\mathbf{P}\Big(\mathbf{X}_\tau + (1-\tau)\,\mathbf{v}(\tau, \mathbf{X}_\tau)\Big), \tag{15}$$

defined for an arbitrary vector field $\mathbf{v}(\tau, \mathbf{X}_\tau)$, where $\mathbf{X}_0$ is the fixed conditional initialization for that sample. In the analysis below we evaluate this map on the ideal trajectory using the ideal field $\mathbf{v}^\star$ (i.e., set $\mathbf{v} = \mathbf{v}^\star$). This is the mechanism used in hard-constraint rectified-flow guidance: form a rectified endpoint estimate, project it, then map it back to time τ.

Fix a sample with initialization $\mathbf{X}_0$ and terminal state $\mathbf{X}^\star_1$ and define the time-dependent effective endpoint

$$\mathbf{X}^\star_{1|\tau} := \mathbf{X}_0 + \frac{\mathbf{X}^\star_\tau - \mathbf{X}_0}{\tau}, \qquad \tau \in (0, 1].$$

Then

$$\mathbf{X}^\star_\tau = (1-\tau)\mathbf{X}_0 + \tau\,\mathbf{X}^\star_{1|\tau}.$$

Because $\mathcal{C}_L$ is a linear subspace and both $\mathbf{X}_0$ and $\mathbf{X}^\star_\tau$ lie in $\mathcal{C}_L$ (for the future block where exact coherence is required), we also have $\mathbf{X}^\star_{1|\tau} \in \mathcal{C}_L$.

Apply Eq. (15) at time τ to the ideal state $\mathbf{X}^\star_\tau$, using the ideal vector field $\mathbf{v}^\star$. Define the rectified endpoint estimate (computed at time τ)

$$\widehat{\mathbf{X}}^\star_{1|\tau} := \mathbf{X}^\star_\tau + (1-\tau)\,\mathbf{v}^\star\big(\tau, \mathbf{X}^\star_\tau\big).$$

Then, substituting $\mathbf{X}^\star_\tau$ into Eq. (15) yields

$$\mathcal{T}_\tau(\mathbf{X}^\star_\tau) = (1-\tau)\mathbf{X}_0 + \tau\,\mathbf{P}(\widehat{\mathbf{X}}^\star_{1|\tau}).$$

Subtracting $\mathbf{X}^\star_\tau = (1-\tau)\mathbf{X}_0 + \tau\mathbf{X}^\star_{1|\tau}$ gives the exact deviation identity

$$\mathcal{T}_\tau(\mathbf{X}^\star_\tau) - \mathbf{X}^\star_\tau = \tau\big(\mathbf{P}(\widehat{\mathbf{X}}^\star_{1|\tau}) - \mathbf{X}^\star_{1|\tau}\big), \qquad \big\|\mathcal{T}_\tau(\mathbf{X}^\star_\tau) - \mathbf{X}^\star_\tau\big\|_F = \tau\,\big\|\mathbf{P}(\widehat{\mathbf{X}}^\star_{1|\tau}) - \mathbf{X}^\star_{1|\tau}\big\|_F. \tag{16}$$

This is the perturbation term induced by a single ECI correction.

A.2 SECOND-ORDER ERROR OF RECTIFIED ENDPOINT EXTRAPOLATION

Assume $\mathbf{X}^\star$ is twice differentiable and define

$$\kappa(\tau) := \sup_{s\in[0,\tau]} \Big\|\frac{d^2\mathbf{X}^\star_s}{ds^2}\Big\|_F.$$

Using the fundamental theorem of calculus and integration by parts,

$$\mathbf{X}^\star_\tau - \mathbf{X}_0 = \int_0^\tau \frac{d\mathbf{X}^\star_s}{ds}\,ds = \tau\,\frac{d\mathbf{X}^\star_\tau}{d\tau} - \int_0^\tau s\,\frac{d^2\mathbf{X}^\star_s}{ds^2}\,ds.$$

Since

$$\mathbf{X}^\star_{1|\tau} = \mathbf{X}_0 + \frac{\mathbf{X}^\star_\tau - \mathbf{X}_0}{\tau}, \qquad \widehat{\mathbf{X}}^\star_{1|\tau} = \mathbf{X}^\star_\tau + (1-\tau)\frac{d\mathbf{X}^\star_\tau}{d\tau},$$

it follows that

$$\mathbf{X}^\star_{1|\tau} - \widehat{\mathbf{X}}^\star_{1|\tau} = (1-\tau)\Big(\frac{\mathbf{X}^\star_\tau - \mathbf{X}_0}{\tau} - \frac{d\mathbf{X}^\star_\tau}{d\tau}\Big) = -\frac{1-\tau}{\tau}\int_0^\tau s\,\frac{d^2\mathbf{X}^\star_s}{ds^2}\,ds.$$

Taking Frobenius norms and bounding $\left\|\frac{d^2\mathbf{X}^\star_s}{ds^2}\right\|_F$ by $\kappa(\tau)$ yields

$$\|\mathbf{X}^\star_{1|\tau} - \widehat{\mathbf{X}}^\star_{1|\tau}\|_F \le \frac{1}{2}\,\kappa(\tau)\,\tau(1-\tau). \tag{17}$$

Unless $\frac{d^2\mathbf{X}^\star_s}{ds^2} \equiv \mathbf{0}$ on $[0,\tau]$ (constant-velocity flow), the right-hand side is strictly positive for some τ, so the rectified endpoint extrapolation has an unavoidable curvature-controlled error.

A.3 Bounding the ECI perturbation and comparing with tangent flows

By construction, $\mathbf{X}^\star_\tau \in \mathcal{C}_L$. Moreover, by the tangency condition Eq. (14),

$$\mathbf{Q}\mathbf{v}^\star(\tau, \mathbf{X}^\star_\tau) = \mathbf{0}.$$

Therefore,

$$\mathbf{Q}\widehat{\mathbf{X}}^\star_{1|\tau} = \mathbf{Q}\mathbf{X}^\star_\tau + (1-\tau)\mathbf{Q}\mathbf{v}^\star(\tau, \mathbf{X}^\star_\tau) = \mathbf{0}.$$

Thus $\widehat{\mathbf{X}}^\star_{1|\tau} \in \mathcal{C}_L$, and since $\mathbf{P}$ is a projection onto $\mathcal{C}_L$,

$$\mathbf{P}(\widehat{\mathbf{X}}^\star_{1|\tau}) = \widehat{\mathbf{X}}^\star_{1|\tau}.$$

Hence

$$\|\mathbf{P}(\widehat{\mathbf{X}}^\star_{1|\tau}) - \mathbf{X}^\star_{1|\tau}\|_F = \|\widehat{\mathbf{X}}^\star_{1|\tau} - \mathbf{X}^\star_{1|\tau}\|_F.$$

Combining this equality with Eq. (16) and Eq. (17) gives

$$\big\|\mathcal{T}_\tau(\mathbf{X}^\star_\tau) - \mathbf{X}^\star_\tau\big\|_F \le \tau\,\|\mathbf{X}^\star_{1|\tau} - \widehat{\mathbf{X}}^\star_{1|\tau}\|_F \le \frac{\tau^2}{2}\,\kappa(\tau)\,(1-\tau). \tag{18}$$

Equation (18) is the additional perturbation term referenced in the main text discussion in Section 4.2. The comparison with tangent constrained flows is immediate: solving Eq. (13) does not apply any instantaneous replacement map, so it does not incur this term.

B Continuous Coherence Guidance Bound

This appendix derives Eq. (19) on the forecast horizon and states the corresponding trajectory-level version.

Assume that $\mathbf{X}(\tau)$ is absolutely continuous in τ, that the residual terms below are integrable along the trajectory, and that

$$\lambda(\tau) \ge 0 \quad \text{for almost every } \tau \in [0,1].$$

All differential identities and inequalities below are understood to hold almost everywhere. Define

$$\Gamma(\tau) := \int_0^\tau \lambda(u)\,du.$$

Fix a future time index $\ell > T$. By Eq. (3),

$$\big(\nabla_{\mathbf{X}}\Phi_{\text{past}}(\mathbf{X}(\tau))\big)_\ell = \mathbf{0}.$$

Assume that $\mathbf{P}$ is idempotent and define

$$\mathbf{Q} := \mathbf{I} - \mathbf{P}.$$

Then

$$\mathbf{P}^2 = \mathbf{P}, \qquad \mathbf{Q}^2 = \mathbf{Q}, \qquad \mathbf{QP} = \mathbf{0}.$$

For the Euclidean coherence penalty in Eq. (2),

$$\nabla_{\mathbf{X}} \Phi_{\mathrm{coh}}(\mathbf{X}) = \mathbf{Q}^\top \mathbf{Q}\mathbf{X}.$$

When $\mathbf{P}$ is orthogonal, this reduces to $\mathbf{QX}$. The argument below only uses idempotence.

On a future slice $\ell > T$, the guided dynamics satisfy

$$\frac{d\mathbf{x}_\ell(\tau)}{d\tau} = \big(\mathbf{R}_\theta(\tau, \mathbf{X}(\tau))\big)_\ell - \lambda(\tau)\mathbf{Q}^\top \mathbf{Q}\mathbf{x}_\ell(\tau).$$

Multiplying by $\mathbf{Q}$ gives

$$\frac{d}{d\tau}\big(\mathbf{Q}\mathbf{x}_\ell(\tau)\big) = \mathbf{Q}\big(\mathbf{R}_\theta(\tau, \mathbf{X}(\tau))\big)_\ell - \lambda(\tau)\mathbf{Q}\mathbf{Q}^\top \mathbf{Q}\mathbf{x}_\ell(\tau).$$

Taking the Euclidean inner product with $\mathbf{Q}\mathbf{x}_\ell(\tau)$, we obtain

$$\begin{aligned}\frac{1}{2}\frac{d}{d\tau}\|\mathbf{Q}\mathbf{x}_\ell(\tau)\|_2^2 = &\left\langle \mathbf{Q}\mathbf{x}_\ell(\tau), \mathbf{Q}\big(\mathbf{R}_\theta(\tau, \mathbf{X}(\tau))\big)_\ell \right\rangle \\ &- \lambda(\tau)\left\langle \mathbf{Q}\mathbf{x}_\ell(\tau), \mathbf{Q}\mathbf{Q}^\top \mathbf{Q}\mathbf{x}_\ell(\tau)\right\rangle.\end{aligned}$$

The key point is that the damping term controls the incoherent component since $\mathbf{Q}^2 = \mathbf{Q}$, we have:

$$\|\mathbf{Q}\mathbf{x}_\ell(\tau)\|_2^2 = \left\langle \mathbf{Q}\mathbf{x}_\ell(\tau), \mathbf{Q}^2\mathbf{x}_\ell(\tau)\right\rangle = \left\langle \mathbf{Q}^\top \mathbf{Q}\mathbf{x}_\ell(\tau), \mathbf{Q}\mathbf{x}_\ell(\tau)\right\rangle.$$

By Cauchy–Schwarz,

$$\|\mathbf{Q}\mathbf{x}_\ell(\tau)\|_2 \le \|\mathbf{Q}^\top \mathbf{Q}\mathbf{x}_\ell(\tau)\|_2,$$

with the zero case being immediate. Therefore,

$$\left\langle \mathbf{Q}\mathbf{x}_\ell(\tau), \mathbf{Q}\mathbf{Q}^\top \mathbf{Q}\mathbf{x}_\ell(\tau)\right\rangle = \|\mathbf{Q}^\top \mathbf{Q}\mathbf{x}_\ell(\tau)\|_2^2 \ge \|\mathbf{Q}\mathbf{x}_\ell(\tau)\|_2^2.$$

Using this bound and applying Cauchy–Schwarz to the residual term gives

$$\frac{1}{2}\frac{d}{d\tau}\|\mathbf{Q}\mathbf{x}_\ell(\tau)\|_2^2 \le \|\mathbf{Q}\mathbf{x}_\ell(\tau)\|_2 \left\|\mathbf{Q}\big(\mathbf{R}_\theta(\tau, \mathbf{X}(\tau))\big)_\ell\right\|_2 - \lambda(\tau)\|\mathbf{Q}\mathbf{x}_\ell(\tau)\|_2^2.$$

Since $\|\mathbf{Q}\mathbf{x}_\ell(\tau)\|_2$ is absolutely continuous, this implies

$$\frac{d}{d\tau}\|\mathbf{Q}\mathbf{x}_\ell(\tau)\|_2 \le -\lambda(\tau)\|\mathbf{Q}\mathbf{x}_\ell(\tau)\|_2 + \left\|\mathbf{Q}\big(\mathbf{R}_\theta(\tau, \mathbf{X}(\tau))\big)_\ell\right\|_2$$

almost everywhere. At points where $\|\mathbf{Q}\mathbf{x}_\ell(\tau)\|_2 > 0$, this follows by dividing the previous inequality by $\|\mathbf{Q}\mathbf{x}_\ell(\tau)\|_2$. At differentiability points where $\|\mathbf{Q}\mathbf{x}_\ell(\tau)\|_2 = 0$, the derivative of this nonnegative norm is zero, so the same inequality holds.

Since $\Gamma'(\tau) = \lambda(\tau)$ almost everywhere, the preceding inequality gives

$$\frac{d}{d\tau}\left(e^{\Gamma(\tau)}\|\mathbf{Q}\mathbf{x}_\ell(\tau)\|_2\right) \le e^{\Gamma(\tau)}\left\|\mathbf{Q}\big(\mathbf{R}_\theta(\tau, \mathbf{X}(\tau))\big)_\ell\right\|_2.$$

Integrating from 0 to τ, and then multiplying by $e^{-\Gamma(\tau)}$, yields

$$\|\mathbf{Q}\mathbf{x}_\ell(\tau)\|_2 \le e^{-\Gamma(\tau)}\|\mathbf{Q}\mathbf{x}_\ell(0)\|_2 + \int_0^\tau e^{-(\Gamma(\tau)-\Gamma(s))}\left\|\mathbf{Q}\big(\mathbf{R}_\theta(s, \mathbf{X}(s))\big)_\ell\right\|_2 ds. \tag{19}$$

Equivalently, since

$$\Gamma(\tau) - \Gamma(s) = \int_s^\tau \lambda(u)\, du,$$

Eq. (19) can be written as

$$\|\mathbf{Q}\mathbf{x}_\ell(\tau)\|_2 \le e^{-\int_0^\tau \lambda(u)\, du}\|\mathbf{Q}\mathbf{x}_\ell(0)\|_2 + \int_0^\tau e^{-\int_s^\tau \lambda(u)\, du}\left\|\mathbf{Q}\big(\mathbf{R}_\theta(s, \mathbf{X}(s))\big)_\ell\right\|_2 ds.$$

The same argument gives the block-level version over the whole forecast horizon. Define

$$\mathbf{X}_{\text{future}}(\tau) := \big(\mathbf{X}(\tau)\big)_{T+1:L}.$$

Because $\mathbf{Q}$ acts independently on each column, the Frobenius-norm version is

$$\|\mathbf{Q}\mathbf{X}_{\text{future}}(\tau)\|_F \le e^{-\Gamma(\tau)}\|\mathbf{Q}\mathbf{X}_{\text{future}}(0)\|_F + \int_0^\tau e^{-(\Gamma(\tau)-\Gamma(s))}\,\Big\|\mathbf{Q}\big(\mathbf{R}_\theta(s,\mathbf{X}(s))\big)_{T+1:L}\Big\|_F\,ds.$$

Equivalently,

$$\|\mathbf{Q}\mathbf{X}_{\text{future}}(\tau)\|_F \le e^{-\int_0^\tau \lambda(u)\,du}\|\mathbf{Q}\mathbf{X}_{\text{future}}(0)\|_F + \int_0^\tau e^{-\int_s^\tau \lambda(u)\,du}\,\Big\|\mathbf{Q}\big(\mathbf{R}_\theta(s,\mathbf{X}(s))\big)_{T+1:L}\Big\|_F\,ds.$$

For completeness, on observed past slices $\ell \le T$, the past-consistency term does not vanish in general. The corresponding projected evolution is

$$\frac{d}{d\tau}\big(\mathbf{Q}\mathbf{x}_\ell(\tau)\big) = \mathbf{Q}\big(\mathbf{R}_\theta(\tau,\mathbf{X}(\tau))\big)_\ell - \lambda(\tau)\mathbf{Q}\mathbf{Q}^\top\mathbf{Q}\mathbf{x}_\ell(\tau) - w_{\text{past}}\mathbf{Q}\big(\nabla_{\mathbf{X}}\Phi_{\text{past}}(\mathbf{X}(\tau))\big)_\ell.$$

Hence Eq. (19) should be read as a forecast-horizon bound, or more generally as a bound on any slice where the past-consistency gradient is zero.

Finally, consider the projected residual variant

$$\mathbf{v}_\theta^{\text{proj}}(\tau,\mathbf{X}(\tau)) = \mathbf{P}\mathbf{R}_\theta(\tau,\mathbf{X}(\tau)) - w_{\text{past}}\nabla_{\mathbf{X}}\Phi_{\text{past}}(\mathbf{X}(\tau)).$$

For every future slice $\ell > T$, the past-consistency gradient vanishes, and therefore

$$\mathbf{Q}\frac{d\mathbf{x}_\ell(\tau)}{d\tau} = \mathbf{Q}\mathbf{P}\big(\mathbf{R}_\theta(\tau,\mathbf{X}(\tau))\big)_\ell = (\mathbf{I}-\mathbf{P})\mathbf{P}\big(\mathbf{R}_\theta(\tau,\mathbf{X}(\tau))\big)_\ell = (\mathbf{P}-\mathbf{P}^2)\big(\mathbf{R}_\theta(\tau,\mathbf{X}(\tau))\big)_\ell = \mathbf{0}.$$

Thus

$$\frac{d}{d\tau}\big(\mathbf{Q}\mathbf{x}_\ell(\tau)\big) = \mathbf{0}.$$

Since the future initialization uses

$$\mathbf{x}_\ell(0) = \mathbf{P}\boldsymbol{\epsilon}_\ell,$$

we have

$$\mathbf{Q}\mathbf{x}_\ell(0) = \mathbf{Q}\mathbf{P}\boldsymbol{\epsilon}_\ell = \mathbf{0}.$$

Therefore,

$$\mathbf{Q}\mathbf{x}_\ell(\tau) = \mathbf{0} \qquad \forall\tau\in[0,1],$$

for every future slice $\ell > T$. Hence the projected residual variant guarantees exact forecast-horizon coherence by construction for any idempotent reconciliation operator $\mathbf{P}$, including BU and OLS.

