NORMALIZING FLOW FILTER: OBSERVATION-ONLY LEARNING FOR NON-GAUSSIAN DATA ASSIMILATION

Anonymous authors
Paper under double-blind review

ABSTRACT

State filtering is central to data assimilation and model-based probabilistic forecasting in high-dimensional dynamical systems. Many scientific applications require filtering methods that can simultaneously (i) represent non-Gaussian posteriors, (ii) scale to high dimensions, (iii) learn from observations alone, and (iv) retain explicit transition and observation models for calibrating physical parameters and sensor biases. To directly address this regime, we introduce the Normalizing Flow Filter (NFF), an amortized variational filter with τ-step memory that learns an explicit non-Gaussian filtering density and calibrates the prescribed state-space model from observations alone. The training objective combines a recursive filtering evidence lower bound with the continuous ranked probability score on short-horizon future observations. For a fixed, correctly specified state-space model including the initial-state prior, exact τ-step memory, and sufficient flow expressivity, every global minimizer of the NFF loss averaged over data-generating observation sequences recovers the exact filtering posterior. NFF closely matches a 10^6-particle-filter reference for a four-dimensional coupled oscillator, remains reliable under highly ambiguous nonlinear observations in 40-dimensional Lorenz–96, and scales to joint state, physical-parameter, and sensor-bias inference in 16,384-dimensional Kolmogorov flow.

1 INTRODUCTION

Data assimilation (DA) infers the latent state of a dynamical system from partial and noisy observations under a prescribed state-space model (SSM). Filtering estimates the posterior over the current state using observations available up to the present, making it the natural target for the model-based probabilistic forecasting (e.g., Carrassi et al. 2018). In physical applications, the transition and observation models encode scientific knowledge. Keeping these models explicit enables direct calibration of their physical and sensor parameters from the same observations used to infer the state.

Nonlinear dynamics and observations make the filtering posterior non-Gaussian, creating a demanding problem regime. Representing this non-Gaussianity broadens the usable observation set: Gaussian DA workflows may downweight or screen informative measurements whose likelihood induces strongly non-Gaussian filtering posteriors (Anderson & Järvinen, 1999; Storto, 2016; Geer et al., 2019). High state dimension and repeated online use demand scalable inference, while unavailable latent trajectories preclude direct state supervision.

We frame this DA challenge through four coupled requirements: (i) representing non-Gaussian and potentially multimodal posteriors, (ii) scaling to high-dimensional physical systems, (iii) learning from observations alone, and (iv) retaining the prescribed SSM for parameter calibration. Generating supervised state trajectories requires choosing physical parameters and model-error statistics, or their sampling distributions, even when these quantities are themselves calibration targets. The challenge is therefore to learn a reusable filtering distribution while calibrating its prescribed generative model from the same observations.

Existing methods offer different compromises. Ensemble Kalman filter (EnKF) uses covariance-based analysis approximations (Evensen, 1994), while weight degeneracy remains a major challenge for particle filters (PFs) in high-dimensional problems (van Leeuwen et al., 2019). Learned filters such as Cui et al. (2026) obtain flexible densities from simulated state-observation pairs. Observation-only approaches such as Amortized Ensemble Filter (AmEnF; McCabe & Brown 2021)

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.

instead learn ensemble analysis updates through a forecast loss. Section 4 discusses these and other non-Gaussian and variational methods.

We develop the Normalizing Flow Filter (NFF), an amortized filtering method targeting precisely this regime. NFF learns an explicit non-Gaussian density over the current physical state, using the prescribed transition and observation densities to constrain posterior learning and calibrate model parameters from observations alone. Shared filtering and backward flows are trained with a recursive ELBO based on the AELBO-2 filtering/backward factorization of Campbell et al. (2021), complemented by the continuous ranked probability score (CRPS; Gneiting & Raftery 2007) on future observations. We establish sufficient conditions for exact filter recovery at global optima of the expected training loss, both with and without predictive CRPS (Sec. 3.3).

Contributions.

- **DA-tailored formulation.** We develop NFF around the four coupled requirements of prescribed-model DA, providing a non-Gaussian physical-state filtering density and model calibration from observations alone. Its shared inference model avoids per-window optimization at deployment.
- **Theoretical characterization.** We derive an exact expected-gap decomposition for the recursive ELBO into predictive and posterior KL divergences (Sec. 3.3, Appendix A). Under the stated specification, memory, and realizability assumptions, global optima recover the exact filter for fixed $\theta = \theta^\star$. With learned θ, they recover identifiable parameters and the exact filter under an observationally equivalent candidate model. Future-observation CRPS preserves these optima (Appendix B). For a fixed, correctly specified SSM, we further bound the attainable finite-window recursive-ELBO gap under quantitative forgetting assumptions (Appendix C).
- **Distributional evaluation.** We test posterior recovery against a large-particle reference in four dimensions, assess density accuracy and supplementary-observation conditioning in 40-dimensional Lorenz-96, and demonstrate joint filtering and model calibration in 16,384-dimensional Kolmogorov flow. The ablations show that strong marginal scores or accurate parameter estimates can coexist with poor filtering distributions (Secs. 5.1, 5.2, 5.3).

2 Problem Setting

We consider a first-order Markov SSM

$$p_\theta(X_{1:T}, Y_{1:T}) = p_0(X_1) \prod_{t=2}^{T} p_\theta^{\mathrm{dyn}}(X_t \mid X_{t-1}) \prod_{t=1}^{T} p_\theta^{\mathrm{obs}}(Y_t \mid X_t), \tag{1}$$

where $X_t \in \mathbb{R}^d$ is the physical state, $Y_t \in \mathbb{R}^p$ is the observation, and θ collects physical and sensor parameters, some of which may be unknown. The distributions p_0, p_θ^{dyn}, and p_θ^{obs} are specified parametrically and evaluable. Any dependence of p_0 on θ is omitted from the notation in equation 1 and subsequent expressions. Given observation sequences $\{Y_{1:T}^{(n)}\}_{n=1}^N$, we consider:

Problem setting.
Provided: an evaluable parametric SSM $(p_0, p_\theta^{\mathrm{dyn}}, p_\theta^{\mathrm{obs}})$ and observation sequences $\{Y_{1:T}^{(n)}\}_{n=1}^N$.
Goal: approximate the filtering distribution $p_\theta(X_t \mid Y_{1:t})$ for all t and, when applicable, estimate the unknown parameters θ.
The latent state sequence $\{X_{1:T}^{(n)}\}_{n=1}^N$ is **not** available for training.

For memory length τ, let $W_t = (\bar{Y}_t, M_t)$, where $\bar{Y}_t$ is the right-aligned, zero-padded window $Y_{\max\{1,t-\tau+1\}:t}$ and M_t has ones at observed entries and zeros elsewhere.

3 Method

NFF approximates the two-slice filtering posterior using filtering and backward conditional flows, trained on observations alone with a recursive ELBO and short-horizon future-observation CRPS.

3.1 TWO-FLOW REPRESENTATION OF THE FILTERING POSTERIOR

The exact filtering update couples two adjacent states:

$$p_\theta(X_t, X_{t-1} \mid Y_{1:t}) \propto p_\theta^{\mathrm{obs}}(Y_t \mid X_t) p_\theta^{\mathrm{dyn}}(X_t \mid X_{t-1}) p_\theta(X_{t-1} \mid Y_{1:t-1}). \tag{2}$$

NFF represents its two-slice variational approximation as

$$q_{\phi,t}(X_t, X_{t-1}) = q_\phi^F(X_t \mid W_t)\, q_\phi^B(X_{t-1} \mid X_t, W_{t-1}). \tag{3}$$

The backward flow parameterizes the predecessor conditional in the two-slice variational posterior and completes the density required by the recursive filtering ELBO. It is used during training; filtering a new observation window requires only q_ϕ^{F}. Conditioning the backward flow on W_{t-1} matches the exact reverse conditional, whose dependence on past data enters through the preceding filtering distribution. Both factors are represented by conditional normalizing flows. For a context c, an invertible map $X = g_\omega(Z; c)$ with $Z \sim \mathcal{N}(0, I)$ defines

$$\log q_\omega(X \mid c) = \log p_Z(Z) - \log \left| \det \frac{\partial g_\omega(Z; c)}{\partial Z} \right|. \tag{4}$$

We use conditional RealNVP flows (Dinh et al., 2017). The context encoders and coupling architectures follow the spatial geometry of each experiment and are detailed in Appendix D. This construction provides both reparameterized samples and the log densities required by the ELBO.