C Continuous Unrolled Methods

This appendix formalizes the discrete-to-continuous intuition used in the main text; related viewpoints appear in prior work on unrolled optimization and plug-and-play reconstruction (Hammernik et al., 2018; Aggarwal et al., 2019).

We start from the update

$$\mathbf{Z}^k = \mathbf{X}^k - \eta\,\nabla_{\mathbf{X}}\Phi(\mathbf{X}^k), \qquad \mathbf{X}^{k+1} = \mathbf{D}_\theta(\tau_k,\mathbf{Z}^k), \tag{20}$$

where $\eta > 0$ is a step size, $\tau_k := k\eta$ is the associated time grid, and $\Phi : \mathbb{R}^{D\times L} \to \mathbb{R}$ is a differentiable consistency penalty (data consistency in inverse-problem terminology). The states $\mathbf{X}^k, \mathbf{Z}^k \in \mathbb{R}^{D\times L}$ are trajectory matrices, and $\mathbf{D}_\theta : [0,1]\times\mathbb{R}^{D\times L} \to \mathbb{R}^{D\times L}$ is a learned refinement map. Allowing $\mathbf{D}_\theta$ to depend on flow time τ_k matches common unrolled networks where parameters may vary by layer, and it also matches the time-dependent vector fields used in flow models.

One formal route from discrete unrolled updates to a continuous-time ODE is to assume a near-identity, step-size-scaled refinement map,

$$\mathbf{D}_\theta(\tau,\mathbf{Z}) = \mathbf{Z} + \eta\,\mathbf{R}_\theta(\tau,\mathbf{Z}) + o(\eta). \tag{21}$$

Here $\mathbf{R}_\theta : [0,1] \times \mathbb{R}^{D\times L} \to \mathbb{R}^{D\times L}$ is a vector field bounded on compact sets, and the remainder $o(\eta)$ is uniform on compact sets of $(\tau, \mathbf{Z})$ as $\eta \to 0$. Under this scaling assumption, the discrete update yields the continuous-time residual-plus-consistency dynamics.

Substituting Eq. (21) into Eq. (20) gives

$$\begin{aligned}\mathbf{X}^{k+1} &= \mathbf{Z}^k + \eta\,\mathbf{R}_\theta(\tau_k, \mathbf{Z}^k) + o(\eta)\\ &= \mathbf{X}^k - \eta\nabla_{\mathbf{X}}\Phi(\mathbf{X}^k) + \eta\,\mathbf{R}_\theta\big(\tau_k, \mathbf{X}^k - \eta\nabla_{\mathbf{X}}\Phi(\mathbf{X}^k)\big) + o(\eta).\end{aligned} \tag{22}$$

If $\mathbf{R}_\theta(\tau,\cdot)$ is continuous, then $\mathbf{R}_\theta(\tau_k, \mathbf{X}^k - \eta\nabla_{\mathbf{X}}\Phi(\mathbf{X}^k)) = \mathbf{R}_\theta(\tau_k, \mathbf{X}^k) + o(1)$ as $\eta \to 0$, because the argument shifts by $\mathcal{O}(\eta)$ while $\nabla_{\mathbf{X}}\Phi(\mathbf{X}^k)$ remains finite on bounded sets. Plugging this into Eq. (22) yields the first-order increment form

$$\mathbf{X}^{k+1} = \mathbf{X}^k + \eta\Big(\mathbf{R}_\theta(\tau_k, \mathbf{X}^k) - \nabla_{\mathbf{X}}\Phi(\mathbf{X}^k)\Big) + o(\eta). \tag{23}$$

Equation (23) already has the structure of a time-discretized ODE. Rearranging,

$$\frac{\mathbf{X}^{k+1} - \mathbf{X}^k}{\eta} = \mathbf{R}_\theta(\tau_k, \mathbf{X}^k) - \nabla_{\mathbf{X}}\Phi(\mathbf{X}^k) + o(1). \tag{24}$$

If the iterates sample a smooth curve $\mathbf{X}(\tau)$ in the limit $\eta \to 0$ (for instance, take the piecewise-linear interpolation of the points $\{(\tau_k, \mathbf{X}^k)\}$ and assume it converges to a differentiable limit), then the left-hand side of Eq. (24) is exactly the forward difference quotient for the derivative. Taking $\eta \to 0$ therefore yields the continuous-time dynamics

$$\frac{d\mathbf{X}(\tau)}{d\tau} = \mathbf{R}_\theta(\tau, \mathbf{X}(\tau)) - \nabla_{\mathbf{X}}\Phi(\mathbf{X}(\tau)). \tag{25}$$

This is the continuous-time plug-and-play form used in Section 4: the gradient term is the model-based consistency force and the learned term is a residual correction.

D Implementation Details

This appendix describes the neural architectures used to parameterize the learnable residual field R_θ in our Hierarchical Conditional Flow Matching model.

D.1 Notation and shapes

Let T denote the observed context length and L the total trajectory length (context plus forecast). We use two parameterizations of the residual field $\mathbf{R}_\theta$.

For the 1D temporal-CNN variant, computations are performed independently per series. In code, batch and node dimensions are flattened into a single index $BD = B \cdot D$, so the network input and output have shapes

$$\mathbf{X}(\boldsymbol{\tau}) \in \mathbb{R}^{BD\times L}, \qquad \boldsymbol{\tau} \in \mathbb{R}^{BD}, \qquad \mathbf{R}_\theta(\boldsymbol{\tau}, \mathbf{X}(\boldsymbol{\tau})) \in \mathbb{R}^{BD\times L}.$$

For the 2D U-Net variant, the network acts jointly on the full node-time state. The corresponding tensors have shapes

$$\mathbf{X}(\boldsymbol{\tau}) \in \mathbb{R}^{B\times D\times L}, \qquad \boldsymbol{\tau} \in \mathbb{R}^{B}, \qquad \mathbf{R}_\theta(\boldsymbol{\tau}, \mathbf{X}(\boldsymbol{\tau})) \in \mathbb{R}^{B\times D\times L},$$

up to the additional channel dimension created after concatenating time features and the observed/forecast mask.

D.2 Input Data Scaling

To ensure numerical stability and scale invariance, we normalize the trajectory $\mathbf{X}$ using a scale factor $s_{b,d}$ calculated for each series $d \in \{1,\dots,D\}$ in batch b. Following our definitions in Section 3, let $x_{b,d,\ell} = [\mathbf{X}]_{b,d,\ell}$ denote the value of the trajectory for batch b, dimension d, at time step $\ell \in \{1,\dots,L\}$, and let $[\mathbf{Y}_{\text{past}}]_{b,d,\ell}$ be the observed value at time $\ell \in \{1,\dots,T\}$ with $[\mathbf{M}_{\text{obs}}]_{d,\ell}$

the binary observation mask. We define the local scale $\nu_{b,d}$ as the average absolute value over the observed past context:

$$\nu_{b,d} = \frac{\sum_{\ell=1}^{T} |[\widetilde{\mathbf{Y}}_{\text{past}}]_{b,d,\ell}| \cdot [\mathbf{M}_{\text{obs}}]_{d,\ell}}{\max(1, \sum_{\ell=1}^{T} [\mathbf{M}_{\text{obs}}]_{d,\ell})}. \tag{26}$$

If a series is entirely unobserved or consists of zeros within the context window, we apply a fallback scale $\bar{\nu}_d$ computed as the average scale for that dimension across the entire batch. The final scale $s_{b,d}$ is given by $\max(s_{\min}, \nu_{b,d})$ if the series has a non-zero observed sum and $\max(s_{\min}, \bar{\nu}_d)$ otherwise, where $s_{\min} = 10^{-10}$ prevents division by zero. Collecting the scales of instance b in $\mathbf{\Lambda}_b := \operatorname{diag}(s_{b,1}, \ldots, s_{b,D})$, the normalized trajectory used for generation is $\mathbf{\Lambda}_b^{-1}\mathbf{X}_b$, where $\mathbf{X}_b \in \mathbb{R}^{D\times L}$ is the trajectory of instance b; element-wise,

$$\left[\mathbf{\Lambda}_b^{-1}\mathbf{X}_b\right]_{d,\ell} = \frac{x_{b,d,\ell}}{s_{b,d}} \quad \text{for all } \ell \in \{1, \ldots, L\}. \tag{27}$$

The main text and Appendix D.4 drop the instance index and write $\mathbf{\Lambda}$.

D.3 Output Data Scaling

Let $\mathbf{X}_b(1)$ denote the ODE endpoint of instance b, which is in normalized coordinates. The forecast in original units is the future block of $\mathbf{\Lambda}_b\mathbf{X}_b(1)$:

$$\left[\mathbf{Y}_{\text{future}}\right]_{b,d,\eta} = s_{b,d}\left[\mathbf{X}_b(1)\right]_{d,T+\eta}, \qquad \eta \in \{1, \ldots, H\}, \tag{28}$$

where $s_{b,d}$ is the scale factor defined above. The soft-guided variant then applies $\mathbf{P}$ in these original units (Appendix D.4).

D.4 Coherence projectors used in the experiments

Projectors. Let N be the number of bottom-level series and $\mathbf{S} \in \{0,1\}^{D\times N}$ the summation matrix, so that $\mathcal{C} = \operatorname{range}(\mathbf{S})$. The OLS projector $\mathbf{P}_{\text{OLS}} = \mathbf{S}(\mathbf{S}^\top\mathbf{S})^{-1}\mathbf{S}^\top$ is orthogonal. The bottom-up projector $\mathbf{P}_{\text{BU}} = \mathbf{SE}$, where $\mathbf{E} \in \{0,1\}^{N\times D}$ selects the bottom-level series ($\mathbf{ES} = \mathbf{I}_N$), is oblique but idempotent, $\mathbf{P}_{\text{BU}}^2 = \mathbf{SESE} = \mathbf{SE}$, with the same range $\mathcal{C}$. Our experiments use $\mathbf{P} = \mathbf{P}_{\text{BU}}$.

Normalized coordinates. The flow is integrated on the normalized state $\mathbf{\Lambda}^{-1}\mathbf{X}$, where $\mathbf{\Lambda} = \operatorname{diag}(s_1, \ldots, s_D)$ holds the per-series scales of Appendix D.2, computed per instance from the observed context only. Coherence is defined in original units ($\mathbf{QX} = \mathbf{0}$), so the coherent set in normalized coordinates is $\mathbf{\Lambda}^{-1}\mathcal{C}$, and we use

$$\mathbf{P}_{\mathbf{\Lambda}} := \mathbf{\Lambda}^{-1}\mathbf{P}\mathbf{\Lambda}, \qquad \mathbf{Q}_{\mathbf{\Lambda}} := \mathbf{I} - \mathbf{P}_{\mathbf{\Lambda}} = \mathbf{\Lambda}^{-1}\mathbf{Q}\mathbf{\Lambda}. \tag{29}$$

These satisfy (i) $\mathbf{P}_{\mathbf{\Lambda}}^2 = \mathbf{\Lambda}^{-1}\mathbf{P}^2\mathbf{\Lambda} = \mathbf{P}_{\mathbf{\Lambda}}$; (ii) $\operatorname{range}(\mathbf{P}_{\mathbf{\Lambda}}) = \mathbf{\Lambda}^{-1}\mathcal{C}$; and (iii) $\mathbf{Q}_{\mathbf{\Lambda}}(\mathbf{\Lambda}^{-1}\mathbf{X}) = \mathbf{\Lambda}^{-1}\mathbf{QX}$, hence $\mathbf{Q}_{\mathbf{\Lambda}}(\mathbf{\Lambda}^{-1}\mathbf{X}) = \mathbf{0} \iff \mathbf{QX} = \mathbf{0}$. Let $\mathcal{B}(i)$ denote the bottom-level descendants of node i ($\mathcal{B}(i) = \{i\}$ for a leaf). Applying the unconjugated $\mathbf{P}_{\text{BU}}$ to $\mathbf{\Lambda}^{-1}\mathbf{X}$ would impose $x_i/s_i = \sum_{j\in\mathcal{B}(i)} x_j/s_j$ on each column, which is not the aggregation constraint unless all scales within a subtree coincide. The conjugated projector instead gives

$$(\mathbf{P}_{\mathbf{\Lambda}}\mathbf{z})_i = \frac{1}{s_i}\sum_{j\in\mathcal{B}(i)} s_j z_j, \tag{30}$$

so it preserves the leaves and aggregates them with scale-ratio weights. In practice $\mathbf{P}_{\mathbf{\Lambda}}\mathbf{z} = \mathbf{\Lambda}^{-1}\mathbf{P}(\mathbf{\Lambda}\mathbf{z})$, so no per-instance $D \times D$ matrix is formed.

Use in HCFM. $\mathbf{P}_{\mathbf{\Lambda}}$ and $\mathbf{Q}_{\mathbf{\Lambda}}$ enter the coherence penalty (Eq. (2)), the projected residual (Eq. (5)), the initialization (Eq. (6)), the tangency penalty (Eq. (8)) and the coherence error (Eq. (34)). The penalty $\frac{1}{2}\|\mathbf{Q}_{\mathbf{\Lambda}}\mathbf{x}_\ell\|_2^2 = \frac{1}{2}\|\mathbf{\Lambda}^{-1}\mathbf{Q}\mathbf{\Lambda}\mathbf{x}_\ell\|_2^2$ measures each node's incoherence relative to its own scale, so the guidance weights do not depend on the magnitude of the data. The proofs in Appendices A and B use only idempotence of $\mathbf{P}$ and hold verbatim for $\mathbf{P}_{\mathbf{\Lambda}}$; in particular, $\mathbf{Q}_{\mathbf{\Lambda}}\mathbf{x}_\ell(\tau) = \mathbf{0}$ implies exact coherence of $\mathbf{\Lambda}\mathbf{x}_\ell(\tau)$ in original units. When the data are nonnegative and coherent and no fallback scale is active, the mean-absolute scales are additive, $s_i = \sum_{j\in\mathcal{B}(i)} s_j$; each aggregate of $\mathbf{P}_{\mathbf{\Lambda}}\boldsymbol{\epsilon}$ is then a convex combination of leaf noises with variance $\sum_{j\in\mathcal{B}(i)}(s_j/s_i)^2 \le 1$, matching the unit scale of the normalized data. After integration, the future block is mapped back to original units by $\mathbf{\Lambda}$, where the soft-guided variant applies $\mathbf{P}$.

D.5 COHERENCE-GUIDANCE WEIGHT SCHEDULE

In the main text, $\lambda(\tau) \geq 0$ denotes the coherence-guidance weight used in the soft-guided vector field. In our implementation, $\lambda(\tau)$ is piecewise constant over $J = 8$ uniform time bins on $\tau \in [0, 1]$. For the j-th bin, where $j \in \{1, \ldots, J\}$, we set

$$\lambda_j = \mathrm{softplus}(\rho_j) + \lambda_{\mathrm{base}},$$

where $\rho_j \in \mathbb{R}$ is an unconstrained trainable scalar and $\lambda_{\mathrm{base}} = 0.1$ is fixed.

This parameterization guarantees a strictly positive guidance weight:

$$\lambda_j \geq \lambda_{\mathrm{base}} > 0.$$

Thus the coherence-guidance term cannot be switched off completely during training or sampling. We initialize all schedule parameters as $\rho_j = 0$, which gives $\lambda_j \approx 0.793$ at initialization.

The purpose of learning the schedule is to adapt the strength of the coherence pull when the training data contain small aggregation inconsistencies or noise. The coherence-guidance part of the vector field is

$$-\lambda(\tau)\nabla_{\mathbf{X}}\Phi_{\mathrm{coh}}(\mathbf{X}(\tau)) = -\lambda(\tau)\mathbf{Q_\Lambda}^\top\mathbf{Q_\Lambda}\mathbf{X}(\tau).$$

Therefore, the direct learning signal for ρ_j through this term is present only when the coherence residual $\mathbf{Q_\Lambda}\mathbf{X}(\tau)$ is nonzero. If the training interpolants are exactly coherent, this term vanishes and the schedule receives little or no gradient from the coherence-guidance component; in that case, the positive initialization and the lower bound keep the guidance active. When the data are only approximately coherent, the nonzero residual $\mathbf{Q_\Lambda}\mathbf{X}(\tau)$ provides a signal for adapting the bin weights.

The theoretical coherence-guidance bound in Appendix B is unaffected by whether the schedule is fixed manually or learned during training. The proof only requires that $\lambda(\tau)$ is nonnegative during the ODE solve, which is guaranteed by the softplus parameterization and the positive base term.

D.6 HYPERPARAMETER SETTINGS

To make the experimental configuration explicit, we summarize the key guidance and regularization hyperparameters here. We set the base α and β to 0.1, as this already provided good results in the soft-guided variant, set $w_{\mathrm{past}} = 20$, and use $\lambda_{\mathrm{adv}} = 0.1$. For sampling we use Heun's method with 20 ODE steps. These loss weights were selected heuristically to balance objective magnitudes and shared across all six benchmarks.

Quantity	Value
α (coherence strength)	0.1
β (consistency strength)	0.1
w_{past}	20
λ_{adv}	0.1
$\lambda(t)$ schedule	$J = 8$ bins; $\lambda_j = \mathrm{softplus}(\rho_j) + \lambda_{\mathrm{base}}$, $\lambda_{\mathrm{base}} = 0.1$
ODE solver	Heun
Default ODE steps (sampling)	20

D.7 RESIDUAL NETWORK ARCHITECTURES

In all experiments, $\mathbf{R}_\theta$ is conditioned on the current state and the flow time, $(\mathbf{X}(\tau), \tau)$. This isolates the role of the neural architecture to the parameterization of the residual field itself, while conditioning on the observed past and enforcement of hierarchical coherence enter through the explicit terms in the overall vector field. We study two parameterizations of $\mathbf{R}_\theta$.

1D TEMPORAL CNN (HCFM-MQCNN)

Figure 3 summarizes the 1D architecture. The network has two main components: a time embedding block and a non-causal dilated temporal convolutional network operating over the length-L axis. This parameterization computes independently per series. As a result, any cross-series coupling arises from the explicit coherence and past consistency dynamics rather than from learned mixing inside the residual field.

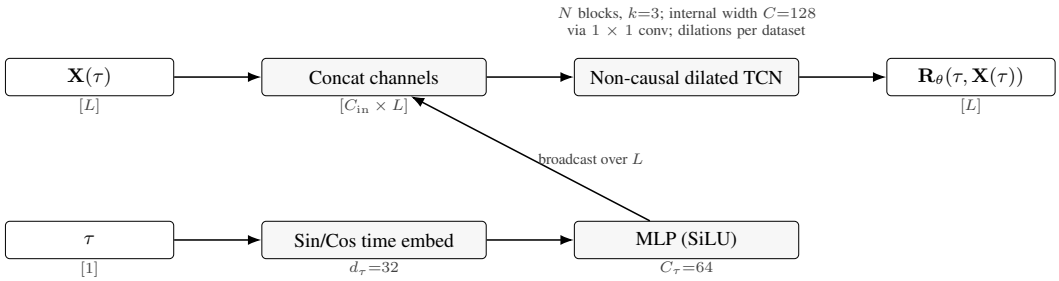


Figure 3: 1D residual network used in HCFM-MQCNN. The model takes the current series trajectory $\mathbf{X}(\tau) \in \mathbb{R}^L$ and the flow time $\tau \in [0, 1]$, embeds τ with a sinusoidal feature map followed by an MLP, broadcasts the time features across the length-L axis, concatenates them with $\mathbf{X}(\tau)$ as channels, and predicts a residual correction over L with a non-causal dilated TCN.

TIME EMBEDDING

For each scalar flow time $\tau \in [0, 1]$ we compute a sinusoidal time embedding of dimension $d_\tau = 32$, followed by a two-layer MLP with SiLU activations that maps to a hidden size of $C_\tau = 64$. The resulting vector in $\mathbb{R}^{C_\tau}$ is broadcast across the L steps and treated as additional input channels.

NON-CAUSAL DILATED TCN OVER LENGTH L

Let $C_{\text{in}} = 1 + C_\tau$ denote the total number of input channels, consisting of the state channel $\mathbf{X}(\tau)$ and the broadcast time channels. We map these channels to an internal width $C = 128$ using a 1×1 convolution. We then apply a stack of N non-causal temporal blocks with residual connections. Each block contains two 1D convolutions with kernel size $k = 3$, ReLU activations, dropout, and a residual skip. The convolutions use symmetric padding ("same" padding) so that the length L is preserved. A final 1×1 convolution maps back to a single output channel, producing the residual correction $\mathbf{R}_\theta(\tau, \mathbf{X}(\tau)) \in \mathbb{R}^L$.

2D U-NET OVER THE $D \times L$ STATE (HCFM-UNET)

Figure 4 summarizes the 2D U-Net parameterization used in HCFM-UNet. The second parameterization acts directly on the full node–time state and therefore allows the learned residual field itself to mix information across series and time. We stack the current state $\mathbf{X}(\tau)$, a binary observed/forecast mask, and broadcast time features as input channels and process the resulting tensor with a 2D U-Net on the (D, L) grid. A single 2D convolution can be viewed as a shared local interaction rule: nearby entries interact according to relative offset, and stacking such layers expands the interaction radius. The encoder–decoder structure makes these interactions multiscale. In our implementation, downsampling is applied only along the time axis, so coarse layers gain a larger temporal receptive field without collapsing the node axis. Skip connections then combine coarse global context with fine local detail in the decoder. The output is mapped back to $\mathbb{R}^{D \times L}$ to produce the residual correction $\mathbf{R}_\theta(\tau, \mathbf{X}(\tau))$.

For HCFM-MQCNN, we use the dataset-specific MQCNN configurations from CLOVER (Olivares et al., 2024c), but remove the cross-series module. The model therefore keeps the same per-series temporal convolutional and decoder configuration as CLOVER, while cross-series interactions are handled by the explicit hierarchy operators and coherence regularizers in HCFM.

For HCFM-UNet, we use a modified DDPM++-style (Song et al., 2021) U-Net that downsamples only along the time dimension and keeps the hierarchy-node dimension at fixed resolution. The channel multiplier schedule is $[1, 2, 4]$ for all datasets. We use 8 base channels with dropout 0.2 for Labour and Tourism-S, and 16 base channels with dropout 0.1 for `Traffic`, `Tourism-L`, `Wiki`, and `Favorita`. These settings were chosen using the validation set in each respective dataset.

$$\mathbf{R}_\theta(\tau, \mathbf{X}(\tau), \mathbf{M}) = \text{U-Net}(\mathbf{X}(\tau), \mathbf{M}, \tau)$$
$$\frac{d\mathbf{X}(\tau)}{d\tau} = \mathbf{R}_\theta(\tau, \mathbf{X}(\tau), \mathbf{M}) - \lambda(\tau)\nabla_{\mathbf{X}}\Phi_{\text{coh}}(\mathbf{X}(\tau)) - w_{\text{past}}\nabla_{\mathbf{X}}\Phi_{\text{past}}(\mathbf{X}(\tau))$$

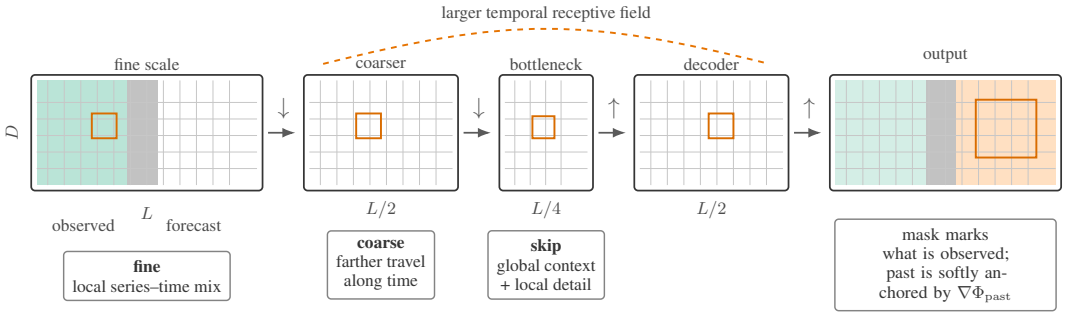


Figure 4: 2D U-Net parameterization of the residual field R_θ used in HCFM-UNet. The U-Net operates on the node–time grid $(D \times L)$ with multiscale temporal downsampling (along L only), so deeper layers have a larger temporal receptive field; skip connections merge coarse global context with fine local structure. The observed/forecast mask indicates which part of the trajectory is given, while past anchoring and coherence enforcement are handled by the explicit gradient terms in the overall vector field.

D.8 Computational Resources

All experiments were run on a machine with a second-generation AMD EPYC 7R32 processor, 64 vCPUs, 256 GiB of RAM, and one NVIDIA A10G GPU with 22 GiB of GPU memory. The environment was standardized using Python 3.12, with PyTorch serving as the foundation for all core architectural and dynamical system implementations.

To ensure complete experimental reproducibility, we list all Python libraries used and their versions in Table 3.

Table 3: Software dependencies within the experimental pipeline.

Package	**Version**
`torch` (Ansel et al., 2024)	`2.10.0+cu128`
`pytorch-lightning` (Falcon & The PyTorch Lightning team, 2019)	`2.6.1`
`hydra-core` (Yadan, 2019)	`1.3.2`
`omegaconf` (Yadan et al., 2019)	`2.3.0`
`numpy` (Harris et al., 2020)	`2.0.2`
`pandas` (The pandas development team, 2010)	`2.2.2`
`scipy` (Virtanen et al., 2020)	`1.15.3`
`tqdm` (da Costa-Luis et al., 2026)	`4.67.3`
`pyyaml` (Simonov et al., 2024)	`6.0.3`
`psutil` (Rodolà, 2024)	`5.9.5`
`tensorboard` (Abadi et al., 2015)	`2.19.0`
`hierarchicalforecast` (Olivares et al., 2024a)	`1.4.0`
`datasetsforecast` (Nixtla, 2023)	`1.0.1.dev0`

E Dataset Details

The `Labour dataset` (Australian Bureau of Statistics, 2019) tracks monthly Australian employment from February 1978 to December 2019, reporting employee counts by employment status, gender, and geography. It forms a 4-level hierarchy with $D = 57$ total series, consisting of 25 aggregate series and 32 bottom-level series. The forecast horizon is set to $H = 12$ months.

The `Traffic` dataset (Ben Taieb & Koo, 2019) records daily freeway occupancy rates aggregated from hourly measurements across 200 car lanes in the San Francisco Bay Area from January 2008 to March 2009. The lanes are structured into four sets of 50, two sets of 100, and one root total of 200 lanes, yielding 4 levels and $D = 207$ total series with 200 bottom-level series. The forecast horizon is $H = 1$ day. Exogenous features include geographic node dummies, weekend indicators, and proximity to Saturday.

The `Tourism-S` dataset (Tourism Australia, Canberra, 2005) contains quarterly visits to Australia spanning 1998 to 2006. The structure comprises 4 aggregation levels totaling $D = 89$ series, with 56 bottom-level regions partitioned by urbanization and 33 aggregate series across country, purpose of travel, state, and region. The forecast horizon is $H = 4$ quarters. Exogenous inputs include state dummies, quarterly indicators, and seasonal naive anchors at lags 4 and 8.

The `Tourism-L` dataset (Wickramasuriya et al., 2019) tracks monthly visits to Australia from January 1998 to December 2016 across 76 regions grouped by zone, state, and national totals, crossed with four travel purposes. This hierarchy contains $4/5$ levels and $D = 555$ total series, with 304 bottom-level leaf entities. The forecast horizon is $H = 12$ months. Exogenous features include purpose of travel, state dummies, month indicators, and a seasonal naive anchor forecast.

The `Wiki` dataset (Anava et al., 2018) contains daily page views from January 2016 to December 2016 across categories of country, access method, and agent type. The hierarchy spans 5 levels with $D = 199$ total series, partitioned into 150 bottom-level series and 49 aggregate series. The forecast horizon is $H = 7$ days. Exogenous features include day-of-week indicators, static categories for country, access, and agent, and a seasonal naive anchor at lag 7.

The `Favorita` dataset (Corporación Favorita et al., 2017) records daily grocery sales from January 2013 to August 2017 across 54 stores, aggregated across store, city, state, and national tiers into 4 levels. We evaluate $4{,}036$ independent item hierarchies, each with $D = 93$ series and 54 bottom-level series, giving $375{,}348$ time series in total. The forecast horizon is $H = 34$ days. Exogenous features include item perishability flags, state indicators, promotional events, and day-of-the-week dummies.

Across all datasets, we adopt the same data processing and feature engineering protocols as in Olivares et al. (2024c).

F Ablation Studies

The following ablations isolate the main design choices in HCFM while keeping the remaining training, architecture, and model-selection settings fixed. We study the effect of soft versus hard coherence enforcement, the ODE solver and step count used at inference time, and the auxiliary losses used to shape the learned trajectory distribution. We omit `Favorita` from the ablation studies because of its substantially larger number of series and the resulting computational cost.

F.1 Soft versus Hard Coherence Enforcement

This ablation compares the default soft-guided field in Eq. (31) to a projected residual variant in Eq. (33). We follow the notation in Section 4: $\mathbf{P}_{\mathbf{\Lambda}}$ and $\mathbf{Q}_{\mathbf{\Lambda}}$ are the normalized-coordinate projectors of Appendix D.4, and $\mathbf{X}(\tau) \in \mathbb{R}^{D \times L}$ denotes the trajectory state at flow time τ.

Our default vector field is

$$\mathbf{v}_\theta(\tau, \mathbf{X}(\tau)) = \mathbf{R}_\theta(\tau, \mathbf{X}(\tau)) - \lambda(\tau)\, \nabla_{\mathbf{X}} \Phi_{\mathrm{coh}}(\mathbf{X}(\tau)) - w_{\mathrm{past}}\, \nabla_{\mathbf{X}} \Phi_{\mathrm{past}}(\mathbf{X}(\tau)), \tag{31}$$

with the tangency penalty

$$\mathcal{L}_{\tan} = \mathbb{E} \left\| \mathbf{Q}_{\mathbf{\Lambda}} \mathbf{R}_\theta(\tau, \mathbf{X}(\tau)) \right\|_F^2. \tag{32}$$

We compare this to a hard-projection variant where the learned residual is projected before integration and we drop the coherence guidance term and the tangency penalty,

$$\mathbf{v}_\theta^{\mathrm{proj}}(\tau, \mathbf{X}(\tau)) = \mathbf{P}_{\mathbf{\Lambda}} \mathbf{R}_\theta(\tau, \mathbf{X}(\tau)) - w_{\mathrm{past}} \nabla_{\mathbf{X}} \Phi_{\mathrm{past}}(\mathbf{X}(\tau)). \tag{33}$$

Since $\mathbf{Q}_{\mathbf{\Lambda}} \mathbf{P}_{\mathbf{\Lambda}} = \mathbf{0}$, the projected residual is tangent to the coherent set by construction, so coherence is exact up to numerical ODE solver tolerance.

Table 4: Soft-guided versus projected-residual coherence mechanisms on overall sCRPS. Each cell reports mean sCRPS ± standard deviation across 3 windows. Lower is better.

Method	Dataset	Soft-guided	Projected residual
HCFM-MQCNN	Traffic	0.0936 ± 0.0142	$\mathbf{0.0831 \pm 0.0228}$
	Tourism-S	$\mathbf{0.0752 \pm 0.0183}$	0.0765 ± 0.0173
	Tourism-L	$\mathbf{0.1466 \pm 0.0165}$	0.1482 ± 0.0154
	Labour	0.0096 ± 0.0013	$\mathbf{0.0095 \pm 0.0016}$
	Wiki	$\mathbf{0.1907 \pm 0.0997}$	0.1922 ± 0.0891
HCFM-UNet	Traffic	0.0245 ± 0.0122	$\mathbf{0.0209 \pm 0.0049}$
	Tourism-S	$\mathbf{0.0808 \pm 0.0117}$	0.0891 ± 0.0159
	Tourism-L	0.1366 ± 0.0069	$\mathbf{0.1339 \pm 0.0127}$
	Labour	$\mathbf{0.0093 \pm 0.0003}$	0.0112 ± 0.0009
	Wiki	$\mathbf{0.1838 \pm 0.0981}$	0.1850 ± 0.1046

Table 5: Future-horizon coherence violation CE (Eq. (34); HCFM-UNet), computed on the raw ODE endpoint before reconciliation. Final reconciled forecasts are exactly coherent by construction.

Dataset	Soft-guided (mean)	Projected residual
Labour	2.981×10^{-3}	≈ 0
Traffic	1.291×10^{-3}	≈ 0
Tourism-S	1.240×10^{-2}	≈ 0
Tourism-L	1.109×10^{-2}	≈ 0
Wiki	5.259×10^{-3}	≈ 0

All other components (encoder, remaining loss terms, and hyperparameters) are unchanged. The default soft-guided field enforces coherence continuously via guidance and a tangency penalty, while the projected-residual variant achieves coherence through the vector-field parameterization itself (up to numerical solver tolerance). We report the same hierarchical metrics as in the main experiments to test whether soft tangency improves forecast quality compared to hard projection.

The comparison shows a genuine tradeoff: hard projection guarantees exact forecast-horizon coherence by construction, while the soft-guided field can be slightly better on some datasets. Neither mechanism uniformly dominates, which supports treating them as alternative operating points rather than as strict substitutes.

We also report future-horizon coherence violation. The default soft-guided model remains close to coherent but is not exactly on the coherent subspace, whereas the projected-residual variant is exact up to numerical solver tolerance.

We measure mean future-horizon coherence violation using

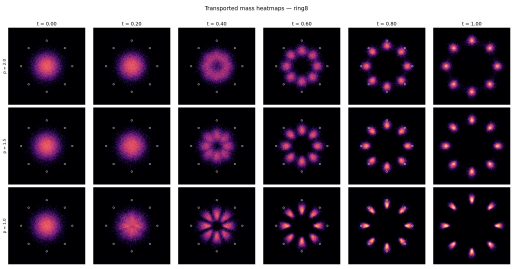
$$\mathrm{CE}(\mathbf{X}(1)) = \frac{\|\mathbf{Q}_{\mathbf{\Lambda}}\mathbf{X}_{\mathrm{future}}(1)\|_F}{\sqrt{HD}} = \frac{\|\mathbf{\Lambda}^{-1}\mathbf{Q}\,\mathbf{\Lambda}\mathbf{X}_{\mathrm{future}}(1)\|_F}{\sqrt{HD}}, \tag{34}$$

where $\mathbf{X}(1)$ is the raw ODE endpoint in normalized coordinates, before unscaling and final reconciliation. CE is the original-unit incoherence of each node relative to its own scale, which makes it comparable across datasets. H and D denote the forecast horizon length and the number of series. We evaluate the mean across the three cross-validation windows with the U-Net variant.

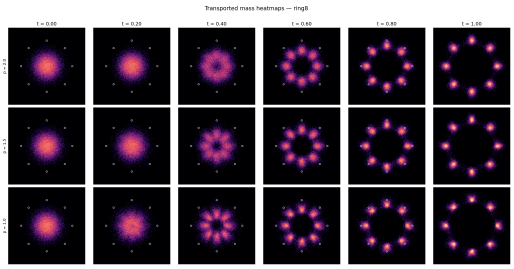
F.2 Role of the Adversarial Term

For related uses of ℓ_1 regression and adversarial training in flow models, see Song et al. (2025) and Lin et al. (2026). Our discriminator $d_\psi : \mathbb{R}^{D\times L} \to \mathbb{R}$ is a lightweight ResNet using stacked one-dimensional convolutions and LeakyReLU activations.

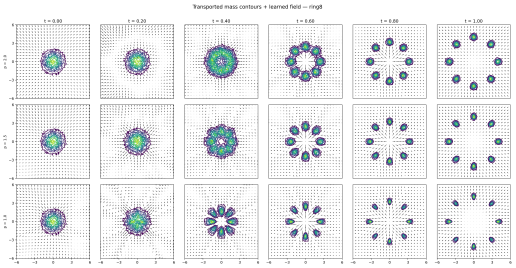
Figure 5 isolates the role of the adversarial term in a Ring-8 toy problem. We compare the learned endpoint density and vector field with and without adversarial training. The purpose of this figure is diagnostic rather than quantitative: it shows how $\mathcal{L}_{\mathrm{adv}}$ reshapes the induced endpoint distribution while leaving the base flow-matching construction unchanged. In the forecasting model, the adversarial term serves as a lightweight distributional signal on rectified endpoint estimates. It complements the base objective rather than replacing it.



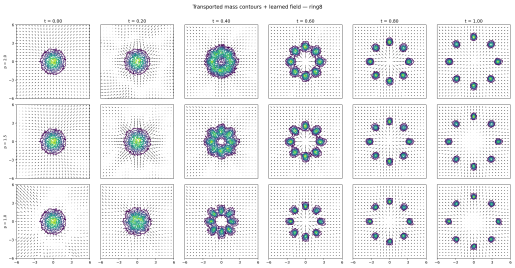
(a) Probability density flow without adversarial loss.



(b) Probability density flow with adversarial loss.



(c) Vector field without adversarial loss.



(d) Vector field with adversarial loss.

Figure 5: Ring-8 toy setting: comparison of the learned endpoint density and vector field with vs. without adversarial training.