3.2 TRAINING OBJECTIVE

3.2.1 RECURSIVE FILTERING ELBO AND BOUNDARY TERM

We begin from the observation evidence under the candidate SSM:

$$\log p_\theta(Y_{1:T}) = \log p_\theta(Y_1) + \sum_{t=2}^{T} \log p_\theta(Y_t \mid Y_{1:t-1}). \tag{5}$$

For $t \geq 2$, replacing the preceding exact filter by q_ϕ^F induces the observation predictor

$$\widetilde{p}_{\theta,\phi,t}(Y_t \mid W_{t-1}) = \iint p_\theta^{\mathrm{obs}}(Y_t \mid X_t) p_\theta^{\mathrm{dyn}}(X_t \mid X_{t-1}) q_\phi^F(X_{t-1} \mid W_{t-1})\, \mathrm{d}X_{t-1} \mathrm{d}X_t. \tag{6}$$

The adjacent-state posterior under the induced predictor is

$$\widetilde{\pi}_{\theta,\phi,t}(X_t, X_{t-1}) = \frac{p_\theta^{\mathrm{obs}}(Y_t \mid X_t) p_\theta^{\mathrm{dyn}}(X_t \mid X_{t-1}) q_\phi^F(X_{t-1} \mid W_{t-1})}{\widetilde{p}_{\theta,\phi,t}(Y_t \mid W_{t-1})}, \tag{7}$$

where its conditioning on (Y_t, W_{t-1}) is suppressed. Using $q_{\phi,t}$ as a variational approximation to $\widetilde{\pi}_{\theta,\phi,t}$ yields

$$\begin{aligned} \mathcal{L}_t(\theta, \phi) &= \mathbb{E}_{q_{\phi,t}} \Big[\log p_\theta^{\mathrm{obs}}(Y_t \mid X_t) + \log p_\theta^{\mathrm{dyn}}(X_t \mid X_{t-1}) + \log q_\phi^F(X_{t-1} \mid W_{t-1}) \\ &\qquad - \log q_\phi^F(X_t \mid W_t) - \log q_\phi^B(X_{t-1} \mid X_t, W_{t-1}) \Big] \\ &= \log \widetilde{p}_{\theta,\phi,t}(Y_t \mid W_{t-1}) - \mathrm{KL}\left(q_{\phi,t} \,\|\, \widetilde{\pi}_{\theta,\phi,t}\right). \end{aligned} \tag{8}$$

Thus, for each fixed observed sequence $Y_{1:t}$, $L_t(\theta, \phi) \leq \log \widetilde{p}_{\theta,\phi,t}(Y_t \mid W_{t-1})$. This bound is on the predictor induced by the preceding amortized filter q_ϕ^F, not generally on the exact model predictor $p_\theta(Y_t \mid Y_{1:t-1})$. At $t = 1$, no preceding state or backward conditional is introduced:

$$\mathcal{L}_1(\theta, \phi) = \mathbb{E}_{q_\phi^F(X_1 \mid W_1)} \left[\log p_\theta^{\mathrm{obs}}(Y_1 \mid X_1) + \log p_0(X_1) - \log q_\phi^F(X_1 \mid W_1) \right]. \tag{9}$$

The recursive objective for one observation sequence is

$$\mathcal{L}_{\mathrm{NFF}}(\theta, \phi; Y_{1:T}) = \mathcal{L}_1(\theta, \phi) + \sum_{t=2}^{T} \mathcal{L}_t(\theta, \phi). \tag{10}$$

For $t < \tau$, the same masked-window representation is used without introducing artificial observations or a state X_0.

3.2.2 FUTURE-OBSERVATION CRPS

Reverse-KL optimization may discard separated posterior alternatives under finite model capacity and non-convex optimization. We therefore add a predictive CRPS computed purely from observed quantities. For each valid forecast origin t and horizon h, we draw K samples of $X_t^{(k)} \sim q_\phi^F(\cdot \mid W_t)$ independently and generate rollouts through p_θ^{dyn} and p_θ^{obs}, including process and observation noise across the sampled members. For scalar observation coordinate j, we use the fair finite-ensemble estimator of Ferro (2014):

$$\widehat{\mathrm{CRPS}}_K = \frac{1}{K}\sum_{k=1}^{K}\left|\widetilde{Y}_{t+h,j}^{(k)} - Y_{t+h,j}\right| - \frac{1}{K(K-1)}\sum_{k<\ell}\left|\widetilde{Y}_{t+h,j}^{(k)} - \widetilde{Y}_{t+h,j}^{(\ell)}\right|. \tag{11}$$

A nonnegative weighted sum over valid forecast origins, horizons, and observed coordinates defines $\widehat{R}_{\mathrm{CRPS}}$; experiment-specific normalizations are given in Appendix D.

$$\widehat{\mathcal{J}}(\theta,\phi) = -\widehat{\mathcal{L}}_{\mathrm{NFF}}(\theta,\phi) + \lambda_{\mathrm{CRPS}}\widehat{\mathcal{R}}_{\mathrm{CRPS}}(\theta,\phi). \tag{12}$$

CRPS supplies a predictive-marginal coverage signal, while the recursive ELBO constrains the dynamics-consistent adjacent-state distribution.

3.3 EXPECTED GAP AND FILTER IDENTIFICATION

Let $p^\star$ denote the data-generating joint distribution of $(X_{1:T}, Y_{1:T})$ and $\theta^\star$ its SSM parameter under correct specification. Averaging over observation sequences gives

$$\begin{aligned}&\mathbb{E}_{p^\star}\left[\log p^\star(Y_t \mid Y_{1:t-1}) - \mathcal{L}_t(\theta,\phi)\right]\\ &= \mathbb{E}_{p^\star(Y_{1:t-1})}\mathrm{KL}\left[p^\star(Y_t \mid Y_{1:t-1}) \,\|\, \widetilde{p}_{\theta,\phi,t}(Y_t \mid W_{t-1})\right] + \mathbb{E}_{p^\star(Y_{1:t})}\mathrm{KL}\left[q_{\phi,t} \,\|\, \widetilde{\pi}_{\theta,\phi,t}\right].\end{aligned} \tag{13}$$

The first KL measures one-step predictive error; for $\theta = \theta^\star$ and exact τ-step memory, it isolates the error from replacing the preceding exact filter by q_ϕ^F. The second is the ELBO gap under the induced predictor. At $t = 1$,

$$\begin{aligned}&\mathbb{E}_{p^\star}\left[\log p^\star(Y_1) - \mathcal{L}_1(\theta,\phi)\right]\\ &= \mathrm{KL}\left[p^\star(Y_1) \,\|\, p_\theta(Y_1)\right] + \mathbb{E}_{p^\star(Y_1)}\mathrm{KL}\left[q_\phi^F(X_1 \mid W_1) \,\middle\|\, p_\theta(X_1 \mid Y_1)\right].\end{aligned} \tag{14}$$

Summing equation 13 and equation 14 yields

$$\Delta_{\mathrm{NFF}}(\theta,\phi) := \mathbb{E}_{p^\star}\left[\log p^\star(Y_{1:T}) - \mathcal{L}_{\mathrm{NFF}}(\theta,\phi;Y_{1:T})\right] \geq 0. \tag{15}$$

Assume correct model specification, including the initial prior, exact τ-step memory under the forgetting assumption, and a shared recognition family that simultaneously contains the exact filtering and backward conditionals on all relevant inputs. With fixed $\theta = \theta^\star$, the exact recursion makes every term in equation 15 zero. Conversely, every global maximizer must make all terms zero. Induction from the prior-anchored $t = 1$ term then gives

$$q_\phi^F(X_t \mid W_t) = p^\star(X_t \mid Y_{1:t}), \qquad q_\phi^B(X_{t-1} \mid X_t, W_{t-1}) = p^\star(X_{t-1} \mid X_t, Y_{1:t-1}) \tag{16}$$

for every t on inputs generated by the corresponding distributions. If point-valued θ is also optimized, zero expected gap instead requires $p_\theta(Y_{1:T}) = p^\star(Y_{1:T})$ and exact filtering and backward conditionals under the candidate model. Consequently, every observationally identifiable component of θ equals its data-generating value, while unidentifiable components are determined only up to observational equivalence.

Finally, let $\mathcal{R}_{\mathrm{CRPS}}(\theta,\phi)$ be the expected future-observation CRPS and let $\Delta_{\mathrm{CRPS}}(\theta,\phi)$ denote its difference from the score of the true conditional predictive distributions. Strict propriety of CRPS implies $\Delta_{\mathrm{CRPS}}(\theta,\phi) \geq 0$. For $\lambda_{\mathrm{CRPS}} \geq 0$, define

$$\mathcal{J}_\lambda(\theta,\phi) = -\mathbb{E}_{p^\star}\mathcal{L}_{\mathrm{NFF}}(\theta,\phi;Y_{1:T}) + \lambda_{\mathrm{CRPS}}\mathcal{R}_{\mathrm{CRPS}}(\theta,\phi). \tag{17}$$

For an exact data-generating solution $(\theta^\star, \phi^\star)$,

$$\mathcal{J}_\lambda(\theta,\phi) - \mathcal{J}_\lambda(\theta^\star,\phi^\star) = \Delta_{\mathrm{NFF}}(\theta,\phi) + \lambda_{\mathrm{CRPS}}\Delta_{\mathrm{CRPS}}(\theta,\phi) \geq 0. \tag{18}$$

Thus, an exact data-generating solution is a global minimizer of the combined objective. Appendix B further shows that the combined objective has the same global-minimizer set as the expected recursive-ELBO loss. The fair estimator in equation 11 is unbiased for the corresponding CRPS. Full derivations are given in Appendices A and B. Appendix C upper-bounds the best achievable finite-window recursive-ELBO gap under approximate forgetting, with θ fixed at $\theta^\star$.

4 Related Work

Filtering with prescribed physical models. Classical DA keeps the physical state and prescribed transition and observation models explicit. Kalman Filter and EnKF provide scalable recursive updates, but use Gaussian or covariance-based analysis approximations (Kalman, 1960; Evensen, 1994). 4D-Var instead computes a MAP trajectory for each assimilation window (Dimet & Talagrand, 1986; Lewis & Derber, 1985). PFs target general non-Gaussian posteriors, but their weights degenerate rapidly with dimension; particle-flow and ensemble-transport methods mitigate this by moving finite ensembles toward the analysis distribution (van Leeuwen et al., 2019; Hu & van Leeuwen, 2021). EnSF (Bao et al., 2024) and EnFF (Transue et al., 2025) use training-free score- and flow-matching-based analysis updates, respectively, for high-dimensional nonlinear filtering. These methods ordinarily return point estimates, Gaussian analyses, or empirical ensembles through per-update computation. NFF instead amortizes an evaluable non-Gaussian physical-state density while retaining the prescribed SSM.