To complement the Ring-8 diagnostic, we evaluate the impact of the adversarial term on the sharpness and calibration of HCFM-UNet's predictive distribution. We evaluate forecast sharpness using the Scaled Interval Width (SIW) and marginal interval calibration using the Centered Calibration Error (CCE) (Adler et al., 2026). For a given hierarchy level ℓ, let I_ℓ denote the set of series, H the forecast horizon, and $\widehat{Q}_{i,t}(q)$ the predicted q-quantile for series i at horizon step t. We evaluate central prediction intervals across nominal coverages $\mathcal{S} = \{0.2, 0.4, 0.6, 0.8\}$. For each coverage level $s \in \mathcal{S}$, the corresponding lower-tail probability is $a_s = \frac{1-s}{2}$. The mean forecast interval width at level ℓ is

$$\overline{W}_{\ell,s} = \frac{1}{|I_\ell| H} \sum_{i \in I_\ell} \sum_{t=1}^{H} \left(\widehat{Q}_{i,t}(1 - a_s) - \widehat{Q}_{i,t}(a_s) \right). \tag{35}$$

To make this width comparable across datasets and hierarchy levels, we normalize it using the matching empirical interval width of the observations. Let $\mathcal{Y}_\ell^{\text{ref}} = \{y_{i,\tau} : i \in I_\ell,\ \tau \in T_{\text{full}}\}$ denote the observations pooled across all series at level ℓ and all available timesteps. The empirical reference width is

$$W_{\ell,s}^{\text{ref}} = Q_{1-a_s}\left(\mathcal{Y}_\ell^{\text{ref}}\right) - Q_{a_s}\left(\mathcal{Y}_\ell^{\text{ref}}\right), \tag{36}$$

where $Q_q(\cdot)$ denotes the empirical q-quantile. We then define

$$\text{SIW}_\ell = \frac{1}{|\mathcal{S}_\ell^{\text{valid}}|} \sum_{s \in \mathcal{S}_\ell^{\text{valid}}} \frac{\overline{W}_{\ell,s}}{W_{\ell,s}^{\text{ref}}} \tag{37}$$

with $\mathcal{S}_\ell^{\text{valid}} = \{s \in \mathcal{S} : W_{\ell,s}^{\text{ref}} > 10^{-7}\}$. Non-finite values are omitted from the corresponding averages.

Let $\mathcal{V}_{\ell,s}$ denote the set of valid forecast cells where predictions and observations are finite. The observed coverage $\widehat{C}_{\ell,s}$ is the fraction of valid observations falling within the nominal s-coverage prediction interval:

$$\widehat{C}_{\ell,s} = \frac{1}{|\mathcal{V}_{\ell,s}|} \sum_{(i,t) \in \mathcal{V}_{\ell,s}} \mathbf{1}\left\{ \widehat{Q}_{i,t}(a_s) \le y_{i,t} \le \widehat{Q}_{i,t}(1 - a_s) \right\} \tag{38}$$

where $\mathbf{1}\{\cdot\}$ is the indicator function. CCE measures the mean deviation between nominal and observed coverage:

$$CCE_\ell = \frac{1}{|\mathcal{S}_\ell^{\text{valid}}|} \sum_{s \in \mathcal{S}_\ell^{\text{valid}}} \left| s - \widehat{C}_{\ell,s} \right|. \tag{39}$$

SIW < 1 indicates that predicted intervals are narrower than the matching empirical range of the data, while SIW > 1 indicates wider intervals. Among similarly calibrated forecasts, lower SIW indicates greater sharpness. CCE measures the absolute gap between nominal and empirical coverage, with zero indicating a match and larger values indicating greater miscalibration. We evaluate CCE jointly with SIW to assess calibration and sharpness. Table 6 shows that enabling the adversarial term lowers mean sCRPS on all five datasets. CCE decreases and SIW increases on `Traffic`, `Tourism-S`, `Tourism-L`, and `Wiki`. On `Labour`, CCE increases and SIW decreases. The lower CCE together with wider intervals on these four datasets is consistent with mitigating the ℓ_1-induced contraction discussed in Appendix G. CRPS aggregates quantile losses, while CCE evaluates coverage at selected nominal levels. Therefore, a localized correction can be more visible in CCE than in aggregate sCRPS. These results support the role of $\mathcal{L}_{\text{adv}}$ as a dataset-dependent auxiliary regularizer for predictive dispersion.

Table 6: Ablation of the adversarial term in HCFM-UNet. Every configuration uses the same U-Net backbone, the same data splits and the same hyperparameters, and differs only in the weight λ_{adv} placed on the adversarial loss. $\lambda_{\text{adv}} = 0.0$ disables the term entirely. Each cell reports the mean selected-test score $\pm$ standard deviation across 3 rolling chained cross-validated windows. sCRPS is the scaled continuous ranked probability score and CCE is the centered calibration error, the unsigned gap between nominal and observed coverage; lower is better for both, and for each dataset the better of the two settings is determined from the mean and shown in bold. SIW is the scaled interval width and is reported as a descriptive sharpness diagnostic only: a narrower interval is preferable only at matched calibration, so SIW is not bolded. The reported standard deviations are descriptive and are not used as a tie-breaker.

Dataset	λ_{adv}	sCRPS	CCE	SIW
Labour	0.0	0.0099 ± 0.0013	$\mathbf{0.0609 \pm 0.0370}$	0.0412 ± 0.0032
	0.1	$\mathbf{0.0093 \pm 0.0003}$	0.0664 ± 0.0514	0.0396 ± 0.0028
Tourism-S	0.0	0.0873 ± 0.0106	0.1648 ± 0.0376	0.2336 ± 0.0230
	0.1	$\mathbf{0.0808 \pm 0.0117}$	$\mathbf{0.0731 \pm 0.0598}$	0.3423 ± 0.0778
Tourism-L	0.0	0.1548 ± 0.0206	0.0842 ± 0.0151	0.3886 ± 0.0660
	0.1	$\mathbf{0.1366 \pm 0.0069}$	$\mathbf{0.0248 \pm 0.0187}$	0.4525 ± 0.0278
Traffic	0.0	0.0303 ± 0.0297	0.1872 ± 0.0242	0.3853 ± 0.1282
	0.1	$\mathbf{0.0245 \pm 0.0122}$	$\mathbf{0.1390 \pm 0.0409}$	0.4863 ± 0.1392
Wiki	0.0	0.1863 ± 0.0718	0.0663 ± 0.0748	0.3340 ± 0.0748
	0.1	$\mathbf{0.1838 \pm 0.0981}$	$\mathbf{0.0273 \pm 0.0124}$	0.4646 ± 0.0651

F.3 The Role of the ODE Solver

As introduced in Section 4, our conditional flow matching framework learns a continuous-time vector field $\mathbf{v}_\theta(\tau, \mathbf{X}(\tau))$ that maps a simple base distribution to the complex target data distribution. During inference, generating forecasts requires numerically simulating the corresponding ODE,

$$\frac{d\mathbf{X}(\tau)}{d\tau} = \mathbf{v}_\theta(\tau, \mathbf{X}(\tau))$$

along this learned vector field. Consequently, the choice of the numerical ODE solver is a critical hyperparameter. It directly dictates the trade-off between integration precision and computational efficiency. Integration precision governs the fidelity of the generated samples and the resulting predictive performance, whereas computational efficiency is vital for scalable inference. To comprehensively evaluate this accuracy-latency trade-off and justify our default solver choice, we ablate four distinct numerical methods of varying complexity and order:

- **Euler:** (Butcher, 2003) A first-order method requiring a single neural function evaluation (NFE) per step. While highly efficient, its linear approximation is susceptible to larger truncation errors in highly curved vector fields.
- **Heun:** (Ascher & Petzold, 1998) A second-order method that requires two NFEs per step, offering a pragmatic middle ground by accounting for the local curvature of the vector field.
- **RK4:** (Butcher, 2003) The classical fourth-order Runge-Kutta method. By evaluating the vector field four times per step, it drastically minimizes integration error and ensures high-fidelity trajectories, albeit at a significantly higher computational cost.
- **RK45:** (Fehlberg, 1969) A fifth-order adaptive step-size method that dynamically adjusts its integration steps to maintain the local error within a specified tolerance. This ensures optimal theoretical precision, though it introduces variable and often higher inference latency.

Table 7 summarizes the impact of different numerical ODE solvers and prediction-time step counts on forecasting accuracy (overall sCRPS).

Table 7: Ablation of the ODE solver and prediction-time step count for HCFM-MQCNN. Each cell reports overall sCRPS $\pm$ standard deviation across 3 rolling chained cross-validated windows. The ablation uses separate runs from Table 2, whose default sampling configuration is Heun with 20 steps. For adaptive RK45, the step-count setting controls the maximum step size.

Dataset	Solver	1	20	50	100
Traffic	Euler	0.0907 ± 0.0181	0.1047 ± 0.0099	0.1030 ± 0.0075	0.1024 ± 0.0097
	Heun	0.1186 ± 0.0220	0.0936 ± 0.0142	0.1024 ± 0.0093	0.1021 ± 0.0092
	RK4	0.1019 ± 0.0257	0.1026 ± 0.0097	0.1015 ± 0.0094	0.1035 ± 0.0086
	RK45	0.1429 ± 0.0607	0.0874 ± 0.0288	0.1033 ± 0.0086	0.1027 ± 0.0097
Tourism-S	Euler	0.0892 ± 0.0273	0.0756 ± 0.0178	0.0742 ± 0.0163	0.0743 ± 0.0163
	Heun	0.0924 ± 0.0105	0.0749 ± 0.0188	0.0748 ± 0.0190	0.0739 ± 0.0152
	RK4	0.0768 ± 0.0184	0.0754 ± 0.0194	0.0743 ± 0.0172	0.0749 ± 0.0190
	RK45	0.1001 ± 0.0205	0.0744 ± 0.0185	0.0744 ± 0.0184	0.0746 ± 0.0159
Tourism-L	Euler	0.1913 ± 0.0218	0.1468 ± 0.0167	0.1443 ± 0.0147	0.1449 ± 0.0177
	Heun	0.1555 ± 0.0181	0.1464 ± 0.0156	0.1443 ± 0.0174	0.1455 ± 0.0155
	RK4	0.1422 ± 0.0142	0.1445 ± 0.0136	0.1457 ± 0.0152	0.1458 ± 0.0163
	RK45	0.1572 ± 0.0126	0.1462 ± 0.0163	0.1476 ± 0.0152	0.1460 ± 0.0156
Labour	Euler	0.0095 ± 0.0028	0.0089 ± 0.0025	0.0091 ± 0.0028	0.0084 ± 0.0014
	Heun	0.0693 ± 0.0059	0.0099 ± 0.0029	0.0085 ± 0.0014	0.0085 ± 0.0014
	RK4	0.0258 ± 0.0003	0.0085 ± 0.0013	0.0085 ± 0.0014	0.0085 ± 0.0014
	RK45	0.0229 ± 0.0029	0.0084 ± 0.0013	0.0084 ± 0.0016	0.0084 ± 0.0016
Wiki	Euler	0.2410 ± 0.0981	0.1933 ± 0.0967	0.1911 ± 0.0923	0.1898 ± 0.0896
	Heun	0.2107 ± 0.0697	0.1923 ± 0.0929	0.1902 ± 0.0887	0.1917 ± 0.0927
	RK4	0.2025 ± 0.0885	0.1911 ± 0.0928	0.1907 ± 0.0925	0.1919 ± 0.0910
	RK45	0.2389 ± 0.1038	0.1922 ± 0.0931	0.1904 ± 0.0917	0.1912 ± 0.0924

Table 7 shows that very coarse integration is often insufficient, whereas moving from 1 to roughly 20 prediction-time steps captures most of the attainable gain. Beyond that point, improvements are typically small and sometimes negligible, with several datasets already near saturation by 20–50 steps. This makes Heun with 20 steps a reasonable default accuracy–latency compromise for the main experiments. Table 8 shows the expected runtime ordering driven largely by neural function evaluations: Euler is fastest, Heun is roughly twice as expensive, RK4 slower again, and RK45 the slowest and most variable because of adaptive stepping.

Table 8: Comparison of ODE solver inference times for HCFM-MQCNN using 20 steps to compute the solution (mean $\pm$ standard deviation, in milliseconds). Runtime is driven mainly by the number of neural function evaluations, so higher-order or adaptive solvers incur larger cost.

Dataset	Euler	Heun	RK4	RK45
Traffic	200 ± 39	500 ± 3.3	1000 ± 4.6	1500 ± 4.4
Tourism-S	56 ± 0.36	110 ± 4.9	230 ± 5.9	340 ± 6.3
Tourism-L	780 ± 8.3	1600 ± 8.0	3100 ± 97	3700 ± 410
Labour	65 ± 0.82	170 ± 6.1	350 ± 4.3	530 ± 4.9
Wiki	120 ± 25	370 ± 17	640 ± 33	1100 ± 57

Table 9: `Wiki` ablation comparing ℓ_1 and ℓ_2 flow-matching regression losses for HCFM-MQCNN and HCFM-UNet under otherwise matched settings within each backbone. This ablation uses separate runs from the main sCRPS and relSE evaluations in Tables 2 and 12. Each cell reports mean overall sCRPS or relSE $\pm$ standard deviation across 3 rolling chained cross-validated windows. Lower is better.

Backbone	Flow loss	Overall sCRPS	Overall relSE
HCFM-MQCNN	ℓ_1	0.1951 ± 0.0934	0.6384 ± 0.4073
HCFM-MQCNN	ℓ_2	0.3481 ± 0.0228	2.6235 ± 0.9385
HCFM-UNet	ℓ_1	0.1830 ± 0.0946	0.5452 ± 0.4039
HCFM-UNet	ℓ_2	0.3099 ± 0.0229	4.1197 ± 2.7929

F.4 The Role of Past-Future Co-generation

To isolate the impact of our proposed architecture, we perform an inference-time ablation that treats forecasting as an inpainting task. In this "past-frozen" setup, the network receives the observed past as a fixed condition at every ODE step. The predicted past-block velocity is discarded, meaning only the future block is integrated and the gradient past-term effect in the ODE is entirely removed. The importance of this ablation lies in comparing different inductive biases under finite data and model capacity. While a standard future-only conditional flow theoretically has the expressivity to represent the true predictive law, relying entirely on a conditioning pathway can cause the network to ignore relevant historical information. Consequently, the model may generate futures that look plausible under the marginal trajectory distribution but are weakly supported by the specific observed history. Designing proper inductive biases to prevent this inconsistency remains an active area of research in forecasting. To address this, HCFM co-evolves the past and future in a shared trajectory space, rather than relying strictly on an amortized conditional generator. Drawing inspiration from model-based inverse problems and data-consistent diffusion bridges, this formulation changes the underlying function-approximation problem. It separates known structural operators (the observed past and the hierarchy) from the learned residual dynamics. By utilizing a paired full-trajectory objective to learn past-future dependence and applying an explicit consistency term to keep the past anchored to actual observations, HCFM provides an independently controllable mechanism for observation fidelity. Results are shown in Table 10. Retaining past dynamics yields lower mean sCRPS on all five datasets. Across-window sCRPS variability decreases on `Labour`, `Tourism-S`, and `Traffic` and increases on `Tourism-L`. In Table 11, we provide a comparison of HCFM-UNet (Projected), HCFM-UNet (Leaves Only) and HCFM-UNet (Past-Frozen) using multivariate scores (see definitions in Appendix I).

Table 10: Inference-time ablation of past dynamics in HCFM-UNet. For each rolling window, both sampling configurations use the same trained U-Net, data split, and ODE solver settings. HCFM-UNet (Projected) co-evolves the past and future blocks using the projected-residual coherence mechanism. HCFM-UNet (Past-Frozen) supplies the observed past as a fixed input at every ODE step, discards the predicted past-block velocity, and integrates only the future block. This inference-time intervention removes the past-consistency term from the ODE while retaining the projected-residual coherence mechanism and the remaining sampler settings. Each cell reports mean scaled CRPS (sCRPS) $\pm$ standard deviation across 3 rolling chained cross-validated windows. Lower is better. The best result in each row is determined from the mean and shown in bold; the reported standard deviations are descriptive and are not used as a tie-breaker. Tourism-S and Tourism-L denote the small and large tourism hierarchies.

Dataset	HCFM-UNet (Projected)	HCFM-UNet (Past-Frozen)
Labour	$\mathbf{0.0112 \pm 0.0009}$	0.0132 ± 0.0036
Tourism-S	$\mathbf{0.0891 \pm 0.0160}$	0.1209 ± 0.0505
Tourism-L	$\mathbf{0.1339 \pm 0.0127}$	0.1362 ± 0.0108
Traffic	$\mathbf{0.0209 \pm 0.0049}$	0.0247 ± 0.0195
Wiki	$\mathbf{0.1851 \pm 0.1046}$	0.1886 ± 0.1045

Table 11: Comparison of three configurations of HCFM-UNet using multivariate scoring rules. All three use the same U-Net backbone, the same data splits and the same hyperparameters. HCFM-UNet (Leaves Only) forecasts only the leaf series and obtains forecasts for every non-leaf node by bottom-up aggregation. HCFM-UNet (Projected) applies the full HCFM dynamics over all nodes of the hierarchy with the projected-residual coherence mechanism. HCFM-UNet (Past-Frozen) holds the observed past fixed as a conditioning input and integrates the flow over the future block alone. Each cell reports the mean energy score (ES) or mean spatial variogram score (VS) $\pm$ standard deviation across 3 rolling chained cross-validated windows. Lower values are better for both metrics. The best configuration for each dataset and each metric is determined from the mean and shown in bold. The reported standard deviations are descriptive and are not used as a tie-breaker.