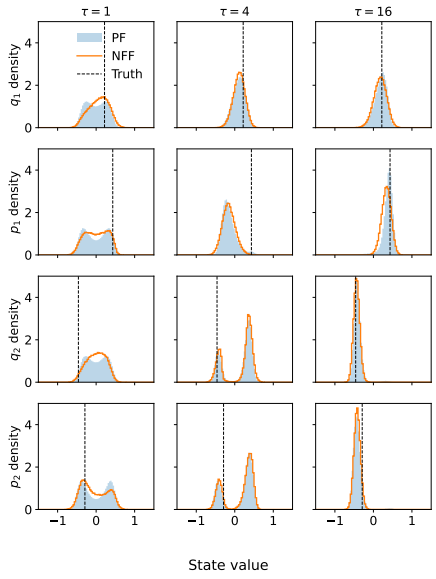
Sequential variational inference and flow-based filtering. Recursive variational objectives replace the preceding filter with an approximation, using factorized adjacent-state (Zhao & Park, 2020) or filtering/backward constructions (TD-VAE; Gregor et al. 2019), termed AELBO-1 and AELBO-2 by Campbell et al. (2021) and distinguished from the full-sequence ELBO. eVKF (Dowling et al., 2023) provides exponential-family variational filtering for joint state and dynamics learning, whereas FIVO (Maddison et al., 2017) and VSMC (Naesseth et al., 2018) use particle-based sequential bounds. NFF builds on the AELBO-2 filtering/backward factorization to construct a finite-window, explicit-density amortized filter for non-Gaussian prescribed-model DA. We derive the exact expected-gap decomposition of this shared recursive objective. Normalizing Kalman filters (de Bézenac et al., 2020) and FBF (Wang et al., 2025) learn transformations into latent linear-Gaussian SSMs; FBF does not retain a prescribed physical SSM. Flows also support VI with prescribed physical models (Xie et al., 2025). Cui et al. (2026) combine filtering and backward flows but trains by maximum likelihood on simulated state-observation trajectories, addressing a different information regime from NFF's observation-only prescribed-model setting. PR-Smoother (Tarumi, 2026) trains a trajectory posterior with a sequence ELBO and Gaussian forward conditionals, establishing posterior inclusiveness in deterministic and linear-Gaussian limits. NFF's filtering and backward flows remove this Gaussian conditional restriction. For a fixed, correctly specified SSM, every global minimizer of its expected recursive-ELBO loss recovers the exact filter under finite-window sufficiency and simultaneous realizability, even with noisy nonlinear dynamics. PSEF (Bach et al., 2026) learns an analysis ensemble using a strictly proper energy score against simulated latent states, proving filter recovery under realizability. NFF instead combines short-horizon future-observation CRPS without latent-state targets (Gneiting & Raftery, 2007; Ferro, 2014) with a recursive ELBO constraining the latent adjacent-state distribution.

Learned updates, priors, and latent-space data assimilation. Some learned DA methods retain the forecast model while learning inverse maps, gains, or noise parameters (Frerix et al., 2021; Revach et al., 2021; Xu & Niu, 2024). Others learn the variational model, analysis operator, or solver itself (Fablet et al., 2021; Boudier et al., 2023). These methods can reduce online cost, but typically retain a point, Gaussian, or ensemble representation, or learn an update without exposing a normalized analysis density. ROAD-EnKF (Chen et al., 2023) learns latent surrogate dynamics and a decoder from noisy observations through a differentiable EnKF, whereas NFF retains and calibrates the prescribed physical transition model. A complementary family learns a trajectory prior, transition surrogate, score model, or low-dimensional representation from simulated physical states and then conditions it on observations. Examples include score-based DA, FlowDAS, reduced-order ensemble filters, latent-space DA, and DBF (Rozet & Louppe, 2023; Chen et al., 2025; Peyron et al., 2021; Tarumi et al., 2025). LEVDA performs per-window 4DEnVar-style optimization in the ensemble subspace of a pretrained differentiable latent-dynamics surrogate, supporting irregular observations and joint state-parameter inference (Si & Chen, 2026). Latent-EnSF and LD-EnSF perform score-based filtering in learned latent spaces, with LD-EnSF also learning latent dynamics and jointly assimilating states and system parameters under sparse observations (Si & Chen, 2025; Xiao et al., 2026). These approaches suit settings with representative state trajectories over a trusted parameter range. NFF instead trains from the observations used for calibration, without fixing a parameter distribution and model-error statistics to generate supervised trajectories.

5 EXPERIMENTS

Experimental design. We test three claims: faithful recovery of multimodal filtering distributions in a 4D coupled oscillator (Sec. 5.1), stable inference in 40D Lorenz–96 when ambiguous absolute-value observations and strong process noise leave the filtering posterior genuinely multimodal (Sec. 5.2), and scalability to 16,384-dimensional Kolmogorov flow (Sec. 5.3). All training uses observations $Y_{1:T}$ and the prescribed models ($f_{\theta_{dyn}}$ and $h_{\theta_{obs}}$). For Lorenz–96 and Kolmogorov flow, we also compare with the AmEnF (McCabe & Brown, 2021), which learns ensemble analysis updates using an observation-only forecast loss. Appendix D documents implementation and computational costs; on an NVIDIA GH200, batch-one generation of 128 filtering samples takes 9.44 ms per Lorenz–96 cycle and 331 ms per Kolmogorov-flow cycle. Unless specified otherwise, results are averaged over five independent seeds, and $\pm$ gives one standard deviation.

5.1 FILTERING UNDER SIGN AMBIGUITY: COUPLED OSCILLATORS



τ	Method	ED↓	W_1 ↓
1	NFF	0.136 ± 0.017	0.131 ± 0.015
	EnKF	0.297 ± 0.002	0.502 ± 0.002
	PFF	$\mathbf{0.027 \pm 0.000}$	$\mathbf{0.112 \pm 0.002}$
	EnSF	0.029 ± 0.001	0.130 ± 0.001
4	NFF	$\mathbf{0.029 \pm 0.010}$	$\mathbf{0.040 \pm 0.008}$
	EnKF	0.796 ± 0.019	0.570 ± 0.020
	PFF	0.528 ± 0.004	0.291 ± 0.001
	EnSF	0.575 ± 0.005	0.271 ± 0.001
16	NFF	$\mathbf{0.003 \pm 0.001}$	$\mathbf{0.012 \pm 0.001}$
	EnKF	0.298 ± 0.062	0.154 ± 0.024
	PFF	0.383 ± 0.007	0.234 ± 0.020
	EnSF	0.869 ± 0.013	0.373 ± 0.003

ED: energy distance in $\mathbb{R}^4$.
W_1: mean marginal Wasserstein distance.

Figure 1: Coupled-oscillator filtering. Marginals of (q_1, p_1, q_2, p_2) (rows) for $\tau = 1, 4, 16$ (columns), compared with the common 10^6-particle windowed PF reference. Dashed lines mark the simulated state. The table reports joint energy distance and mean marginal W_1.

We first study a four-dimensional oscillator system, permitting direct distributional comparison with a large-particle PF reference. Let $X_t = (q_{1,t}, p_{1,t}, q_{2,t}, p_{2,t}) \in \mathbb{R}^4$, and let Φ_Δ denote one numerical step of two coupled, weakly nonlinear oscillators. The SSM is

$$X_1 \sim \mathcal{N}(0, I_4), \qquad X_t = \Phi_\Delta(X_{t-1}) + \eta_t, \qquad Y_t = X_t \odot X_t + \epsilon_t, \tag{19}$$

where $\Delta = 0.5$ and $\eta_t, \epsilon_t \sim \mathcal{N}(0, 0.1^2 I_4)$ independently. The componentwise square leaves coordinatewise signs ambiguous, whereas the coupled, forced dynamics are not invariant to arbitrary sign flips. A temporal window can therefore reject sign branches that do not form a plausible trajectory.

We train separate NFF models with $\tau \in \{1, 4, 16\}$ from observations alone. At $T = 128$, all methods use only $Y_{T-\tau+1:T}$ and are evaluated against the same 10^6-particle PF reference, computed over that observation window with prior $\mathcal{N}(0, I_4)$ at $T - \tau + 1$. We additionally compare with a 1000-member EnKF, a particle flow filter (PFF), and an ensemble score filter (EnSF). Figure 1 shows one prespecified held-out sequence, while the table averages results over ten held-out sequences and five independent method seeds. Energy distance (ED) measures the joint discrepancy in $\mathbb{R}^4$, while W_1 is averaged over the four coordinatewise marginals.

In the displayed sequence, NFF captures the remaining separated modes at $\tau = 4$ and closely matches the concentrated PF marginals at $\tau = 16$. Across the ten held-out sequences, NFF achieves

the lowest mean joint ED and marginal W_1 at both $\tau = 4, 16$. At $\tau = 1$, PFF and EnSF achieve lower joint ED. This may partly reflect a mismatch with the reference's reset prior: predictive CRPS favors data-generating future-observation marginals, which can differ from those implied by the reference posterior (Appendix D.1).

Takeaways. NFF represents unresolved sign alternatives and closely matches the prescribed-prior PF reference at $\tau = 4, 16$, without latent-state or PF supervision.

5.2 LORENZ96, 40D

Table 1: Nonlinear Lorenz–96 filtering (five-seed mean ± standard deviation). E/C denote recursive ELBO/future-observation CRPS; F/G denote flow/diagonal Gaussian. The final column scores coordinates not directly measured by supplementary sensors. Best means are boldfaced. The supplementary update reflects both posterior quality and finite-sample reweighting effects.