Dataset	Configuration	ES	Spatial VS
Labour	HCFM-UNet (Leaves Only)	148.579750 $\pm$ 23.365106	**5.100e+03 $\pm$ 3.160e+02**
	HCFM-UNet (Projected)	**139.109962 $\pm$ 23.147473**	5.162e+03 $\pm$ 9.398e+02
	HCFM-UNet (Past-Frozen)	170.863553 $\pm$ 48.275232	6.107e+03 $\pm$ 1.948e+03
Tourism-S	HCFM-UNet (Leaves Only)	**5262.928874 $\pm$ 937.969621**	7.527e+05 $\pm$ 1.748e+05
	HCFM-UNet (Projected)	5702.393555 $\pm$ 1695.886710	**7.409e+05 $\pm$ 1.246e+05**
	HCFM-UNet (Past-Frozen)	8756.702962 $\pm$ 4990.768044	9.377e+05 $\pm$ 3.017e+05
Tourism-L	HCFM-UNet (Leaves Only)	3099.247233 $\pm$ 261.635966	2.848e+07 $\pm$ 3.989e+06
	HCFM-UNet (Projected)	**2733.050212 $\pm$ 335.038377**	**2.551e+07 $\pm$ 3.352e+06**
	HCFM-UNet (Past-Frozen)	2875.653158 $\pm$ 327.820718	2.597e+07 $\pm$ 2.602e+06
Traffic	HCFM-UNet (Leaves Only)	43.175425 $\pm$ 16.353738	3.021e+03 $\pm$ 2.368e+03
	HCFM-UNet (Projected)	**26.223241 $\pm$ 12.506992**	**2.318e+03 $\pm$ 1.583e+03**
	HCFM-UNet (Past-Frozen)	41.362301 $\pm$ 41.248384	2.411e+03 $\pm$ 2.152e+03
Wiki	HCFM-UNet (Leaves Only)	**24164.621094 $\pm$ 19997.038774**	3.498e+07 $\pm$ 3.001e+07
	HCFM-UNet (Projected)	26529.038737 $\pm$ 23238.743120	**3.440e+07 $\pm$ 3.323e+07**
	HCFM-UNet (Past-Frozen)	28680.189128 $\pm$ 24027.489431	3.618e+07 $\pm$ 3.384e+07

G Effect of the Norm in the Regression Loss

This appendix studies how the regression exponent p changes the local population-optimal CFM field and its induced transport. We first characterize the coordinatewise L_p Bayes estimator and its ordered $p \to 0^+$ limit for a concentrating posterior mixture, then analyze posterior-tilt sensitivity and local moment dynamics. The calculations isolate the unrestricted regression objective; the structural operators and auxiliary regularizers enter the full HCFM objective separately. For convenience, we use $\mathbf{x}(\tau) = \mathrm{vec}(\mathbf{X}(\tau)) \in \mathbb{R}^d$, where $d = DL$.

G.1 Replacing L_2 changes the local Bayes projection

Conditional flow matching starts from a family of conditional probability paths indexed by the endpoint $\mathbf{x}(1) \sim q$. Marginalizing the conditional transport yields a vector field for the prescribed marginal path. For this analysis, let $\mathbf{x}(0) \sim p_0$ and $\mathbf{x}(1) \sim q$ be independent, and define

$$\mathbf{x}(\tau) = (1-\tau)\mathbf{x}(0) + \tau\mathbf{x}(1), \qquad \tau \in [0,1). \tag{40}$$

For each fixed $\mathbf{x}(1)$, the conditional probability distribution $p_\tau(\cdot \mid \mathbf{x}(1))$ is transported by

$$\mathbf{u}(\tau, \mathbf{x} \mid \mathbf{x}(1)) = \frac{\mathbf{x}(1) - \mathbf{x}}{1-\tau}, \tag{41}$$

and therefore satisfies

$$\partial_\tau p_\tau(\mathbf{x} \mid \mathbf{x}(1)) + \nabla_{\mathbf{x}} \cdot \big(p_\tau(\mathbf{x} \mid \mathbf{x}(1))\mathbf{u}(\tau, \mathbf{x} \mid \mathbf{x}(1))\big) = 0. \tag{42}$$

Integrating Eq. (42) over $\mathbf{x}(1)$ gives the marginal path

$$p_\tau(\mathbf{x}) = \int p_\tau(\mathbf{x} \mid \mathbf{x}(1))\, q(\mathbf{x}(1))\, d\mathbf{x}(1) \tag{43}$$

and the corresponding marginal continuity equation

$$\partial_\tau p_\tau(\mathbf{x}) + \nabla_{\mathbf{x}} \cdot \big(p_\tau(\mathbf{x})\mathbf{u}(\tau, \mathbf{x})\big) = 0 \tag{44}$$

with marginal field

$$\mathbf{u}(\tau, \mathbf{x}) = \frac{\int p_\tau(\mathbf{x} \mid \mathbf{x}(1)) \mathbf{u}(\tau, \mathbf{x} \mid \mathbf{x}(1))\, q(\mathbf{x}(1))\, d\mathbf{x}(1)}{\int p_\tau(\mathbf{x} \mid \mathbf{x}(1))\, q(\mathbf{x}(1))\, d\mathbf{x}(1)} = \mathbb{E}[\mathbf{u}(\tau, \mathbf{x}(\tau) \mid \mathbf{x}(1)) \mid \mathbf{x}(\tau) = \mathbf{x}]\,. \quad (45)$$

This conditional-mean construction yields a vector field satisfying the marginal continuity equation. The regression objective recovers this field from samples of the random target

$$\mathbf{u}(\tau, \mathbf{x}(\tau) \mid \mathbf{x}(1)) = \frac{\mathbf{x}(1) - \mathbf{x}(\tau)}{1 - \tau}. \quad (46)$$

Under the standard squared loss,

$$\mathcal{L}_2(\mathbf{v}) = \int_0^1 \mathbb{E}\|\mathbf{v}(\tau, \mathbf{x}(\tau)) - \mathbf{u}(\tau, \mathbf{x}(\tau) \mid \mathbf{x}(1))\|_2^2\, d\tau, \quad (47)$$

the pointwise Bayes minimizer is the conditional mean

$$\mathbf{v}^\star(\tau, \mathbf{x}) = \mathbb{E}[\mathbf{u}(\tau, \mathbf{x}(\tau) \mid \mathbf{x}(1)) \mid \mathbf{x}(\tau) = \mathbf{x}]\,, \quad (48)$$

which agrees exactly with Eq. (45). This is why L_2 plays a distinguished role in standard CFM. It is the choice that recovers the conditional expectation appearing in the marginal continuity equation.

We now replace Eq. (47) by the coordinatewise L_p objective

$$\mathcal{L}_p(\mathbf{v}) = \int_0^1 \mathbb{E}\left[\frac{1}{d}\sum_{j=1}^{d} |\big(\mathbf{v}(\tau, \mathbf{x}(\tau))\big)_j - \big(\mathbf{u}(\tau, \mathbf{x}(\tau) \mid \mathbf{x}(1))\big)_j|^p\right] d\tau, \qquad 0 < p \le 2. \quad (49)$$

The bridge Eq. (40), conditional field Eq. (41), and target Eq. (46) are held fixed. The exponent p determines how the posterior endpoint distribution is summarized into a deterministic regression estimate. For $0 < p < 1$, the objective uses coordinatewise power losses associated with a quasi-norm.

To keep the notation light, fix one coordinate index j. Write s for the corresponding j-th scalar coordinate of the endpoint $\mathbf{x}(1)$, and let x_j denote the j-th coordinate of the fixed current state $\mathbf{x}$. Also fix $(\mathbf{x}, \tau)$, and let $\rho_{\mathbf{x},\tau}$ denote the posterior density of s given $\mathbf{x}(\tau) = \mathbf{x}$. Define the scalar L_p Bayes estimator

$$\eta_p(\mathbf{x}, \tau) \in \arg\min_{m \in \mathbb{R}} \int |m - s|^p \rho_{\mathbf{x},\tau}(s)\, ds. \quad (50)$$

Then the corresponding coordinate of every population minimizer of Eq. (49) is

$$\big(\mathbf{v}_p^\star(\tau, \mathbf{x})\big)_j = \frac{\eta_p(\mathbf{x}, \tau) - x_j}{1 - \tau}. \quad (51)$$

Indeed, conditioning on $\mathbf{x}(\tau) = \mathbf{x}$ gives the scalar pointwise risk

$$a \mapsto \mathbb{E}\left[\left|a - \frac{s - x_j}{1 - \tau}\right|^p \,\middle|\, \mathbf{x}(\tau) = \mathbf{x}\right] = (1 - \tau)^{-p} \int |(1 - \tau)a + x_j - s|^p \rho_{\mathbf{x},\tau}(s)\, ds, \quad (52)$$

and the minimizer is obtained by the change of variables $m = (1 - \tau)a + x_j$. Eq. (51) describes the population-optimal field for the coordinate-wise power loss. The cases $p = 2$ and $p = 1$ yield the posterior mean and a posterior median, respectively. The small-exponent limit below characterizes the same regression family as $p \to 0^+$.

To see how reducing p changes the regression target, consider a scalar posterior with a few narrow peaks:

$$\rho_{\mathbf{x},\tau}^{(\sigma)}(s) = \sum_{\ell=1}^{M} w_\ell\, \mathcal{N}(s; m_\ell, s_\sigma^2), \qquad w_\ell > 0, \qquad \sum_{\ell=1}^{M} w_\ell = 1. \quad (53)$$

Each peak represents a plausible endpoint value m_ℓ with probability weight w_ℓ. We hold the distinct centers and their weights fixed while the common width $s_\sigma \to 0^+$ as $\sigma \to 0^+$. The expected regression loss is

$$R_{p,\sigma}(m) = \sum_{\ell=1}^{M} w_\ell \mathbb{E}|m - s_{\ell,\sigma}|^p, \qquad s_{\ell,\sigma} \sim \mathcal{N}(m_\ell, s_\sigma^2). \quad (54)$$

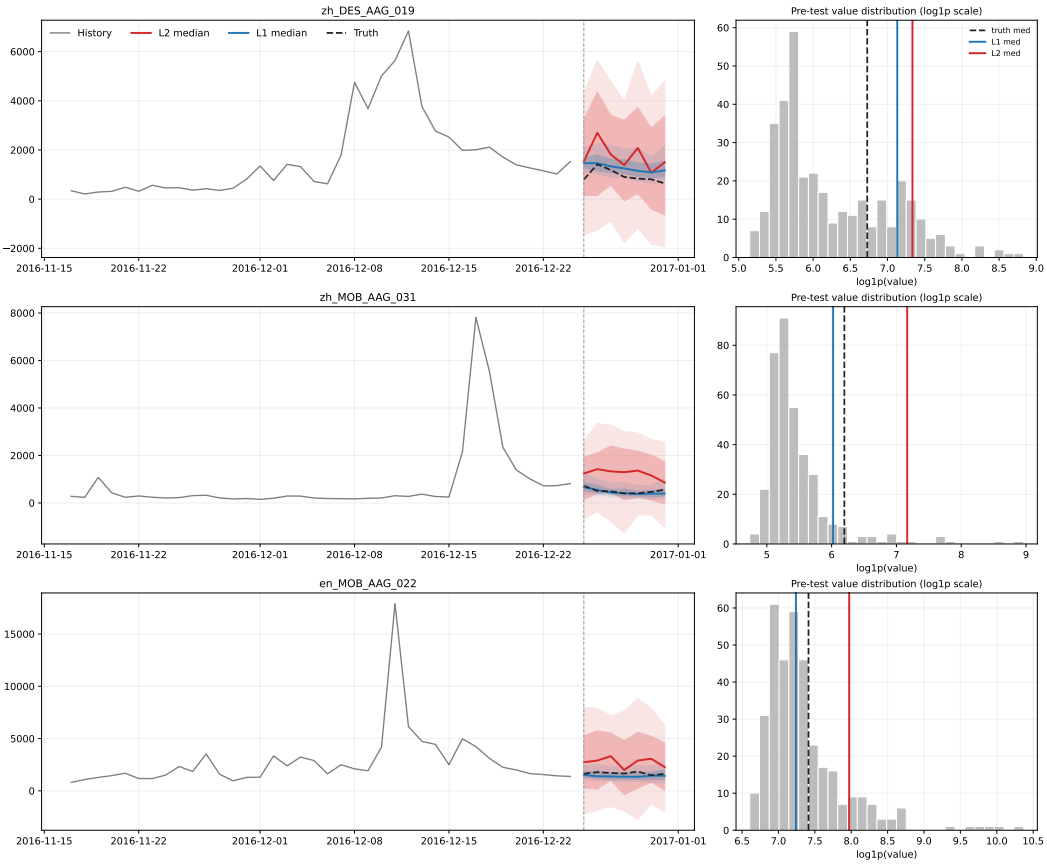


Figure 6: Comparison of L_1- and L_2-trained HCFM forecasts on bursty Wiki series. Left: recent history (gray), forecast origin (dashed gray), ground truth (dashed black), and median forecasts (blue/red) with 50%/80% intervals. Right: $\log(1+x)$ histogram of past values with lines for the true and forecast medians. The examples illustrate changes in forecast location and dispersion under the two regression losses.

First concentrate each peak at its center, keeping $p > 0$ fixed. Then let $p \to 0^+$. Every fixed nonzero error approaches a cost of one, while an exact match has zero cost:

$$\lim_{p\to 0^+}\left[\lim_{\sigma\to 0^+}\mathbb{E}|m - s_{\ell,\sigma}|^p\right] = \mathbf{1}_{\{m\neq m_\ell\}}. \tag{55}$$

Thus the limiting loss counts the posterior probability mass left unmatched:

$$R_0(m) := \lim_{p\to 0^+}\left[\lim_{\sigma\to 0^+} R_{p,\sigma}(m)\right] = 1 - \sum_{\ell: m_\ell = m} w_\ell. \tag{56}$$

Choosing a center m_j gives cost $1 - w_j$, and choosing a point between centers gives cost one. The limiting loss is therefore smallest at a center carrying the largest posterior mass. In this concentrated-posterior picture, the small-exponent limit favors a dominant endpoint value. Through Eq. (51), that endpoint choice determines the local CFM velocity. This illustrates the loss-dependent estimation viewpoint also used in denoising (Lehtinen et al., 2018).

G.2 Posterior tilt and moment dynamics

We now study how the regression exponent changes posterior-tilt sensitivity and local moment dynamics in a full-dimensional Gaussian-source diagnostic model. The source $\mathbf{x}(0) \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$ is independent of $\mathbf{x}(1) \sim q$, and the sensitivity calculations fix $\tau \in (0,1)$. Assume that the

scalar baseline posterior has $\int (1+s^2)e^{a|s|}\rho_0(s)\,ds < \infty$ for some $a > 0$, and that the posterior densities are continuous and positive at their unique medians for sufficiently small displacements. These assumptions justify the differentiations below. The moment identities apply on intervals where the selected population field generates a sufficiently regular transport with finite second moments, integrable $x_k(\mathbf{v}_p^\star)_k$, and boundary decay sufficient for integration by parts. The local Taylor calculation assumes a twice continuously differentiable field near the expansion point. The continuity equation then yields exact moment identities, while the element approximation below specifies when a diagonal Jacobian controls the leading variance change.

Fix a flow time τ, fix a point $\bar{\mathbf{x}} \in \mathbb{R}^d$, and fix one coordinate k. Our goal is to understand

$$\partial_{x_k}\big(\mathbf{v}_p^\star(\tau, \bar{\mathbf{x}})\big)_k.$$

To compute this derivative, we perturb only the k-th coordinate of the current state. For a scalar y, define

$$\mathbf{x}(y)_i = \bar{\mathbf{x}}_i \quad \text{for } i \neq k, \qquad \mathbf{x}(y)_k = \bar{\mathbf{x}}_k + y. \tag{57}$$

Thus $\mathbf{x}(0) = \bar{\mathbf{x}}$. The variable y is introduced only because we want a derivative. The value $y = 0$ is simply the base point where the Jacobian is evaluated.

Let s denote the k-th coordinate of $\mathbf{x}(1)$, and let ρ_y denote the posterior density of s given $\mathbf{x}(\tau) = \mathbf{x}(y)$. The key point is that ρ_y is not an arbitrary family of posteriors. It is obtained from the posterior at the base point by a very specific reweighting dictated by Bayes' rule. Starting from the straight-path model

$$\mathbf{x}(\tau) = (1-\tau)\mathbf{x}(0) + \tau\mathbf{x}(1), \qquad \mathbf{x}(0) \sim p_0, \qquad \mathbf{x}(1) \sim q,$$

with $\mathbf{x}(0) \sim \mathcal{N}(0, \mathbf{I})$, the conditional law of $\mathbf{x}(\tau)$ given $\mathbf{x}(1)$ is

$$p_\tau(\mathbf{x} \mid \mathbf{x}(1)) \propto \exp\left(-\frac{\|\mathbf{x} - \tau\mathbf{x}(1)\|^2}{2(1-\tau)^2}\right).$$

Hence Bayes' rule gives

$$q_\tau(\mathbf{x}(1) \mid \mathbf{x}) \propto q(\mathbf{x}(1)) \exp\left(-\frac{\|\mathbf{x} - \tau\mathbf{x}(1)\|^2}{2(1-\tau)^2}\right). \tag{58}$$

We now evaluate this posterior at the perturbed point $\mathbf{x}(y)$. Writing $s = (\mathbf{x}(1))_k$, we have

$$\|\mathbf{x}(y) - \tau\mathbf{x}(1)\|^2 = \|\bar{\mathbf{x}} - \tau\mathbf{x}(1)\|^2 + 2y(\bar{\mathbf{x}}_k - \tau s) + y^2.$$

Substituting this into Eq. (58) yields

$$q_\tau(\mathbf{x}(1) \mid \mathbf{x}(y)) \propto q(\mathbf{x}(1)) \exp\left(-\frac{\|\bar{\mathbf{x}} - \tau\mathbf{x}(1)\|^2}{2(1-\tau)^2}\right) \exp\left(-\frac{2y(\bar{\mathbf{x}}_k - \tau s) + y^2}{2(1-\tau)^2}\right).$$

Split the last exponential into the part that depends on $\mathbf{x}(1)$ and the part that does not:

$$\exp\left(-\frac{2y(\bar{\mathbf{x}}_k - \tau s) + y^2}{2(1-\tau)^2}\right) = \exp\left(-\frac{2y\bar{\mathbf{x}}_k + y^2}{2(1-\tau)^2}\right) \exp\left(\frac{\tau}{(1-\tau)^2} ys\right).$$

The first factor does not depend on $\mathbf{x}(1)$ and is absorbed into the normalization constant. Therefore

$$q_\tau(\mathbf{x}(1) \mid \mathbf{x}(y)) \propto e^{\gamma_\tau ys}\, q_\tau(\mathbf{x}(1) \mid \bar{\mathbf{x}}), \qquad \gamma_\tau := \frac{\tau}{(1-\tau)^2}. \tag{59}$$

This is the full posterior-tilt identity. It says that moving the current state by an amount y in coordinate k multiplies the posterior weight of an endpoint $\mathbf{x}(1)$ by a factor that depends only on its k-th coordinate $s = (\mathbf{x}(1))_k$.