	Marginal posterior		Joint density and update	
Method	State CRPS ↓	Calib. error [pp] ↓	Filter NLL/dim ↓	Unobserved state CRPS after additional obs. ↓
NFF				
E+C, F	0.829 ± 0.004	1.252 ± 0.160	$\mathbf{1.602 \pm 0.005}$	$\mathbf{0.814 \pm 0.004}$
E+C, G	0.993 ± 0.026	1.715 ± 0.182	3.156 ± 0.101	0.986 ± 0.036
E, F	3.352 ± 0.155	18.974 ± 0.576	15.672 ± 0.675	3.567 ± 0.201
C, F	$\mathbf{0.823 \pm 0.003}$	7.476 ± 0.556	$(7.534 \pm 3.384) \times 10^{11}$	0.876 ± 0.007
EnKF	1.746 ± 0.053	7.044 ± 0.332	N/A	2.041 ± 0.086
PFF	2.246 ± 0.004	14.216 ± 0.017	N/A	2.223 ± 0.021
EnSF	2.532 ± 0.004	$\mathbf{0.536 \pm 0.021}$	N/A	3.137 ± 0.024
AmEnF	6.496 ± 0.004	42.684 ± 0.075	N/A	6.489 ± 0.027

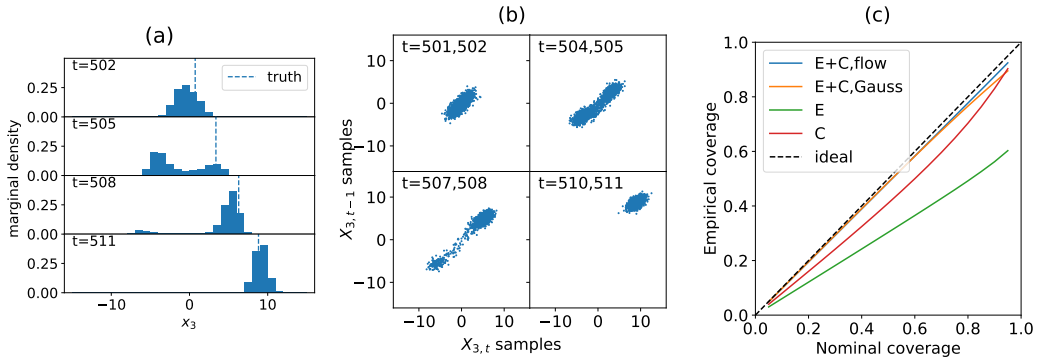


Figure 2: Lorenz–96 with $h(x) = |x|$. (a) NFF marginals of $X_{3,t}$ at four selected times, with $\tau = 16$; dashed lines mark truth. (b) Corresponding adjacent-state samples from $q^F q^B$. (c) Marginal interval coverage for the four ablations.

We experiment with 40-dimensional Lorenz–96 with forcing $F = 8$, observation interval $\Delta t = 0.03$, and independent Gaussian process noise of standard deviation 1 per coordinate and interval. All coordinates are observed through $Y_t = h(X_t) + \epsilon_t$, with $\epsilon_t \sim \mathcal{N}(0, I_{40})$ and either $h(x) = x$ or $h(x) = |x|$. We focus here on the sign-ambiguous nonlinear case. The linear results (Appendix, Table 8) provide an implementation sanity check, with most methods achieving low marginal state CRPS among successful runs. We have four NFF variants: flow and diagonal-Gaussian models trained with ELBO+CRPS, and ELBO-only and CRPS-only flow models using $T = 512$ and $\tau = 16$. Forcing and both noise levels remain fixed during training. Marginal scores and filtering NLL use every time $t = 257, \ldots, 512$. Implementation details appear in Appendix D.2.

Compared with the ELBO+CRPS flow, CRPS-only slightly improves state CRPS but worsens calibration and filtering NLL. The extreme NLL persists in FP64 subset checks despite precision sensitivity (Appendix E.3), showing that strong marginal scores can coexist with an inaccurate joint filtering density. ELBO-only instead produces poor marginal accuracy and severe undercoverage, both substantially improved by adding predictive CRPS. With ELBO+CRPS training, replacing the flow with a diagonal Gaussian worsens every reported metric, supporting the use of an expressive posterior family under sign ambiguity.

At $T = 512$, we supplement the original 40 absolute-value measurements with ten signed linear measurements (noise standard deviation 3), withheld from the encoder. We assimilate them by likelihood reweighting of filtering samples and report CRPS over the remaining 30 coordinates (Table 1, final column). The ELBO+CRPS flow achieves the lowest post-update CRPS. This tests information transfer through the filtering distribution, complementing direct NLL evaluation.

Figure 2(a,b) illustrates separated marginal modes and adjacent-state dependence in the ELBO+CRPS flow, while panel (c) shows reduced undercoverage after adding CRPS. The ELBO+CRPS flow also outperforms all three classical filters and AmEnF in state and post-update CRPS. EnSF nevertheless achieves the lowest calibration error, emphasizing that near-nominal coverage alone is insufficient to establish distributional accuracy.

Takeaways. (i) NFF represents separated posterior modes in a 40-dimensional state space. (ii) CRPS-only training achieves low state CRPS but yields an inaccurate joint filtering density and less reliable conditioning on additional observations. (iii) The ELBO+CRPS flow combines competitive state recovery with favorable joint-density and conditioning diagnostics.

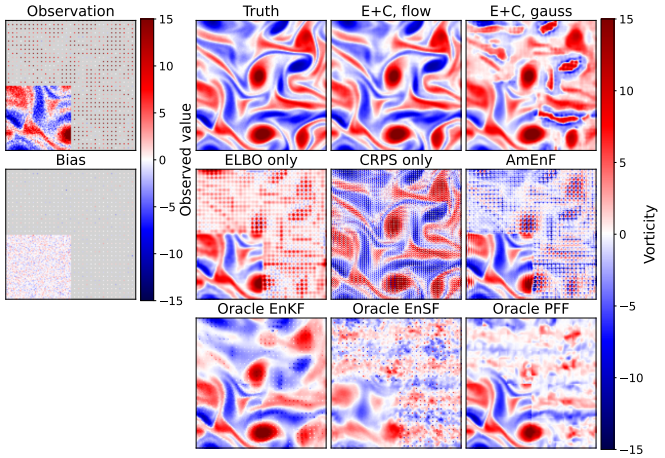
5.3 KOLMOGOROV FLOW

Table 2: Kolmogorov filtering with joint parameter and bias learning. Notation follows Table 1; classical filters (EnKF, EnSF, PFF) assume true parameters and biases. The final column excludes the four supplementary sensor locations and uses likelihood reweighting with 4,096 NFF samples, 128 AmEnF members, or 32 members per classical filter; it reflects both posterior quality and finite-sample effects.

	Marginal posterior		Joint density and update	
Method	State CRPS ↓	Calib. error [pp] ↓	Filter NLL/dim ↓	Unobserved state CRPS after additional obs. ↓
NFF				
E+C, F	$\mathbf{0.231 \pm 0.007}$	4.627 ± 0.953	$\mathbf{1.696 \pm 0.411}$	$\mathbf{0.232 \pm 0.007}$
E+C, G	2.404 ± 0.268	43.531 ± 0.567	129.450 ± 23.867	2.407 ± 0.269
E, F	2.639 ± 0.031	42.910 ± 0.191	180.214 ± 176.003	2.658 ± 0.014
C, F	2.068 ± 0.252	24.652 ± 4.492	$(7.756 \pm 4.202) \times 10^5$	2.213 ± 0.262
EnKF	1.003 ± 0.047	5.930 ± 4.318	N/A	2.826 ± 1.874
PFF	1.157 ± 0.005	$\mathbf{1.216 \pm 0.042}$	N/A	2.022 ± 0.056
EnSF	2.284 ± 0.012	11.611 ± 0.084	N/A	5.413 ± 0.114
AmEnF	2.399 ± 0.075	28.689 ± 3.936	N/A	2.562 ± 0.084

We consider Kolmogorov flow on a 128×128 grid, jointly learning the filtering distribution, forcing amplitude F_0, log Reynolds number $\log \mathrm{Re}$, and sensor biases from observations alone. A 64×64 dense region is observed linearly, while stride-four sensors elsewhere measure x^2, leaving signs ambiguous (Figure 3, "Observation" panel). We use $T = 512$, $\tau = 64$, and the same reporting interval as in Lorenz–96. The classical filters (EnKF, EnSF, PFF) assimilate all 512 observations using the true physical parameters and sensor biases. Appendix D.3 reports implementation details and sensitivity studies (Figs. 6–8).

Figure 3 shows that the ELBO+CRPS flow reconstructs coherent vortical structures beyond the densely observed region, despite the ambiguity of the sparse measurements. The state CRPS in



Metric	Method	Error(MAE) ↓
$\lvert\Delta F_0\rvert$	E+C,F	$\mathbf{0.006 \pm 0.001}$
	E+C,G	0.471 ± 0.075
	E,F	1.000 ± 0.000
	C,F	0.010 ± 0.002
	AmEnF	0.869 ± 0.026
$\lvert\Delta \log Re\rvert$	E+C,F	$\mathbf{0.079 \pm 0.015}$
	E+C,G	11.516 ± 0.000
	E,F	1.866 ± 0.447
	C,F	0.100 ± 0.023
	AmEnF	0.423 ± 0.304
$\lvert\Delta b\rvert$	E+C,F	0.030 ± 0.010
	E+C,G	1.583 ± 0.155
	E,F	0.908 ± 0.031
	C,F	$\mathbf{0.028 \pm 0.013}$
	AmEnF	0.517 ± 0.022

Figure 3: Kolmogorov state reconstruction and model calibration. The table reports five-seed mean $\pm$ standard deviation of $|\Delta F_0|$, $|\Delta \log \mathrm{Re}|$, and bias-field MAE over observed locations.