We want a formula not for the full posterior of $\mathbf{x}(1)$, but only for the posterior of the single coordinate $s = (\mathbf{x}(1))_k$. This step is not completely obvious, so we write it explicitly. Write

$$\mathbf{x}(1) = (\tilde{\mathbf{x}}(1), s),$$

where $\tilde{\mathbf{x}}(1) \in \mathbb{R}^{d-1}$ collects the remaining coordinates. Define the posterior density of s by

$$\rho_y(s) := \int q_\tau(\tilde{\mathbf{x}}(1), s \mid \mathbf{x}(y))\, d\tilde{\mathbf{x}}(1). \tag{60}$$

Likewise define

$$\rho_0(s) := \int q_\tau(\tilde{\mathbf{x}}(1), s \mid \bar{\mathbf{x}})\, d\tilde{\mathbf{x}}(1).$$

Since the tilt factor in Eq. (59) depends only on s, integrating out $\tilde{\mathbf{x}}(1)$ preserves the same exponential reweighting:

$$\begin{aligned}\rho_y(s) &= \int q_\tau(\tilde{\mathbf{x}}(1), s \mid \mathbf{x}(y))\, d\tilde{\mathbf{x}}(1) \\ &= \frac{e^{\gamma_\tau y s} \int q_\tau(\tilde{\mathbf{x}}(1), s \mid \bar{\mathbf{x}})\, d\tilde{\mathbf{x}}(1)}{\int e^{\gamma_\tau y u} \left[\int q_\tau(\tilde{\mathbf{u}}(1), u \mid \bar{\mathbf{x}})\, d\tilde{\mathbf{u}}(1)\right] du} \\ &= \frac{e^{\gamma_\tau y s} \rho_0(s)}{\int e^{\gamma_\tau y u} \rho_0(u)\, du}. \end{aligned} \tag{61}$$

Equation (61) is the precise posterior-tilt formula. It shows that changing the current state in coordinate k does not arbitrarily change the posterior of the endpoint coordinate s. It only tilts the baseline posterior ρ_0 by the exponential factor $e^{\gamma_\tau y s}$ and then renormalizes.

The L_p loss compresses this tilted posterior into a single scalar estimator. Define

$$\eta_p(y) \in \arg\min_{m \in \mathbb{R}} \int |m - s|^p \rho_y(s)\, ds. \tag{62}$$

The corresponding coordinate of the exact population field is then

$$\big(\mathbf{v}_p^\star(\tau, \mathbf{x}(y))\big)_k = \frac{\eta_p(y) - (\bar{\mathbf{x}}_k + y)}{1 - \tau}. \tag{63}$$

This identity contains the entire deformation mechanism. The term $-(\bar{\mathbf{x}}_k + y)/(1-\tau)$ is the restoring contribution inherited from the straight conditional field. The term $\eta_p(y)/(1-\tau)$ is the correction supplied by the posterior estimator. Whether the field contracts or expands in coordinate k is determined by how strongly $\eta_p(y)$ reacts to the posterior tilt.

Differentiating Eq. (63) at $y = 0$ gives

$$J_p(\bar{\mathbf{x}}, \tau) := \partial_{x_k} \big(\mathbf{v}_p^\star(\tau, \bar{\mathbf{x}})\big)_k = \frac{\eta_p'(0) - 1}{1 - \tau}. \tag{64}$$

Thus the sign of the local Jacobian is governed by the tilt sensitivity $\eta_p'(0)$. If $\eta_p'(0) < 1$, then $J_p(\bar{\mathbf{x}}, \tau) < 0$, so the field contracts in coordinate k. If $\eta_p'(0)$ is smaller, then the Jacobian is more negative and the contraction is stronger.

We now connect this local Jacobian to exact moment dynamics. Let $\tilde{p}_\tau$ denote the probability distribution transported by the exact population field $\mathbf{v}_p^\star(\tau, \cdot)$, so that

$$\partial_\tau \tilde{p}_\tau(\mathbf{x}) + \nabla_{\mathbf{x}} \cdot \big(\tilde{p}_\tau(\mathbf{x}) \mathbf{v}_p^\star(\tau, \mathbf{x})\big) = 0. \tag{65}$$

For any smooth test function $\varphi : \mathbb{R}^d \to \mathbb{R}$, integration by parts gives

$$\frac{d}{d\tau} \int \varphi(\mathbf{x}) \tilde{p}_\tau(\mathbf{x})\, d\mathbf{x} = \int \nabla_{\mathbf{x}} \varphi(\mathbf{x}) \cdot \mathbf{v}_p^\star(\tau, \mathbf{x})\, \tilde{p}_\tau(\mathbf{x})\, d\mathbf{x}. \tag{66}$$

Choosing $\varphi(\mathbf{x}) = x_k$ yields

$$\frac{d}{d\tau} \mathbb{E}_{\tilde{p}_\tau}[x_k] = \mathbb{E}_{\tilde{p}_\tau}\big[\big(\mathbf{v}_p^\star(\tau, \mathbf{x})\big)_k\big]. \tag{67}$$

Choosing $\varphi(\mathbf{x}) = x_k^2$ yields

$$\frac{d}{d\tau} \mathbb{E}_{\tilde{p}_\tau}[x_k^2] = 2\, \mathbb{E}_{\tilde{p}_\tau}\big[x_k \big(\mathbf{v}_p^\star(\tau, \mathbf{x})\big)_k\big]. \tag{68}$$

Therefore the coordinate-wise variance

$$V_k(\tau) := \mathrm{Var}_{\tilde{p}_\tau}(x_k) = \mathbb{E}_{\tilde{p}_\tau}[x_k^2] - \mathbb{E}_{\tilde{p}_\tau}[x_k]^2 \tag{69}$$

satisfies the exact identity

$$\dot{V}_k(\tau) = 2\, \mathrm{Cov}_{\tilde{p}_\tau}\Big(x_k, \big(\mathbf{v}_p^\star(\tau, \mathbf{x})\big)_k\Big). \tag{70}$$

Equation (70) is the deterministic transport analogue of the standard second-moment equation for an SDE, without the diffusion term.

We then expand the k-th coordinate of the field around the base point $\bar{\mathbf{x}}$:

$$\big(\mathbf{v}_p^\star(\tau,\mathbf{x})\big)_k = \big(\mathbf{v}_p^\star(\tau,\bar{\mathbf{x}})\big)_k + J_p(\bar{\mathbf{x}},\tau)\,(x_k-\bar{\mathbf{x}}_k) + \sum_{\ell\neq k}\partial_{x_\ell}\big(\mathbf{v}_p^\star(\tau,\bar{\mathbf{x}})\big)_k\,(x_\ell-\bar{\mathbf{x}}_\ell) + r_\tau(\mathbf{x}), \quad (71)$$

where the remainder $r_\tau(\mathbf{x})$ is higher order in $\mathbf{x}-\bar{\mathbf{x}}$. Substituting Eq. (71) into Eq. (70) gives the exact identity

$$\dot{V}_k(\tau) = 2\,J_p(\bar{\mathbf{x}},\tau)V_k(\tau) + 2\sum_{\ell\neq k}\partial_{x_\ell}\big(\mathbf{v}_p^\star(\tau,\bar{\mathbf{x}})\big)_k\,\mathrm{Cov}_{\tilde{p}_\tau}(x_k,x_\ell) + 2\,\mathrm{Cov}_{\tilde{p}_\tau}\big(x_k,r_\tau(\mathbf{x})\big). \quad (72)$$

The diagonal Jacobian contributes alongside cross-coordinate covariances and the Taylor remainder. For a family of elements localized near $\bar{\mathbf{x}}$, assume $\lambda_k>0$ and

$$\sum_{\ell\neq k}\big|\partial_{x_\ell}\big(\mathbf{v}_p^\star(\tau,\bar{\mathbf{x}})\big)_k\,\mathrm{Cov}_{\tilde{p}_\tau}(x_k,x_\ell)\big| + \big|\mathrm{Cov}_{\tilde{p}_\tau}\big(x_k,r_\tau(\mathbf{x})\big)\big| = o(\lambda_k)$$

as the element width tends to zero. Under this condition,

$$\dot{\lambda}_k(\tau) = 2J_p(\bar{\mathbf{x}},\tau)V_k(\tau) + o(\lambda_k). \quad (73)$$

A fixed negative $J_p(\bar{\mathbf{x}},\tau)$ therefore yields decreasing coordinatewise variance for sufficiently localized elements satisfying this condition. Comparing two fields on the same element gives a comparison of their leading local variance rates.

It remains to compute $\eta_2'(0)$ and $\eta_1'(0)$. From the posterior-tilt formula

$$\rho_y(s) = \frac{e^{\gamma_\tau y s}\rho_0(s)}{\int e^{\gamma_\tau y u}\rho_0(u)\,du},$$

one gets by direct differentiation

$$\partial_y\rho_y(s) = \gamma_\tau\big(s-\mu_y\big)\rho_y(s), \qquad \mu_y := \int u\,\rho_y(u)\,du. \quad (74)$$

For $p=2$, $\eta_2(y)$ is simply the mean of the posterior distribution ρ_y. Therefore

$$\eta_2(y) = \int s\,\rho_y(s)\,ds,$$

and differentiating gives

$$\eta_2'(0) = \int s\,\partial_y\rho_y(s)\big|_{y=0}\,ds = \gamma_\tau\int s(s-\mu_0)\rho_0(s)\,ds = \gamma_\tau\mathrm{Var}_{\rho_0}(s).$$

Hence

$$\eta_2'(0) = \gamma_\tau\mathrm{Var}_{\rho_0}(s). \quad (75)$$

For $p=1$, $\eta_1(y)$ is the median of the posterior distribution ρ_y. Thus it is defined by

$$\int_{-\infty}^{\eta_1(y)}\rho_y(s)\,ds = \frac{1}{2}.$$

Differentiating this identity with respect to y gives

$$\eta_1'(y) = -\frac{\int_{-\infty}^{\eta_1(y)}\partial_y\rho_y(s)\,ds}{\rho_y(\eta_1(y))}.$$

Evaluating at $y=0$ and substituting Eq. (74) yields

$$\eta_1'(0) = -\frac{\gamma_\tau}{\rho_0(\eta_1(0))}\int_{-\infty}^{\eta_1(0)}(s-\mu_0)\rho_0(s)\,ds, \qquad \mu_0 := \int s\,\rho_0(s)\,ds. \quad (76)$$

If the coordinate is centered so that $\eta_1(0) = 0$ and the baseline posterior is symmetric, then $\mu_0 = 0$ and the formula simplifies to

$$\eta_1'(0) = \gamma_\tau \frac{\int |s|\rho_0(s)\, ds}{2\rho_0(0)}. \tag{77}$$

The posterior tilt is determined by Bayes' rule and the Gaussian-source path. The regression exponent determines the estimator's response: Eq. (75) depends on posterior variance, while Eq. (77) depends on the first absolute moment and the density at the median. For an even posterior centered at zero, these formulas give the exact criterion

$$\eta_1'(0) < \eta_2'(0) \quad \Longleftrightarrow \quad \int |s|\rho_0(s)\, ds < 2\rho_0(0)\mathrm{Var}_{\rho_0}(s).$$

When this condition holds, Eq. (64) gives a smaller diagonal Jacobian under L_1 and, for elements satisfying the condition preceding Eq. (73), a smaller leading local variance rate. The following Gaussian-mixture example exhibits this regime explicitly.

G.3 Gaussian-to-eight-Gaussian example

These statements become completely explicit in the Gaussian-to-eight-Gaussian example. Let

$$q(\mathbf{x}(1)) = \frac{1}{8}\sum_{\ell=0}^{7} \mathcal{N}(\mathbf{x}(1); \mu_\ell, \sigma^2 \mathbf{I}_2), \qquad \mu_\ell = R\big(\cos(2\pi\ell/8), \sin(2\pi\ell/8)\big). \tag{78}$$

Here, $R > 0$ denotes the radius of the target ring, which distributes the centers of the eight Gaussian modes evenly around the origin. Choose the base point on the positive horizontal axis,

$$\bar{\mathbf{x}} = (r, 0),$$

and take $k = 2$, so the selected coordinate is the vertical coordinate. In this case the baseline posterior ρ_0 is an even Gaussian mixture of the form

$$\rho_0(s) = w_0\, \phi_{S_\tau}(s) + \frac{w_1}{2}\big[\phi_{S_\tau}(s - m_1(\tau)) + \phi_{S_\tau}(s + m_1(\tau))\big] + \frac{w_2}{2}\big[\phi_{S_\tau}(s - m_2(\tau)) + \phi_{S_\tau}(s + m_2(\tau))\big], \tag{79}$$

where $\phi_S(s) = (2\pi S)^{-1/2}\exp(-s^2/(2S))$ denotes the centered Gaussian density with variance S, and

$$\nu_\tau^2 = (1-\tau)^2 + \tau^2\sigma^2, \qquad S_\tau = \frac{\sigma^2(1-\tau)^2}{\nu_\tau^2}, \qquad m_1(\tau) = \frac{(1-\tau)^2(R/\sqrt{2})}{\nu_\tau^2}, \qquad m_2(\tau) = \frac{(1-\tau)^2 R}{\nu_\tau^2}. \tag{80}$$

The grouped posterior weights satisfy $w_0 + w_1 + w_2 = 1$, where w_0 corresponds to the two modes with zero vertical coordinate, w_1 to the four diagonal modes, and w_2 to the two vertical modes. Their unnormalized forms are

$$\begin{aligned}
\bar{w}_0 &= \exp\left(-\frac{(r - \tau R)^2}{2\nu_\tau^2}\right) + \exp\left(-\frac{(r + \tau R)^2}{2\nu_\tau^2}\right), \\
\bar{w}_1 &= 2\exp\left(-\frac{(r - \tau R/\sqrt{2})^2 + \tau^2 R^2/2}{2\nu_\tau^2}\right) + 2\exp\left(-\frac{(r + \tau R/\sqrt{2})^2 + \tau^2 R^2/2}{2\nu_\tau^2}\right), \\
\bar{w}_2 &= 2\exp\left(-\frac{r^2 + \tau^2 R^2}{2\nu_\tau^2}\right),
\end{aligned} \tag{81}$$

and

$$w_i = \frac{\bar{w}_i}{\bar{w}_0 + \bar{w}_1 + \bar{w}_2}, \qquad i \in \{0, 1, 2\}.$$

Since Eq. (79) is even, the median at the base point is zero and Eq. (77) applies.

For $p = 2$, Eq. (75) gives

$$\eta_2'(0) = \gamma_\tau\Big(S_\tau + w_1 m_1(\tau)^2 + w_2 m_2(\tau)^2\Big). \tag{82}$$

For $p = 1$, Eq. (77) gives

$$\eta_1'(0) = \gamma_\tau \frac{\int |s|\rho_0(s)\, ds}{2\rho_0(0)}. \tag{83}$$

Consider the narrow-component limit $\sigma \to 0^+$ with $\tau \in (0,1)$, $r > 0$, and $R > 0$ fixed. Then γ_τ is fixed, $S_\tau \sim \sigma^2$, $m_1(\tau) \to R/\sqrt{2}$, and $m_2(\tau) \to R$. The posterior weights converge to positive limits $w_i^0 = \lim_{\sigma\to 0^+} w_i$, so

$$\rho_0(0) \sim \frac{w_0^0}{\sqrt{2\pi S_\tau}}, \qquad \int |s|\rho_0(s)\, ds \longrightarrow w_1^0 R/\sqrt{2} + w_2^0 R.$$

Consequently,

$$\eta_1'(0) \sim \frac{\gamma_\tau\sqrt{2\pi}}{2w_0^0}\big(w_1^0 R/\sqrt{2} + w_2^0 R\big)\sqrt{S_\tau} = \mathcal{O}(S_\tau^{1/2}), \tag{84}$$

whereas Eq. (82) gives

$$\eta_2'(0) \longrightarrow \gamma_\tau R^2\big(w_1^0/2 + w_2^0\big) > 0.$$

Thus $\eta_1'(0)/\eta_2'(0) \to 0$, and for sufficiently small σ,

$$\partial_{x_2}\big(\mathbf{v}_1^\star(\tau,\bar{\mathbf{x}})\big)_2 < \partial_{x_2}\big(\mathbf{v}_2^\star(\tau,\bar{\mathbf{x}})\big)_2, \qquad \partial_{x_2}\big(\mathbf{v}_1^\star(\tau,\bar{\mathbf{x}})\big)_2 \longrightarrow -\frac{1}{1-\tau}.$$

Reflection symmetry gives $\big(\mathbf{v}_p^\star(\tau,(x_1,0))\big)_2 = 0$ for $p \in \{1,2\}$, so the cross derivative $\partial_{x_1}(\mathbf{v}_p^\star)_2$ vanishes on this axis. For localized elements satisfying the remainder condition preceding Eq. (73), the derivative ordering yields a smaller leading vertical variance rate under L_1. The L_1 rate is negative for sufficiently small σ. This establishes a local transverse contraction mechanism in the fixed-time, narrow-component regime.

We characterize how the regression exponent p changes posterior averaging and local transport. In straight-path CFM, L_2 regression averages endpoint-conditioned velocities by their posterior weights (Lipman et al., 2023; Tong et al., 2023). Under independent coupling, this averaging can curve marginal trajectories and complicate learning and integration (Tong et al., 2023). After the scalar posterior peaks concentrate, the limit $p \to 0^+$ favors endpoints carrying the largest posterior mass. For $p \in \{1,2\}$, the posterior-tilt analysis relates estimator sensitivity to local variance dynamics. In the specified narrow-component regime of the Gaussian-to-eight-Gaussian example, L_1 yields a more negative transverse derivative and stronger leading-order local contraction. Figure 6 illustrates changes in forecast location and dispersion, while Table 9 reports lower overall sCRPS and relSE with L_1 for both HCFM-MQCNN and HCFM-UNet on bursty `Wiki` series.

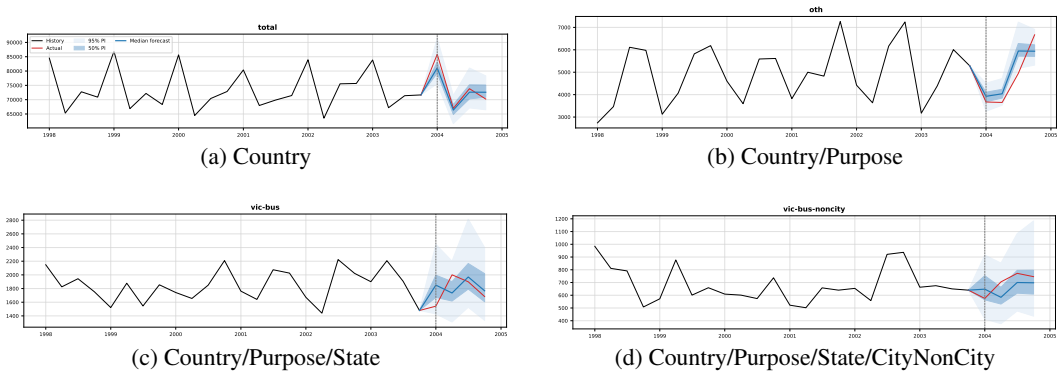
H Visualizing Forecast Distributions

We provide exemplar forecast distributions generated by HCFM across all aggregation levels for the `Tourism-S`, `Tourism-L`, `Wiki`, and `Labour` datasets. In Figures 7–10, the black curve shows the observed context window, the red curve shows the realized future, the blue line shows the median forecast, and the blue ribbons show the central 50% and 95% prediction intervals. A vertical dashed line marks the forecast origin, i.e. the boundary between the observed history and the forecast horizon. No individual sample paths are shown. The plotted examples are randomly selected from each hierarchy and are intended to illustrate the qualitative behavior of the predictive distribution under coherence-guided dynamics.