Table 2 supports this qualitative agreement. With ELBO+CRPS training, replacing the flow with a diagonal Gaussian substantially degrades both state inference and parameter recovery.

The ELBO-only flow performs poorly in both state inference and physical-parameter estimation. The CRPS-only flow accurately estimates physical parameters and sensor biases but yields a much poorer filtering distribution. Its state-inference failure therefore cannot be explained solely by incorrect model calibration. Combining predictive CRPS with the recursive ELBO improves both tasks, supporting their complementary roles in observation-only learning. At $T = 512$, we similarly assimilate four supplementary signed linear measurements at previously unmeasured cells (noise standard deviation 1). After likelihood reweighting, the ELBO+CRPS flow achieves the lowest CRPS averaged over the remaining 16,380 cells (Table 2, final column). Additional ablations (Appendix E.2.2) show benefits from longer observation histories and useful state inference after training on only 32 observation sequences, highlighting NFF's data efficiency in this setting.

Takeaways. NFF recovers the state while jointly calibrating physical parameters and sensor biases from observations alone in 16,384 dimensions.

6 Limitations

NFF requires differentiable transition and observation models. Its exact-recovery guarantees assume a correctly specified SSM, including the initial prior, exact τ-step memory, sufficient recognition capacity, and global optimization. Longer windows reduce memory truncation but increase computational cost; Appendix C analyzes the recursive-ELBO gap. Finite capacity, imperfect optimization, and distribution shift can degrade inference. Misspecification can produce concentrated but incorrect posteriors, motivating held-out predictive checks. Generative parameters are identifiable only up to observational equivalence, so the learned model's latent posterior need not match the data-generating latent posterior. A highly sparse observation network may limit practical parameter identifiability, as our Kolmogorov observation-density study suggests (Appendix E.2.1; Fig. 10). Parameters are point estimates; uncertainty quantification remains future work.

7 Conclusion

NFF provides an amortized variational approach that jointly learns non-Gaussian filtering densities and calibrates prescribed physical models from observations alone, without latent-state targets or replacement dynamics. Theory characterizes the expected objective's global optima and quantifies finite-window gaps under stated assumptions, while experiments demonstrate this approach from controlled multimodality to 16,384-dimensional Kolmogorov flow.

AI USE STATEMENT

The authors conceived the research ideas, including the use of normalizing flows and the ELBO decomposition, without generative AI assistance. Generative AI was used to write the mathematical proof, which the authors subsequently checked.

We also used generative AI tools to provide feedback on experimental methodology and evaluation protocols, assist with implementing and debugging methods and evaluation code, identify and summarize related literature, and draft, organize, and edit the manuscript. The authors interpreted the experimental results and used generative AI tools to cross-check their interpretations.

Generative AI was used to write the simulation-based data-generation code. The numerical datasets were produced by executing this code, and the authors visually inspected the resulting simulation outputs. Dataset preprocessing was not part of the workflow.

The authors take full responsibility for the final manuscript and accompanying code and data, including all AI-assisted content.

REPRODUCIBILITY STATEMENT

We provide the code as a supplementary material to reproduce the core NFF experiments on the coupled oscillator, Lorenz–96, and Kolmogorov flow, together with data-generation scripts for Lorenz–96 and Kolmogorov flow. The oscillator implementation generates data online, whereas the Lorenz–96 and Kolmogorov-flow experiments use datasets generated in a separate step with the supplied scripts. An accompanying `README.md` provides the exact steps for data generation, training, and evaluation. Appendix D specifies the SSMs, data-generation procedures, model architectures, training hyperparameters, and evaluation protocols. Appendices A, B,and C state the assumptions and provide the derivations and proofs supporting the theoretical results.

REFERENCES

Erik Anderson and Heikki Järvinen. Variational quality control. *Quarterly Journal of the Royal Meteorological Society*, 125(554):697–722, 1999. doi: 10.1002/qj.49712555416.

Eviatar Bach, Ricardo Baptista, Jochen Bröcker, Bohan Chen, and Andrew Stuart. Learning probabilistic filters with strictly proper scoring rules, 2026. URL https://arxiv.org/abs/2606.26497.

Feng Bao, Zezhong Zhang, and Guannan Zhang. An ensemble score filter for tracking high-dimensional nonlinear dynamical systems. *Computer Methods in Applied Mechanics and Engineering*, 432:117447, 2024. doi: 10.1016/j.cma.2024.117447.

Pierre Boudier, Anthony Fillion, Serge Gratton, Selime Gürol, and Sixin Zhang. Data assimilation networks. *Journal of Advances in Modeling Earth Systems*, 15(4):e2022MS003353, 2023. doi: 10.1029/2022MS003353.

Andrew Campbell, Yuyang Shi, Thomas Rainforth, and Arnaud Doucet. Online variational filtering and parameter learning. In *Advances in Neural Information Processing Systems*, volume 34, pp. 18633–18645, 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/9a6a1aaafe73c572b7374828b03a1881-Paper.pdf.

Alberto Carrassi, Marc Bocquet, Laurent Bertino, and Geir Evensen. Data assimilation in the geosciences: An overview of methods, issues, and perspectives. *WIREs Climate Change*, 9(5):e535, 2018. doi: 10.1002/wcc.535.

Siyi Chen, Yixuan Jia, Qing Qu, He Sun, and Jeffrey Fessler. FlowDAS: A stochastic interpolant-based framework for data assimilation. In *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 103061–103102, 2025. doi: 10.52202/085713-3445.

Yuming Chen, Daniel Sanz-Alonso, and Rebecca Willett. Reduced-order autodifferentiable ensemble Kalman filters. *Inverse Problems*, 39(12):124001, oct 2023. doi: 10.1088/1361-6420/acff14.

Tiangang Cui, Xiaodong Feng, Chenlong Pei, Xiaoliang Wan, and Tao Zhou. Amortized filtering and smoothing with conditional normalizing flows, 2026. URL https://arxiv.org/abs/2604.07169.

Emmanuel de Bézenac, Syama Sundar Rangapuram, Konstantinos Benidis, Michael Bohlke-Schneider, Richard Kurle, Lorenzo Stella, Hilaf Hasson, Patrick Gallinari, and Tim Januschowski. Normalizing Kalman filters for multivariate time series analysis. In *Advances in Neural Information Processing Systems*, volume 33, pp. 2995–3007, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/1f47cef5e38c952f94c5d61726027439-Paper.pdf.

François-Xavier Le Dimet and Olivier Talagrand. Variational algorithms for analysis and assimilation of meteorological observations: theoretical aspects. *Tellus A*, 38A(2):97–110, 1986. doi: 10.1111/j.1600-0870.1986.tb00459.x.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real NVP. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=HkpbnH9lx.

Randal Douc, Eric Moulines, and Ya'acov Ritov. Forgetting of the initial condition for the filter in general state-space hidden Markov chain: a coupling approach, 2007. URL https://arxiv.org/abs/0712.0285.

Matthew Dowling, Yuan Zhao, and Il Memming Park. Real-time variational method for learning neural trajectory and its dynamics. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=M_MvkWgQSt.

Geir Evensen. Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *Journal of Geophysical Research: Oceans*, 99(C5): 10143–10162, 1994. doi: 10.1029/94JC00572.

R. Fablet, B. Chapron, L. Drumetz, E. Mémin, O. Pannekoucke, and F. Rousseau. Learning variational data assimilation models and solvers. *Journal of Advances in Modeling Earth Systems*, 13 (10):e2021MS002572, 2021. doi: 10.1029/2021MS002572.

C. A. T. Ferro. Fair scores for ensemble forecasts. *Quarterly Journal of the Royal Meteorological Society*, 140(683):1917–1923, 2014. doi: 10.1002/qj.2270.

Thomas Frerix, Dmitrii Kochkov, Jamie Smith, Daniel Cremers, Michael Brenner, and Stephan Hoyer. Variational data assimilation with a learned inverse observation operator. In *International Conference on Machine Learning*, volume 139, pp. 3449–3458, 2021. URL https://proceedings.mlr.press/v139/frerix21a.html.

A. J. Geer, S. Migliorini, and M. Matricardi. All-sky assimilation of infrared radiances sensitive to mid- and upper-tropospheric moisture and cloud. *Atmospheric Measurement Techniques*, 12 (9):4903–4929, 2019. doi: 10.5194/amt-12-4903-2019. URL https://amt.copernicus.org/articles/12/4903/2019/.

Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007. doi: 10.1198/016214506000001437.

Karol Gregor, George Papamakarios, Frederic Besse, Lars Buesing, and Theophane Weber. Temporal difference variational auto-encoder. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=S1x4ghC9tQ.

Chih-Chi Hu and Peter Jan van Leeuwen. A particle flow filter for high-dimensional system applications. *Quarterly Journal of the Royal Meteorological Society*, 147(737):2352–2374, 2021. doi: 10.1002/qj.4028.

R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, 1960. doi: 10.1115/1.3662552.

François Le Gland and Laurent Mevel. Exponential forgetting and geometric ergodicity in hidden Markov models. *Mathematics of Control, Signals and Systems*, 13:63–93, 2000. doi: 10.1007/PL00009861.

John M. Lewis and John C. Derber. The use of adjoint equations to solve a variational adjustment problem with advective constraints. *Tellus A*, 37A(4):309–322, 1985. doi: 10.1111/j.1600-0870.1985.tb00430.x.

Chris J Maddison, John Lawson, George Tucker, Nicolas Heess, Mohammad Norouzi, Andriy Mnih, Arnaud Doucet, and Yee Teh. Filtering variational objectives. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/fa84632d742f2729dc32ce8cb5d49733-Paper.pdf.