I Additional results

I.1 Evaluation of point-wise forecasts

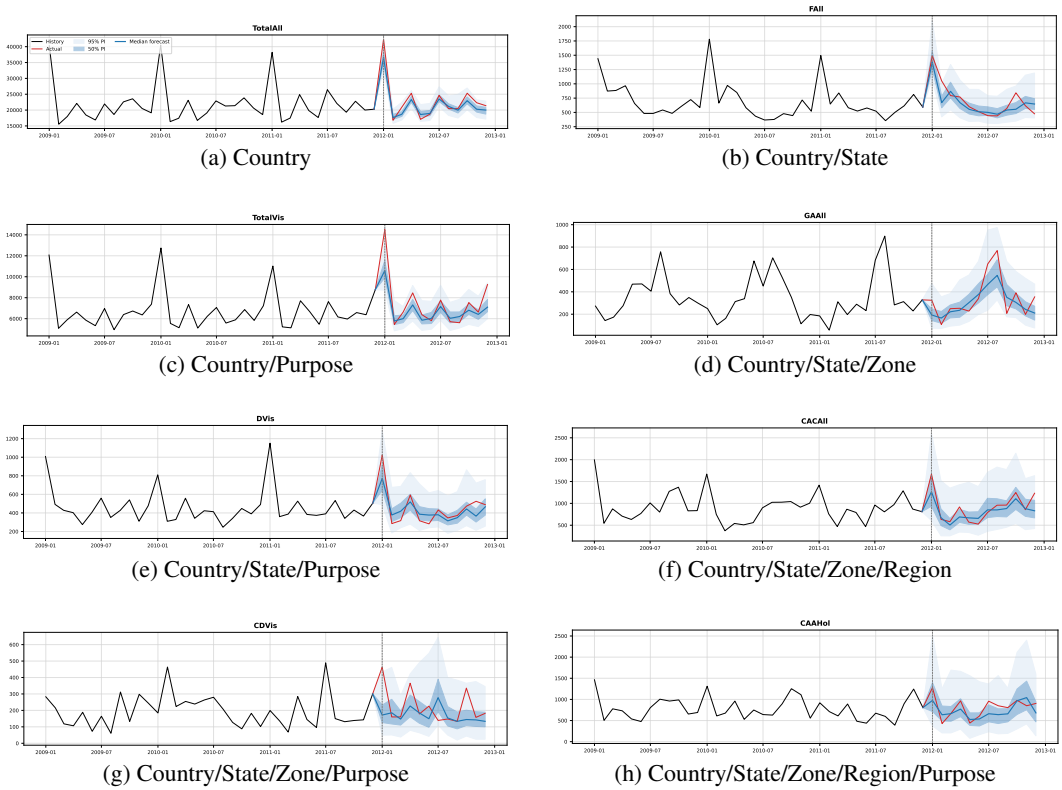
To complement the probabilistic results in Section 5, we also evaluate mean forecasts through the relative squared error (relSE). Let $\bar{y}_{[i][h],t} := (\bar{y}_{[i],1,t}, \ldots, \bar{y}_{[i],N_h,t})$ denote the mean forecast for series i at forecast time origin t over horizons $h \in \{1,\ldots,N_h\}$. relSE compares the squared error of these mean forecasts (aggregated across hierarchy levels) to the squared error of the Naive baseline that uses the last observation $y_{[i],t}$ as a point forecast. Table 12 reports relSE across datasets and hierarchy levels.



(a) Country (b) Country/Purpose

(c) Country/Purpose/State (d) Country/Purpose/State/CityNonCity

Figure 7: Exemplar forecast distributions for the `Tourism-S` dataset across its four hierarchical levels. Black denotes the observed context, red the realized future, the blue line the median forecast, and the blue ribbons the central 50% and 95% prediction intervals. The vertical dashed line marks the forecast origin.



(a) Country (b) Country/State

(c) Country/Purpose (d) Country/State/Zone

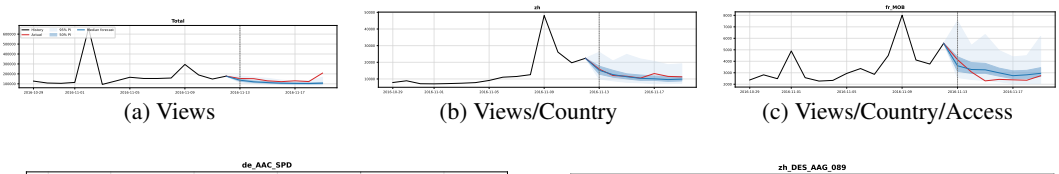
(e) Country/State/Purpose (f) Country/State/Zone/Region

(g) Country/State/Zone/Purpose (h) Country/State/Zone/Region/Purpose

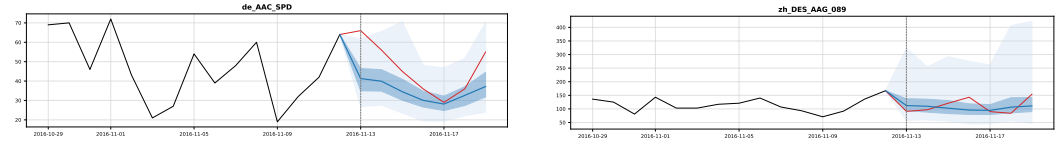
Figure 8: Hierarchical forecast distributions for the `Tourism-L` dataset. Black denotes the observed context, red the realized future, the blue line the median forecast, and the blue ribbons the central 50% and 95% prediction intervals. The vertical dashed line marks the forecast origin.

I.2 Comparisons with general-purpose forecasting models

We compare HCFM-UNet (Full Soft) with its Leaves Only ablation and N-HiTS (Bottom-Up) (Challu et al., 2023). Leaves Only retains our ℓ_1 trajectory forecaster and past consistency on the leaves, followed by bottom-up aggregation. Full Soft achieves the lowest overall mean sCRPS on all five datasets (Table 13).

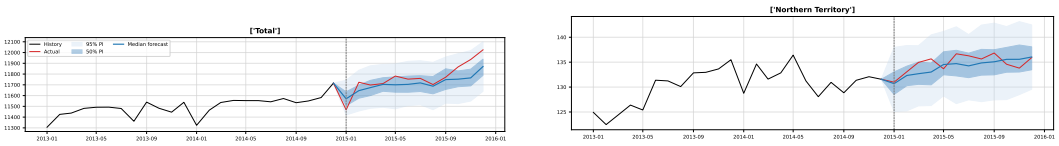


(a) Views (b) Views/Country (c) Views/Country/Access

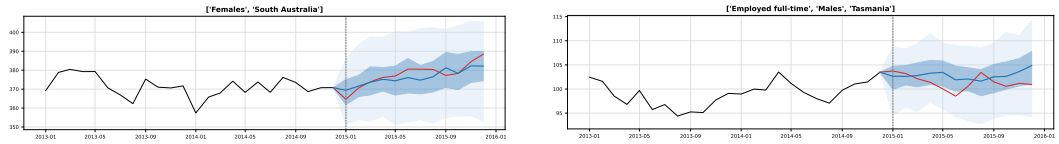


(d) Views/Country/Access/Agent (e) Views/Country/Access/Agent/Topic

Figure 9: Predictive distributions for the `Wiki` dataset. Black denotes the observed context, red the realized future, the blue line the median forecast, and the blue ribbons the central 50% and 95% prediction intervals. The vertical dashed line marks the forecast origin.



(a) Country (b) Country/Region



(c) Country/Gender/Region (d) Country/Employment/Gender/Region

Figure 10: Probabilistic forecasts for the `Labour` dataset. Black denotes the observed context, red the realized future, the blue line the median forecast, and the blue ribbons the central 50% and 95% prediction intervals. The vertical dashed line marks the forecast origin.

N-HiTS (Bottom-Up). For each leaf series i, the N-HiTS backbone processes its observed history $\mathbf{x}_i$ to produce the horizon output

$$\mathbf{z}_{i,1:H} = \mathrm{NHiTS}(\mathbf{x}_i).$$

A learned linear output head maps $\mathbf{z}_{i,1:H}$ to location and scale vectors $\boldsymbol{\mu}_i$, $\boldsymbol{\sigma}_i$, and a factor-loading matrix $\mathbf{F}_i$. Leaf forecast samples are generated as

$$\tilde{\mathbf{y}}_i^{(k)} = \mathrm{ReLU}\Big(\boldsymbol{\mu}_i + \boldsymbol{\sigma}_i \odot \boldsymbol{\epsilon}_i^{(k)} + \mathbf{F}_i \boldsymbol{\eta}^{(k)}\Big),$$

where $\boldsymbol{\epsilon}_i^{(k)}$ is series-specific noise and $\boldsymbol{\eta}^{(k)}$ is shared across leaf series for sample k. The loadings $\mathbf{F}_i$ couple the leaf forecasts through the shared factors, $\odot$ denotes elementwise multiplication, and ReLU clips negative sampled values to zero. The model is trained with a Monte Carlo CRPS objective. Stacking the leaf trajectories into $\widetilde{\mathbf{Y}}^{(k)}_{\mathrm{bottom}}$, we reconstruct the full-hierarchy forecast by bottom-up aggregation:

$$\widetilde{\mathbf{Y}}^{(k)}_{\mathrm{all}} = \mathbf{S}\widetilde{\mathbf{Y}}^{(k)}_{\mathrm{bottom}},$$

where $\mathbf{S}$ is the summation matrix. This reconstruction yields coherent forecasts at every hierarchy level and forecast step for each sample.

I.3 Multivariate probabilistic evaluation

sCRPS is a univariate score evaluated independently per series. It fails to evaluate how well different models capture cross-series correlations. We cannot use the Negative Log-Likelihood (NLL) because HCFM lacks a tractable predictive likelihood, and the baselines evaluated in Section 5 do not all define a likelihood over the same joint object. We therefore complement our analysis with two

Table 12: Empirical evaluation of mean hierarchical forecasts. Each cell reports mean relative squared error (relSE) ± standard deviation across 3 rolling chained cross-validated windows. Lower is better. HCFM entries use the default soft-guided dynamics in Eq. (4). Best and second-best results are determined from the unrounded mean; the reported standard deviations are descriptive and are not used as a tie-breaker.

Dataset	Level / Hierarchy	HCFM-MQCNN	HCFM-UNet	CLOVER	HierE2E	ProbHardE2E	Hier-Transformer-CNF	PERMBU-MinT*
Traffic	Overall	1.3508 ± 1.7304	0.1676 ± 0.2143	**0.1334 ± 0.1228**	0.3821 ± 0.2833	1.0617 ± 1.5814	0.5988 ± 0.1787	0.6740 ± 0.2135
	Level1	2.3079 ± 3.4107	0.3054 ± 0.4743	**0.1191 ± 0.1026**	0.3948 ± 0.3456	1.6993 ± 2.7855	0.5923 ± 0.1841	0.5186 ± 0.4034
	Level2	2.1530 ± 3.1283	0.2566 ± 0.3926	**0.1318 ± 0.1133**	0.4979 ± 0.5206	1.8364 ± 2.9616	0.6024 ± 0.1742	0.8398 ± 0.2977
	Level3	0.9387 ± 0.9687	**0.1308 ± 0.1185**	0.1565 ± 0.2097	0.3825 ± 0.2756	0.8528 ± 1.0256	0.6049 ± 0.1726	0.6394 ± 0.1427
	Level4	0.8171 ± 0.1015	**0.1381 ± 0.1987**	0.3658 ± 0.3881	0.6333 ± 0.2077	0.8571 ± 0.1360	0.8188 ± 0.1034	1.1047 ± 0.4676
Tourism-S	Overall	0.2540 ± 0.1876	0.2721 ± 0.1425	**0.2315 ± 0.0951**	0.3332 ± 0.1928	0.4883 ± 0.4791	0.7483 ± 0.2800	0.7967 ± 0.2418
	Country	0.2464 ± 0.3010	0.2558 ± 0.1609	**0.2462 ± 0.1689**	0.3141 ± 0.2840	0.4758 ± 0.5930	0.7778 ± 0.3170	0.9276 ± 0.1652
	Country/Purpose	0.2501 ± 0.1738	0.2442 ± 0.1741	**0.1982 ± 0.0973**	0.3502 ± 0.2185	0.4964 ± 0.4364	0.7277 ± 0.3154	0.7179 ± 0.3474
	Country/Purpose/State	0.2622 ± 0.1397	0.3463 ± 0.1467	**0.2463 ± 0.0958**	0.3382 ± 0.1226	0.4980 ± 0.3949	0.7503 ± 0.2210	0.7417 ± 0.2020
	Country/Purpose/State/CityNonCity	0.2628 ± 0.1259	0.3361 ± 0.1209	**0.2506 ± 0.1009**	0.3355 ± 0.1019	0.4824 ± 0.3619	0.7099 ± 0.1916	0.7545 ± 0.1712
Tourism-L	Overall	0.1611 ± 0.0283	**0.1379 ± 0.0359**	0.1928 ± 0.0712	2.1403 ± 3.2739	0.5658 ± 0.4548	0.8059 ± 0.3188	–
	Country	0.1490 ± 0.0266	**0.1069 ± 0.0293**	0.2107 ± 0.1372	2.9612 ± 4.6529	0.5593 ± 0.4326	0.9927 ± 0.4702	–
	Country/State	**0.1655 ± 0.0437**	0.1732 ± 0.0971	0.2121 ± 0.0591	2.4069 ± 3.6723	0.6204 ± 0.4471	0.8660 ± 0.4023	–
	Country/Purpose	0.1194 ± 0.0393	**0.0923 ± 0.0150**	0.1310 ± 0.0616	1.5607 ± 2.3729	0.4699 ± 0.4131	0.6829 ± 0.2489	–
	Country/State/Zone	0.2473 ± 0.0356	**0.2446 ± 0.0754**	0.2693 ± 0.0128	1.8104 ± 2.5870	0.7150 ± 0.5459	0.7160 ± 0.1470	–
	Country/State/Purpose	0.1671 ± 0.0416	**0.1603 ± 0.0398**	0.1766 ± 0.0258	1.2878 ± 1.8415	0.5938 ± 0.5359	0.6493 ± 0.2009	–
	Country/State/Zone/Region	0.2927 ± 0.0630	**0.2869 ± 0.0813**	0.2939 ± 0.0401	1.3583 ± 1.7821	0.8791 ± 0.8110	0.6255 ± 0.0909	–
	Country/State/Zone/Purpose	0.2621 ± 0.0441	**0.2591 ± 0.0585**	0.2631 ± 0.0336	1.0724 ± 1.3785	0.6333 ± 0.5345	0.5953 ± 0.1032	–
	Country/State/Zone/Region/Purpose	0.3239 ± 0.0786	0.3200 ± 0.0758	**0.3068 ± 0.0603**	0.8834 ± 0.9991	0.7089 ± 0.6174	0.5675 ± 0.1127	–
Labour	Overall	0.7228 ± 0.3399	0.5437 ± 0.0730	0.6361 ± 0.4770	0.5017 ± 0.1852	2.6825 ± 0.9036	1.7932 ± 0.3861	**0.1267 ± 0.1122**
	Country	0.7097 ± 0.6084	0.3401 ± 0.0984	0.4202 ± 0.3913	**0.1121 ± 0.0379**	2.5023 ± 0.8089	1.5036 ± 0.7674	0.1137 ± 0.1533
	Country/Region	0.6357 ± 0.1719	0.7197 ± 0.2446	0.9036 ± 0.7299	1.0138 ± 0.4723	3.5385 ± 1.4269	2.4094 ± 0.4165	**0.1043 ± 0.0618**
	Country/Gender/Region	0.7872 ± 0.0359	0.7541 ± 0.2400	0.8283 ± 0.5394	0.9019 ± 0.2554	3.0931 ± 1.3508	2.0540 ± 0.1406	**0.1430 ± 0.0647**
	Country/Employment/Gender/Region	0.8496 ± 0.1263	0.8571 ± 0.1918	0.9330 ± 0.3211	1.0026 ± 0.2482	2.1349 ± 0.3497	1.9132 ± 0.1060	**0.3087 ± 0.0669**
Wiki	Overall	0.5773 ± 0.4059	**0.5502 ± 0.4157**	1.3036 ± 0.4471	1.7208 ± 0.4582	2.6860 ± 1.8138	4.2882 ± 3.4797	3.1739 ± 3.9667
	Views	**0.4178 ± 0.4421**	0.4225 ± 0.4203	0.8950 ± 0.5363	1.2392 ± 0.3767	3.1036 ± 2.7122	4.5986 ± 3.5754	2.6987 ± 3.6913
	Views/Country	0.6192 ± 0.3742	**0.5879 ± 0.4249**	1.7442 ± 0.9584	2.2091 ± 1.8515	2.6871 ± 1.5101	5.5671 ± 5.4350	4.1029 ± 5.7199
	Views/Country/Access	0.7172 ± 0.4208	**0.6615 ± 0.4276**	1.5303 ± 0.5178	1.7314 ± 0.6301	2.0445 ± 1.0504	3.4253 ± 2.4928	2.8712 ± 2.9638
	Views/Country/Access/Agent	0.7163 ± 0.4029	**0.6580 ± 0.4053**	1.6597 ± 0.7462	1.7615 ± 0.7520	2.0425 ± 0.9206	3.5460 ± 2.6593	2.9662 ± 3.0873
	Views/Country/Access/Agent/Topic	0.7158 ± 0.3246	**0.6646 ± 0.3531**	1.5923 ± 0.7250	2.8580 ± 1.5160	3.1270 ± 2.0354	3.8338 ± 3.6443	4.3663 ± 5.3920
Favorita	Overall	0.7185 ± 0.0648	0.6482 ± 0.1544	**0.6232 ± 0.1027**	1.1795 ± 0.0658	1.1940 ± 0.1101	2.1179 ± 0.6558	0.7790 ± 0.0432
	Country	0.7704 ± 0.1297	0.6907 ± 0.2750	**0.6452 ± 0.1551**	1.3689 ± 0.0639	1.4049 ± 0.2341	2.4400 ± 1.1231	0.7573 ± 0.0474
	Country/State	0.6862 ± 0.0437	0.6205 ± 0.1172	**0.5989 ± 0.0780**	1.1141 ± 0.1124	1.1163 ± 0.0876	2.0652 ± 0.5563	0.8001 ± 0.0475
	Country/State/City	0.6865 ± 0.0510	0.6216 ± 0.1340	**0.5994 ± 0.0864**	1.0962 ± 0.1458	1.0924 ± 0.1177	2.0106 ± 0.5105	0.8005 ± 0.0455
	Country/State/City/Store	0.6511 ± 0.0266	**0.5970 ± 0.0544**	0.6313 ± 0.0257	0.7420 ± 0.0393	0.7463 ± 0.0174	1.2758 ± 0.4819	0.8947 ± 0.0624

*PERMBU-MinT requires a strict hierarchy and does not support grouped hierarchies like Tourism-L.

multivariate proper scoring rules that explicitly target spatial correlations: the Energy Score (ES) (Gneiting & Raftery, 2007) and the Variogram Score (VS) (Scheuerer & Hamill, 2015). Let $\mathbf{y}$ be a multivariate target vector (e.g., all series at a given hierarchy level) and let $\widetilde{\mathbf{Y}}$ be the corresponding multivariate forecast distribution with an independent copy $\widetilde{\mathbf{Y}}'$. The Energy Score with exponent 1 is defined as:

$$\mathrm{ES}(\mathbf{y}, \widetilde{\mathbf{Y}}) = \mathbb{E}\|\widetilde{\mathbf{Y}} - \mathbf{y}\|_2 - \frac{1}{2}\mathbb{E}\|\widetilde{\mathbf{Y}} - \widetilde{\mathbf{Y}}'\|_2, \tag{85}$$

where $\|\cdot\|_2$ denotes the Euclidean norm.