Michael McCabe and Jed Brown. Learning to assimilate in chaotic dynamical systems. In *Advances in Neural Information Processing Systems*, volume 34, pp. 12237–12250, 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/65cc2c8205a05d7379fa3a6386f710e1-Paper.pdf.

Christian Naesseth, Scott Linderman, Rajesh Ranganath, and David Blei. Variational sequential Monte Carlo. In *International Conference on Artificial Intelligence and Statistics*, volume 84, pp. 968–977, 2018. URL https://proceedings.mlr.press/v84/naesseth18a.html.

Mathis Peyron, Anthony Fillion, Selime Gürol, Victor Marchais, Serge Gratton, Pierre Boudier, and Gael Goret. Latent space data assimilation by using deep learning. *Quarterly Journal of the Royal Meteorological Society*, 147(740):3759–3777, 2021. doi: 10.1002/qj.4153.

Guy Revach, Nir Shlezinger, Ruud J. G. van Sloun, and Yonina C. Eldar. KalmanNet: Data-driven Kalman filtering. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 3905–3909, 2021. doi: 10.1109/ICASSP39728.2021.9413750.

François Rozet and Gilles Louppe. Score-based data assimilation. In *Advances in Neural Information Processing Systems*, volume 36, pp. 40521–40541, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/7f7fa581cc8a1970a4332920cdf87395-Paper-Conference.pdf.

Phillip Si and Peng Chen. Latent-EnSF: A latent ensemble score filter for high-dimensional data assimilation with sparse observation data. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=urcEYsZOBz.

Phillip Si and Peng Chen. LEVDA: Latent ensemble variational data assimilation via differentiable dynamics, 2026. URL https://arxiv.org/abs/2602.19406.

Andrea Storto. Variational quality control of hydrographic profile data with non-Gaussian errors for global ocean variational data assimilation systems. *Ocean Modelling*, 104:226–241, 2016. doi: 10.1016/j.ocemod.2016.06.011.

Yuta Tarumi. PR-Smoother: Simulator-preserving non-gaussian smoothing for data assimilation, 2026. URL https://arxiv.org/abs/2609.26890.

Yuta Tarumi, Keisuke Fukuda, and Shin-Ichi Maeda. Deep Bayesian filter for Bayes-faithful data assimilation. In *International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 59182–59209, 2025. URL https://proceedings.mlr.press/v267/tarumi25a.html.

Taos Transue, Bohan Chen, So Takao, and Bao Wang. Flow matching for efficient and scalable data assimilation. 2025. doi: 10.48550/arXiv.2508.13313.

Peter Jan van Leeuwen, Hans R. Künsch, Lars Nerger, Roland Potthast, and Sebastian Reich. Particle filters for high-dimensional geoscience applications: A review. *Quarterly Journal of the Royal Meteorological Society*, 145(723):2335–2365, 2019. doi: 10.1002/qj.3551.

Xintong Wang, Xiaofei Guan, Ling Guo, and Hao Wu. Flow-based Bayesian filtering for high-dimensional nonlinear stochastic dynamical systems. *Computer Methods in Applied Mechanics and Engineering*, 446:118285, 2025. doi: 10.1016/j.cma.2025.118285.

Pengpeng Xiao, Phillip Si, and Peng Chen. LD-EnSF: Synergizing latent dynamics with ensemble score filters for fast data assimilation with sparse observations. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=AWSVzzhbr7.

H. W. Xie, D. Sujono, T. Ryder, E. B. Sudderth, and S. D. Allison. A framework for variational inference and data assimilation of soil biogeochemical models using normalizing flows. *Journal of Advances in Modeling Earth Systems*, 17(8):e2024MS004547, 2025. doi: 10.1029/2024MS004547.

Liang Xu and Ruixin Niu. EKFNet: Learning system noise covariance parameters for nonlinear tracking. *IEEE Transactions on Signal Processing*, 72:3139–3152, 2024. doi: 10.1109/TSP.2024.3417350.

Yuan Zhao and Il Memming Park. Variational online learning of neural dynamics. *Frontiers in Computational Neuroscience*, Volume 14 - 2020, 2020. doi: 10.3389/fncom.2020.00071.

A EXPECTED-GAP DECOMPOSITION FOR THE RECURSIVE ELBO

This appendix derives the expected-gap identity used in the main text (equation 13). Let W_t be the masked τ-step window constructed deterministically from $Y_{1:t}$. We assume that the conditional densities below exist and that the displayed expectations and KL divergences are finite. The expected-gap identity is exact for any window length. The forgetting assumption is needed only to show that the exact filtering recursion is representable by the windowed recognition model and can attain zero gap.

Local recursive objective. For $t \geq 2$, define the two-slice variational distribution

$$q_{\phi,t}(X_t, X_{t-1} \mid W_t, W_{t-1}) = q_\phi^F(X_t \mid W_t) q_\phi^B(X_{t-1} \mid X_t, W_{t-1}). \tag{20}$$

Here, W_t and W_{t-1} together cover $Y_{t-\tau:t}$. The local NFF objective is

$$\mathcal{L}_t(\theta, \phi; Y_{1:t}) = \iint q_{\phi,t}(X_t, X_{t-1} \mid W_t, W_{t-1}) \log \frac{p_\theta^{\text{obs}}(Y_t \mid X_t) p_\theta^{\text{dyn}}(X_t \mid X_{t-1}) q_\phi^F(X_{t-1} \mid W_{t-1})}{q_{\phi,t}(X_t, X_{t-1} \mid W_t, W_{t-1})} \, dX_t \, dX_{t-1}. \tag{21}$$

Define the one-step observation predictor induced by the preceding amortized filter (equation 6),

$$\widetilde{p}_{\theta,\phi,t}(Y_t \mid W_{t-1}) = \iint p_\theta^{\text{obs}}(Y_t \mid X_t) p_\theta^{\text{dyn}}(X_t \mid X_{t-1}) q_\phi^F(X_{t-1} \mid W_{t-1}) \, dX_t \, dX_{t-1}, \tag{22}$$

and its adjacent-state posterior (equation 7),

$$\widetilde{\pi}_{\theta,\phi,t}(X_t, X_{t-1} \mid Y_t, W_{t-1}) = \frac{p_\theta^{\text{obs}}(Y_t \mid X_t) p_\theta^{\text{dyn}}(X_t \mid X_{t-1}) q_\phi^F(X_{t-1} \mid W_{t-1})}{\widetilde{p}_{\theta,\phi,t}(Y_t \mid W_{t-1})}. \tag{23}$$

Substitution into equation 21 gives the ordinary ELBO identity

$$\mathcal{L}_t(\theta, \phi; Y_{1:t}) = \log \widetilde{p}_{\theta,\phi,t}(Y_t \mid W_{t-1}) - \text{KL}\left[q_{\phi,t}(X_t, X_{t-1} \mid W_t, W_{t-1}) \| \widetilde{\pi}_{\theta,\phi,t}(X_t, X_{t-1} \mid Y_t, W_{t-1})\right]. \tag{24}$$

Thus, L_t is a pointwise lower bound on $\log \tilde{p}_{\theta,\phi,t}(Y_t \mid W_{t-1})$. It is not generally a pointwise lower bound on $\log p_\theta(Y_t \mid Y_{1:t-1})$, because the induced predictor uses $q_\phi^F(X_{t-1} \mid W_{t-1})$ in place of the exact candidate-model filter $p_\theta(X_{t-1} \mid Y_{1:t-1})$.

Exact expected-gap identity. Let $p^\star$ denote the data-generating distribution. From equation 24,

$$\begin{aligned}
&\mathbb{E}_{p^\star}\left[\log p^\star(Y_t \mid Y_{1:t-1}) - \mathcal{L}_t(\theta,\phi;Y_{1:t})\right] \\
&= \mathbb{E}_{p^\star(Y_{1:t-1})}\mathrm{KL}\left[p^\star(Y_t \mid Y_{1:t-1}) \| \widetilde{p}_{\theta,\phi,t}(Y_t \mid W_{t-1})\right] \\
&\quad + \mathbb{E}_{p^\star(Y_{1:t})}\mathrm{KL}\left[q_{\phi,t}(X_t, X_{t-1} \mid W_t, W_{t-1}) \| \widetilde{\pi}_{\theta,\phi,t}(X_t, X_{t-1} \mid Y_t, W_{t-1})\right].
\end{aligned} \tag{25}$$

For each fixed complete history $Y_{1:t-1}$, the first term follows by averaging the log-density ratio over $Y_t \sim p^\star(Y_t \mid Y_{1:t-1})$. The candidate density in that KL is constrained to depend on the history only through W_{t-1}, but the true conditional remains $p^\star(Y_t \mid Y_{1:t-1})$. Therefore, equation 25 is exact without a forgetting assumption. When forgetting is not exact, this KL can retain an irreducible contribution from information discarded by the finite window.