The Variogram Score with exponent 0.5 evaluates cross-sectional dependencies between series indices i and j:

$$\mathrm{VS}(\mathbf{y}, \widetilde{\mathbf{Y}}) = \sum_i \sum_j \left(|y_i - y_j|^{0.5} - \mathbb{E}\left[|\widetilde{Y}_i - \widetilde{Y}_j|^{0.5} \right] \right)^2. \tag{86}$$

Because no natural distance metric is defined between hierarchy nodes, we apply uniform weights to all pairs.

Following Marcotte et al. (2023), we report both metrics, as the VS is a powerful sample-based diagnostic for covariance errors that serves as a robust complement to CRPS and ES. Both expectations are approximated using the same $K = 200$–400 samples used for sCRPS, and we compute both scores at every hierarchy level as well as overall. Results are reported in Tables 14 and 15.

An HCFM variant attains the best ES and VS on 4 of 5 datasets at the leaf level, and on 5 of 5 when pooling all levels. Margins are largest on `Traffic`, where the U-Net variant improves the VS by $\sim 2\times$ over the best baseline ($2.97\text{e}3 \pm 3.11\text{e}3$ vs $5.94\text{e}3 \pm 5.63\text{e}3$) and the ES by $\sim 2.5\times$ (29.4 ± 12.1 vs 72.2 ± 45.2). On `Wiki` and `Labour`, HCFM is best under both scores at all levels. On `Tourism-L`, the leaf-level margins over CLOVER are within one window standard deviation and are regarded as ties, while the overall margins are distinct (ES $2.74\text{e}3 \pm 81$ vs $3.33\text{e}3 \pm 599$). CLOVER's one clear win is the leaf ES on `Tourism-S` ($1.60\text{e}3 \pm 232$ vs $1.75\text{e}3 \pm 168$), which is consistent with its Gaussian factor model explicitly parameterising bottom-level covariance.

The energy and variogram scores complement marginal sCRPS by evaluating multivariate forecast quality. The matched-backbone comparison in Table 11 examines the contribution of full-hierarchy dynamics.

Table 13: Comparison of the leaf-only HCFM-UNet configuration, the full soft-guided HCFM-UNet model, and N-HiTS (Bottom-Up). Both HCFM-UNet configurations use the same U-Net backbone. HCFM-UNet (Leaves Only) forecasts only the leaf series and obtains forecasts for all non-leaf nodes by bottom-up aggregation. HCFM-UNet (Full Soft) applies the complete soft-guided HCFM dynamics over all nodes in the hierarchy. N-HiTS (Bottom-Up) uses our probabilistic N-HiTS implementation to generate leaf samples and aggregates them sample-wise through the hierarchy (Appendix I.2). Each cell reports mean scaled CRPS (sCRPS) $\pm$ standard deviation across 3 rolling chained cross-validated windows. Lower is better. The best result in each row is determined from the mean and shown in bold; the reported standard deviations are descriptive and are not used as a tie-breaker.

Dataset	Level / Hierarchy	HCFM-UNet (Leaves Only)	HCFM-UNet (Full Soft)	N-HiTS (Bottom-Up)
Traffic	Overall	0.0324 ± 0.0089	$\mathbf{0.0245 \pm 0.0122}$	0.0911 ± 0.0252
	Level1	0.0189 ± 0.0085	$\mathbf{0.0125 \pm 0.0050}$	0.0485 ± 0.0270
	Level2	0.0190 ± 0.0082	$\mathbf{0.0128 \pm 0.0049}$	0.0505 ± 0.0257
	Level3	0.0225 ± 0.0078	$\mathbf{0.0172 \pm 0.0085}$	0.0580 ± 0.0168
	Level4	0.0690 ± 0.0131	$\mathbf{0.0554 \pm 0.0356}$	0.2075 ± 0.0323
Tourism-S	Overall	0.0864 ± 0.0093	$\mathbf{0.0808 \pm 0.0117}$	0.1238 ± 0.0494
	Country	$\mathbf{0.0388 \pm 0.0050}$	0.0403 ± 0.0198	0.1034 ± 0.0615
	Country/Purpose	0.0686 ± 0.0188	$\mathbf{0.0592 \pm 0.0143}$	0.1062 ± 0.0528
	Country/Purpose/State	0.1083 ± 0.0118	$\mathbf{0.1013 \pm 0.0100}$	0.1331 ± 0.0444
	Country/Purpose/State/CityNonCity	0.1299 ± 0.0089	$\mathbf{0.1223 \pm 0.0095}$	0.1526 ± 0.0394
Tourism-L	Overall	0.1467 ± 0.0108	$\mathbf{0.1366 \pm 0.0069}$	0.1570 ± 0.0137
	Country	0.0557 ± 0.0074	$\mathbf{0.0529 \pm 0.0066}$	0.0983 ± 0.0198
	Country/State	0.0831 ± 0.0114	$\mathbf{0.0787 \pm 0.0100}$	0.1075 ± 0.0166
	Country/Purpose	0.0949 ± 0.0143	$\mathbf{0.0825 \pm 0.0117}$	0.1144 ± 0.0210
	Country/State/Zone	0.1305 ± 0.0068	$\mathbf{0.1205 \pm 0.0078}$	0.1379 ± 0.0118
	Country/State/Purpose	0.1376 ± 0.0166	$\mathbf{0.1269 \pm 0.0107}$	0.1471 ± 0.0147
	Country/State/Zone/Region	0.1754 ± 0.0078	$\mathbf{0.1638 \pm 0.0053}$	0.1748 ± 0.0105
	Country/State/Zone/Purpose	0.2126 ± 0.0164	$\mathbf{0.1983 \pm 0.0084}$	0.2067 ± 0.0118
	Country/State/Zone/Region/Purpose	0.2840 ± 0.0180	$\mathbf{0.2695 \pm 0.0055}$	0.2697 ± 0.0100
Labour	Overall	0.0112 ± 0.0013	$\mathbf{0.0093 \pm 0.0003}$	0.0100 ± 0.0007
	Country	0.0073 ± 0.0030	$\mathbf{0.0049 \pm 0.0012}$	0.0062 ± 0.0001
	Country/Region	0.0101 ± 0.0011	$\mathbf{0.0081 \pm 0.0001}$	0.0086 ± 0.0006
	Country/Gender/Region	0.0111 ± 0.0009	$\mathbf{0.0095 \pm 0.0004}$	0.0098 ± 0.0010
	Country/Employment/Gender/Region	0.0161 ± 0.0006	$\mathbf{0.0145 \pm 0.0005}$	0.0156 ± 0.0011
Wiki	Overall	0.1877 ± 0.0849	$\mathbf{0.1838 \pm 0.0981}$	0.2399 ± 0.0842
	Views	$\mathbf{0.1001 \pm 0.0847}$	0.1121 ± 0.1115	0.1274 ± 0.0935
	Views/Country	0.1530 ± 0.0844	$\mathbf{0.1522 \pm 0.1058}$	0.2070 ± 0.0810
	Views/Country/Access	0.1983 ± 0.0866	$\mathbf{0.1892 \pm 0.0957}$	0.2523 ± 0.0888
	Views/Country/Access/Agent	0.2017 ± 0.0886	$\mathbf{0.1925 \pm 0.0969}$	0.2565 ± 0.0896
	Views/Country/Access/Agent/Topic	0.2853 ± 0.0809	$\mathbf{0.2728 \pm 0.0807}$	0.3562 ± 0.0738

I.4 Statistical Significance Tests

Our evaluation uses three non-overlapping windows per dataset (see Section 5 for details). While necessary due to high computational cost, it is insufficient to support per-dataset significance claims. We therefore conduct formal statistical tests following standard methodology for comparing forecast performance across multiple datasets (Demšar, 2006).

Across Methods. A Friedman test over the methods evaluated in Table 2, using datasets as blocks, rejects the null hypothesis of equal performance (6 methods $\times$ 6 datasets: $\chi^2 = 22.28$, $p = 0.00046$; including PERMBU-MinT on its 5 applicable datasets: $\chi^2 = 25.71$, $p = 0.0003$). HCFM-UNet ranks first with a mean rank of 1.33. CLOVER (2.33) and HierE2E (4.00) are the only competitive baselines. Because no other baseline achieves the best score on any dataset, we restrict our pairwise tests to these two alternatives.

Pairwise Tests. The paired analysis comprises 15 test segments: three windows each from `Labour`, `Tourism-S`, `Tourism-L`, `Traffic`, and `Wiki`. All methods share the same segments, and the HCFM variant is selected using validation sCRPS only. Table 16 reports 14/15 wins against HierE2E, with a median improvement of 25% (Holm-adjusted $p = 0.0031$), and 11/15 against CLOVER, with a median improvement of 17% (Holm-adjusted $p = 0.0151$). These results describe the five-dataset subset. Segment-level p-values are exploratory because windows within a dataset may be dependent; individual dataset rankings are descriptive.

Table 14: Empirical evaluation of Energy Score. HCFM-MQCNN and HCFM-UNet are the two backbone instantiations of our proposed model. CLOVER and HierE2E are the next best competitive end-to-end probabilistic hierarchical baselines. Each cell reports mean ± standard deviation across 3 rolling chained cross-validated windows, on the raw data scale, so scores are comparable within a row but not across rows. Lower values are better. The best method in each row is determined from the mean and shown in bold; the reported standard deviations are descriptive and are not used as a tie-breaker.

Dataset	HCFM-MQCNN	HCFM-UNet	CLOVER	HierE2E
Leaf level				
Labour	**46.6 ± 3.65**	50.4 ± 2.99	50.7 ± 7.27	56.5 ± 9.07
Tourism-S	1.75e+03 ± 168	1.78e+03 ± 219	**1.6e+03 ± 232**	1.82e+03 ± 296
Tourism-L	779 ± 14.2	**749 ± 16.8**	762 ± 56.8	866 ± 180
Traffic	32.7 ± 6.94	**10.3 ± 8.32**	17.1 ± 14.4	29.1 ± 5.41
Wiki	**1.03e4 ± 7.92e+03**	1.06e4 ± 7.7e+03	1.38e+04 ± 5.84e+03	1.83e+04 ± 6.21e+03
All levels pooled				
Labour	**94.5 ± 22.3**	110 ± 12.7	119 ± 24.4	119 ± 24.1
Tourism-S	4.68e+03 ± 1.52e+03	**4.66e+03 ± 1.11e+03**	4.73e+03 ± 874	5.29e+03 ± 1.75e+03
Tourism-L	3.12e+03 ± 306	**2.74e+03 ± 81**	3.33e+03 ± 599	5.33e+03 ± 2.67e+03
Traffic	121 ± 60.6	**29.4 ± 12.1**	72.2 ± 45.2	83.5 ± 26.2
Wiki	**2.59e+04 ± 2.28e+04**	2.71e4 ± 2.24e+04	3.10e+04 ± 9.19e+03	3.76e+04 ± 7.17e+03

Table 15: Empirical evaluation of Variogram Score. HCFM-MQCNN and HCFM-UNet are the two backbone instantiations of our proposed model. CLOVER and HierE2E are the next best competitive end-to-end probabilistic hierarchical baselines. Each cell reports mean ± standard deviation across 3 rolling chained cross-validated windows, on the raw data scale, so scores are comparable within a row but not across rows. Lower values are better. The best method in each row is determined from the mean and shown in bold; the reported standard deviations are descriptive and are not used as a tie-breaker.

Dataset	HCFM-MQCNN	HCFM-UNet	CLOVER	HierE2E
Leaf level				
Labour	**1.5e+03 ± 172**	1.52e+03 ± 166	1.56e+03 ± 276	2.07e+03 ± 493
Tourism-S	**1.86e5 ± 2.80e+04**	2.09e+05 ± 3.09e+04	1.90e+05 ± 2.40e+04	2.09e+05 ± 2.96e+04
Tourism-L	5.39e+06 ± 4.38e+05	**5.31e+06 ± 2.13e+05**	5.33e+06 ± 4.05e+05	6.81e+06 ± 1.78e+06
Traffic	1.24e+04 ± 2.74e+03	**2.85e+03 ± 3.02e+03**	5.27e+03 ± 5.71e+03	1.17e+04 ± 2.05e+03
Wiki	1.19e+07 ± 7.34e+06	**1.13e+07 ± 7.82e+06**	1.69e+07 ± 3.91e+06	2.61e+07 ± 5.31e+06
All levels pooled				
Labour	**3.78e+03 ± 632**	4.2e+03 ± 408	4.14e+03 ± 1.2e+03	5.99e+03 ± 1.29e+03
Tourism-S	**5.79e+05 ± 1.34e+05**	6.36e+05 ± 9.85e+04	5.98e+05 ± 2.85e+04	6.65e+05 ± 1.18e+05
Tourism-L	2.77e+07 ± 4.15e+06	**2.53e+07 ± 1.86e+06**	2.73e+07 ± 3.53e+06	4.50e+07 ± 2.38e+07
Traffic	1.38e+04 ± 3.52e+03	**2.97e+03 ± 3.11e+03**	5.94e+03 ± 5.63e+03	1.28e+04 ± 1.63e+03
Wiki	3.57e+07 ± 2.95e+07	**3.45e+07 ± 3.19e+07**	4.94e+07 ± 1.01e+07	6.36e+07 ± 9.68e+06

Table 16: Paired analysis across 15 test segments from five datasets. Wilcoxon p-values are Holm-adjusted; within-dataset window dependence limits their interpretation.

Comparison	Wins	Median improvement	W	Holm-adj p
HCFM vs HierE2E	14/15	25%	8	0.0031
HCFM vs CLOVER	11/15	17%	18	0.0151

Robustness. In the same five-dataset subset, aggregated differences favor HCFM against both baselines on 5/5 datasets (exact one-sided sign test $p = 0.031$). Table 17 reports leave-one-dataset-out segment-level tests, whose interpretation is limited by within-dataset window dependence. The p-values against CLOVER exceed 0.05 when `Tourism-L` is excluded. Individual dataset rankings remain descriptive.

Table 17: Leave-one-dataset-out segment-level tests within the five-dataset analysis. Within-dataset window dependence limits the interpretation of the p-values.

Excluded	vs HierE2E (Wins)	p	vs CLOVER (Wins)	p
Tourism-S	11/12	0.0049	9/12	0.0342
Traffic	11/12	0.0122	9/12	0.0425
Labour	11/12	0.0049	9/12	0.0342
Tourism-L	11/12	0.0093	8/12	0.0640
Wiki	12/12	0.0005	9/12	0.0068

J Conditional Flow Matching Background

Continuous-time flow models generate samples by integrating an ODE on a state matrix $\mathbf{X}(\tau)$,

$$\frac{d\mathbf{X}(\tau)}{d\tau} = \mathbf{v}_\theta(\tau, \mathbf{X}(\tau)), \qquad \tau \in [0,1], \qquad \mathbf{X}(0) \sim p_0, \tag{87}$$

where p_0 is a simple base distribution and $\mathbf{v}_\theta$ is a time-dependent vector field. Sampling is performed by numerically integrating Eq. (87) from $\tau = 0$ to $\tau = 1$.

Flow matching trains $\mathbf{v}_\theta$ by regressing it to a target velocity along a prescribed path between endpoint samples (Lipman et al., 2023). We use the deterministic linear bridge between endpoints. Endpoint pairs are sampled according to a coupling distribution $\pi(\mathbf{X}(0), \mathbf{X}(1))$, whose simplest instance is independent sampling, $\pi(\mathbf{X}(0), \mathbf{X}(1)) = p_0(\mathbf{X}(0))\, p_1(\mathbf{X}(1))$. Given $(\mathbf{X}(0), \mathbf{X}(1)) \sim \pi$, we sample $\tau \sim \mathcal{U}([0,1])$, set $\mathbf{X}(\tau) = (1-\tau)\mathbf{X}(0) + \tau\mathbf{X}(1)$, and use the constant target velocity $\mathbf{X}(1) - \mathbf{X}(0)$. The standard squared-loss objective for this bridge is

$$\mathcal{L}^{(2)}_{\mathrm{FM}}(\theta) = \mathbb{E}_{(\mathbf{X}(0),\mathbf{X}(1))\sim\pi,\,\tau}\Big[\, \big\|\mathbf{v}_\theta(\tau, \mathbf{X}(\tau)) - (\mathbf{X}(1) - \mathbf{X}(0))\big\|_F^2 \,\Big], \tag{88}$$

where $\|\cdot\|_F$ is the Frobenius norm on $\mathbb{R}^{D\times L}$ and the expectation is over $(\mathbf{X}(0), \mathbf{X}(1)) \sim \pi$ and $\tau \sim \mathcal{U}([0,1])$. This objective yields a simulation-free training procedure, while sampling remains a deterministic ODE solve.