The initial-time objective is

$$\mathcal{L}_1(\theta,\phi;Y_1) = \mathbb{E}_{q_\phi^F(X_1|W_1)}\left[\log \frac{p_0(X_1)p_\theta^{\mathrm{obs}}(Y_1 \mid X_1)}{q_\phi^F(X_1 \mid W_1)}\right]. \tag{26}$$

Its expected gap is

$$\begin{aligned}
&\mathbb{E}_{p^\star}\left[\log p^\star(Y_1) - \mathcal{L}_1(\theta,\phi;Y_1)\right] \\
&= \mathrm{KL}\left[p^\star(Y_1) \| p_\theta(Y_1)\right] + \mathbb{E}_{p^\star(Y_1)}\mathrm{KL}\left[q_\phi^F(X_1 \mid W_1) \| p_\theta(X_1 \mid Y_1)\right].
\end{aligned} \tag{27}$$

Summing equation 25 and equation 27 gives

$$\begin{aligned}
\Delta_{\mathrm{NFF}}(\theta,\phi) &:= \mathbb{E}_{p^\star}\left[\log p^\star(Y_{1:T}) - \mathcal{L}_{\mathrm{NFF}}(\theta,\phi;Y_{1:T})\right] \\
&= \mathrm{KL}\left[p^\star(Y_1) \| p_\theta(Y_1)\right] + \mathbb{E}_{p^\star(Y_1)}\mathrm{KL}\left[q_\phi^F(X_1 \mid W_1) \| p_\theta(X_1 \mid Y_1)\right] \\
&\quad + \sum_{t=2}^{T} \mathbb{E}_{p^\star(Y_{1:t-1})}\mathrm{KL}\left[p^\star(Y_t \mid Y_{1:t-1}) \| \widetilde{p}_{\theta,\phi,t}(Y_t \mid W_{t-1})\right] \\
&\quad + \sum_{t=2}^{T} \mathbb{E}_{p^\star(Y_{1:t})}\mathrm{KL}\left[q_{\phi,t}(X_t, X_{t-1} \mid W_t, W_{t-1}) \| \widetilde{\pi}_{\theta,\phi,t}(X_t, X_{t-1} \mid Y_t, W_{t-1})\right] \\
&\geq 0.
\end{aligned} \tag{28}$$

Hence the expected data-generating log likelihood is bounded below by the expected recursive ELBO. The averaging over observation sequences is the one induced by amortized training. Empirical averages over independently generated sequences from the data-generating SSM, initialized from p_0, approximate this expectation. If interior blocks from longer trajectories are treated as separate sequences, the boundary term must instead use the corresponding block-start state distribution; approximation by averages over dependent blocks additionally requires the usual ergodicity conditions. Repeated realizations following exactly the same history are not required.

Recovery with a fixed, correctly specified SSM. Assume the following conditions.

1. The candidate SSM is fixed at the data-generating model, $\theta = \theta^\star$, including the initial prior p_0.
2. The filtering distribution has exact τ-step memory on histories generated by the system:
$$p^\star(X_t \mid Y_{1:t}) = p^\star(X_t \mid W_t), \qquad t = 1, \ldots, T. \tag{29}$$
3. A single shared parameter vector $\phi^\star$ can represent, for every relevant t, both
$$q_{\phi^\star}^F(X_t \mid W_t) = p^\star(X_t \mid Y_{1:t}), \tag{30}$$
$$q_{\phi^\star}^B(X_{t-1} \mid X_t, W_{t-1}) = p^\star(X_{t-1} \mid X_t, Y_{1:t-1}). \tag{31}$$

The backward conditional in equation 31 depends on the past through the preceding filter,

$$p^\star(X_{t-1} \mid X_t, Y_{1:t-1}) \propto p_{\theta^\star}^{\mathrm{dyn}}(X_t \mid X_{t-1}) p^\star(X_{t-1} \mid Y_{1:t-1}), \tag{32}$$

so equation 29 makes W_{t-1} sufficient for this factor as well.

Under these assumptions, $(\theta^\star, \phi^\star)$ makes every term in equation 28 zero. Conversely, let $\widehat{\phi}$ be any global maximizer of the expected recursive ELBO with $\theta = \theta^\star$. Since zero gap is attainable, every nonnegative term in equation 28 must vanish at $\widehat{\phi}$.

At $t = 1$, the posterior KL gives

$$q^F_{\widehat{\phi}}(X_1 \mid W_1) = p^\star(X_1 \mid Y_1). \tag{33}$$

Suppose inductively that

$$q^F_{\widehat{\phi}}(X_{t-1} \mid W_{t-1}) = p^\star(X_{t-1} \mid Y_{1:t-1}). \tag{34}$$

Then equation 22 and equation 23 reduce to the true one-step predictor and true adjacent-state posterior:

$$\widetilde{p}_{\theta^\star,\widehat{\phi},t}(Y_t \mid W_{t-1}) = p^\star(Y_t \mid Y_{1:t-1}), \tag{35}$$

$$\widetilde{\pi}_{\theta^\star,\widehat{\phi},t}(X_t, X_{t-1} \mid Y_t, W_{t-1}) = p^\star(X_t, X_{t-1} \mid Y_{1:t}). \tag{36}$$

The adjacent-state KL in equation 25 therefore gives

$$q_{\widehat{\phi},t}(X_t, X_{t-1} \mid W_t, W_{t-1}) = p^\star(X_t, X_{t-1} \mid Y_{1:t}). \tag{37}$$

Marginalizing equation 37 over X_{t-1} recovers the filter at time t. Taking the conditional distribution of X_{t-1} given X_t recovers the backward factor. The SSM conditional independence gives

$$p^\star(X_{t-1} \mid X_t, Y_{1:t}) = p^\star(X_{t-1} \mid X_t, Y_{1:t-1}). \tag{38}$$

Thus the induction proves, for every t and for inputs generated by the corresponding distributions,

$$q^F_{\widehat{\phi}}(X_t \mid W_t) = p^\star(X_t \mid Y_{1:t}), \tag{39}$$

$$q^B_{\widehat{\phi}}(X_{t-1} \mid X_t, W_{t-1}) = p^\star(X_{t-1} \mid X_t, Y_{1:t-1}). \tag{40}$$

Learning point-valued generative parameters. Now let θ also be optimized, and retain the realizability assumptions above at $(\theta^\star, \phi^\star)$. Since this pair attains zero gap, any global maximizer $(\widehat{\theta}, \widehat{\phi})$ must also satisfy $\Delta_{\mathrm{NFF}}(\widehat{\theta}, \widehat{\phi}) = 0$.

The initial two KL terms give

$$p_{\widehat{\theta}}(Y_1) = p^\star(Y_1), \tag{41}$$

$$q^F_{\widehat{\phi}}(X_1 \mid W_1) = p_{\widehat{\theta}}(X_1 \mid Y_1). \tag{42}$$

Assume inductively that the recognition filter equals the candidate-model filter at time $t-1$. The initial equality $p_\theta(Y_1) = p^\star(Y_1)$ and the already established predictive equalities for times $2, \dots, t-1$ imply, by the chain rule, that $p_\theta(Y_{1:t-1}) = p^\star(Y_{1:t-1})$. The inductive filtering equality then makes the induced predictor equal to $p_\theta(Y_t \mid Y_{1:t-1})$ on histories generated by $p^\star$. Since the predictive-observation KL vanishes, we obtain

$$p_{\widehat{\theta}}(Y_t \mid Y_{1:t-1}) = p^\star(Y_t \mid Y_{1:t-1}). \tag{43}$$

The adjacent-state KL gives the exact candidate-model adjacent-state posterior. Its marginal and, using the same SSM conditional independence as above, its backward conditional yield

$$q^F_{\widehat{\phi}}(X_t \mid W_t) = p_{\widehat{\theta}}(X_t \mid Y_{1:t}), \tag{44}$$

$$q^B_{\widehat{\phi}}(X_{t-1} \mid X_t, W_{t-1}) = p_{\widehat{\theta}}(X_{t-1} \mid X_t, Y_{1:t-1}). \tag{45}$$

Applying the chain rule to equation 41 and equation 43 gives

$$p_{\widehat{\theta}}(Y_{1:T}) = p^\star(Y_{1:T}). \tag{46}$$

Conversely, if a pair (θ, ϕ) reproduces the complete observation law and its windowed recognition factors equal the exact candidate-model filtering and backward conditionals on histories generated by $p^\star$, then every term in equation 28 is zero. Therefore,

$$\Delta_{\mathrm{NFF}}(\theta, \phi) = 0 \quad \Longleftrightarrow \quad \begin{cases} p_\theta(Y_{1:T}) = p^\star(Y_{1:T}), & \\ q^F_\phi(X_t \mid W_t) = p_\theta(X_t \mid Y_{1:t}), & t = 1, \dots, T, \\ q^B_\phi(X_{t-1} \mid X_t, W_{t-1}) = p_\theta(X_{t-1} \mid X_t, Y_{1:t-1}), & t = 2, \dots, T. \end{cases} \tag{47}$$

The equalities concern histories generated by $p^\star$ and, for the backward factor, states generated by the corresponding candidate-model filter. Values of the recognition networks outside these inputs do not affect the expected objective. Since the left-hand sides of the last two equalities are functions of finite windows, zero gap also implies that the candidate-model conditionals admit this windowed representation on the relevant histories. No separate forgetting assumption is imposed on every observationally equivalent candidate model.

Define the finite-horizon observational equivalence class

$$\Theta^\star_{\mathrm{obs}} := \{\theta : p_\theta(Y_{1:T}) = p^\star(Y_{1:T})\}. \tag{48}$$

Every global maximizer has $\widehat{\theta} \in \Theta^\star_{\mathrm{obs}}$. Hence every component identifiable from the observation law equals its data-generating value. Without full observational identifiability, equation 44 and equation 45 recover the exact conditionals under the observationally equivalent candidate model, which need not coincide with the data-generating latent posteriors. With fixed $\theta = \theta^\star$, or with full observational identifiability, they reduce to equation 39 and equation 40.

B Preservation of the global optima after adding future-observation CRPS

This appendix shows that future-observation CRPS preserves the global minimizers of the expected recursive-ELBO loss. We again work directly with the finite-window filter. Exact forgetting is needed only to ensure that the data-generating filtering distribution is available to the windowed recognition model, as established in Appendix A. Throughout this appendix, we retain the correct-specification, exact-window-memory, and simultaneous-realizability assumptions of Appendix A.

Future-observation distribution induced by NFF. For a valid forecast origin t and horizon $h \geq 1$ with $t + h \leq T$, draw

$$X_t \sim q^F_\phi(\cdot \mid W_t), \tag{49}$$

then propagate through the candidate SSM,

$$X_{t+r} \sim p^{\mathrm{dyn}}_\theta(\cdot \mid X_{t+r-1}), \qquad r = 1, \ldots, h, \tag{50}$$

$$\widetilde{Y}_{t+h} \sim p^{\mathrm{obs}}_\theta(\cdot \mid X_{t+h}). \tag{51}$$

All process and observation noise in the SSM is included. For scalar observation coordinate j, let

$$F_{\theta,\phi,t,h,j}(z \mid W_t) := \Pr_{\theta,\phi}\left(\widetilde{Y}_{t+h,j} \leq z \mid W_t\right) \tag{52}$$

be the induced predictive CDF, and let

$$F^\star_{t,h,j}(z \mid Y_{1:t}) := \overset{\star}{\Pr}\left(Y_{t+h,j} \leq z \mid Y_{1:t}\right) \tag{53}$$

be the data-generating conditional CDF. Future observations are verification targets only; they are not inputs to $q^F_\phi(X_t \mid W_t)$.

Expected CRPS gap. For a scalar CDF F and scalar observation Y,

$$\mathrm{CRPS}(F, Y) = \int_{\mathbb{R}} \left(F(z) - \mathbf{1}\{Y \leq z\}\right)^2 \, dz. \tag{54}$$

Assume that, for every scored (t, h, j), both $F_{\theta,\phi,t,h,j}(\cdot \mid W_t)$ and $F^\star_{t,h,j}(\cdot \mid Y_{1:t})$ have finite first moments on histories generated by $p^\star$, so that the corresponding CRPS values are finite.

$$\begin{aligned}
&\mathbb{E}_{p^\star}\left[\mathrm{CRPS}\left(F_{\theta,\phi,t,h,j}(\cdot \mid W_t), Y_{t+h,j}\right) - \mathrm{CRPS}\left(F^\star_{t,h,j}(\cdot \mid Y_{1:t}), Y_{t+h,j}\right) \middle| Y_{1:t}\right] \\
&= \int_{\mathbb{R}} \left[F_{\theta,\phi,t,h,j}(z \mid W_t) - F^\star_{t,h,j}(z \mid Y_{1:t})\right]^2 \, dz \geq 0.
\end{aligned} \tag{55}$$

Equality holds exactly when the two conditional predictive distributions coincide for histories generated by $p^\star$. This is the strict propriety of CRPS.

Let $\mathcal{I}$ contain the scored triples (t, h, j), all satisfying $t + h \le T$, and let $w_{t,h,j} \ge 0$. Define

$$\mathcal{R}_{\mathrm{CRPS}}(\theta, \phi) = \sum_{(t,h,j)\in\mathcal{I}} w_{t,h,j} \mathbb{E}_{p^\star} \mathrm{CRPS}\left(F_{\theta,\phi,t,h,j}(\cdot \mid W_t), Y_{t+h,j}\right). \tag{56}$$

Let $\mathcal{R}^\star_{\mathrm{CRPS}}$ denote the same weighted score evaluated at the true conditional CDFs. Averaging equation 55 gives

$$\begin{aligned}\Delta_{\mathrm{CRPS}}(\theta, \phi) &:= \mathcal{R}_{\mathrm{CRPS}}(\theta, \phi) - \mathcal{R}^\star_{\mathrm{CRPS}} \\ &= \sum_{(t,h,j)\in\mathcal{I}} w_{t,h,j} \mathbb{E}_{p^\star(Y_{1:t})} \int_{\mathbb{R}} \left[F_{\theta,\phi,t,h,j}(z \mid W_t) - F^\star_{t,h,j}(z \mid Y_{1:t})\right]^2 dz \\ &\ge 0. \end{aligned} \tag{57}$$

At $(\theta^\star, \phi^\star)$, exact filtering and propagation through the correctly specified SSM give

$$F_{\theta^\star,\phi^\star,t,h,j}(\cdot \mid W_t) = F^\star_{t,h,j}(\cdot \mid Y_{1:t}), \tag{58}$$

so $\Delta_{\mathrm{CRPS}}(\theta^\star, \phi^\star) = 0$.

The fair finite-ensemble estimator. The implementation uses $K \ge 2$ forecast members. Conditional on $Y_{1:t}$, the members

$$\widetilde{Y}^{(1)}_{t+h,j}, \ldots, \widetilde{Y}^{(K)}_{t+h,j} \overset{\text{i.i.d.}}{\sim} F_{\theta,\phi,t,h,j}(\cdot \mid W_t) \tag{59}$$

are generated using independent recognition-base, process-noise, and observation-noise draws. They are also independent of the realized verification target $Y_{t+h,j}$ conditional on $Y_{1:t}$. The fair estimator is

$$\begin{aligned}\widehat{\mathrm{CRPS}}_K = &\frac{1}{K}\sum_{k=1}^{K} \left|\widetilde{Y}^{(k)}_{t+h,j} - Y_{t+h,j}\right| \\ &- \frac{1}{K(K-1)} \sum_{k<\ell} \left|\widetilde{Y}^{(k)}_{t+h,j} - \widetilde{Y}^{(\ell)}_{t+h,j}\right|. \end{aligned} \tag{60}$$

Let Z, Z' be independent draws from the candidate predictive distribution. Conditional on the history and realized target,

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K} |Z_k - Y_{t+h,j}| \middle| Y_{1:t}, Y_{t+h,j}\right] = \mathbb{E}|Z - Y_{t+h,j}|, \tag{61}$$

$$\mathbb{E}\left[\frac{1}{K(K-1)} \sum_{k<\ell} |Z_k - Z_\ell| \middle| Y_{1:t}, Y_{t+h,j}\right] = \frac{1}{2}\mathbb{E}|Z - Z'|. \tag{62}$$

Therefore,

$$\mathbb{E}\left[\widehat{\mathrm{CRPS}}_K \middle| Y_{1:t}, Y_{t+h,j}\right] = \mathrm{CRPS}\left(F_{\theta,\phi,t,h,j}(\cdot \mid W_t), Y_{t+h,j}\right). \tag{63}$$

The finite ensemble changes the Monte Carlo variance but not the expected training target. Accordingly, the global-minimizer statement below concerns the objective averaged over both observation sequences and forecast-sampling randomness, not a fixed realization of the Monte Carlo draws.

Combined expected objective. Consider

$$\mathcal{J}_\lambda(\theta, \phi) = -\mathbb{E}_{p^\star}\mathcal{L}_{\mathrm{NFF}}(\theta, \phi; Y_{1:T}) + \lambda_{\mathrm{CRPS}}\mathcal{R}_{\mathrm{CRPS}}(\theta, \phi), \qquad \lambda_{\mathrm{CRPS}} \ge 0. \tag{64}$$

For any parameter pair $(\theta^\star, \phi^\star)$ that represents the data-generating SSM and its exact windowed filtering and backward conditionals,

$$\begin{aligned}\mathcal{J}_\lambda(\theta, \phi) - \mathcal{J}_\lambda(\theta^\star, \phi^\star) &= \Delta_{\mathrm{NFF}}(\theta, \phi) + \lambda_{\mathrm{CRPS}}\Delta_{\mathrm{CRPS}}(\theta, \phi) \\ &\ge 0. \end{aligned} \tag{65}$$

Thus the data-generating solution is a global minimizer.

Let $\mathcal{Z}_{\mathrm{NFF}}$ denote the set of parameter pairs satisfying the three equalities in equation 47. Every pair in $\mathcal{Z}_{\mathrm{NFF}}$ also has zero CRPS gap. Indeed, its recognition filter is the exact filter under p_θ, and $p_\theta(Y_{1:T}) = p^\star(Y_{1:T})$. Propagation through the candidate SSM therefore gives

$$F_{\theta,\phi,t,h,j}(z \mid W_t) = \Pr_\theta(Y_{t+h,j} \le z \mid Y_{1:t}) = F^\star_{t,h,j}(z \mid Y_{1:t}) \tag{66}$$

for every scored triple and for histories generated by $p^\star$, where $t + h \le T$ ensures that the required conditional is contained in the matched observation law.

Conversely, any global minimizer of equation 64 must make the right-hand side of equation 65 zero. Since both terms are nonnegative, it must satisfy $\Delta_{\mathrm{NFF}} = 0$ and therefore belong to $\mathcal{Z}_{\mathrm{NFF}}$. Hence, for every $\lambda_{\mathrm{CRPS}} \ge 0$,

$$\arg\min_{\theta,\phi} \mathcal{J}_\lambda(\theta,\phi) = \mathcal{Z}_{\mathrm{NFF}} = \arg\min_{\theta,\phi} \left\{-\mathbb{E}_{p^\star} \mathcal{L}_{\mathrm{NFF}}(\theta,\phi;Y_{1:T})\right\}. \tag{67}$$

Under full observational identifiability, every pair in this set has $\theta = \theta^\star$, and its recognition factors equal the data-generating filtering and backward conditionals on the relevant histories. If only some components of θ are identifiable, those components equal their data-generating values. The remaining components are determined only up to observational equivalence, and the recognition factors are exact under the corresponding candidate model but need not equal the data-generating latent posteriors. The equality in equation 67 concerns global minimizers of the expected objective under the stated realizability assumptions. With finite data, limited recognition capacity, model misspecification, or imperfect optimization, the choice of λ_{CRPS} can affect the fitted solution.

Why the recursive ELBO remains necessary. Coordinatewise CRPS identifies only the scored scalar future-observation marginals. It does not determine cross-coordinate dependence, adjacent-state dependence, or the latent filtering distribution. Distinct latent distributions can therefore attain the same coordinatewise predictive CRPS. The recursive ELBO supplies the dynamics-consistent joint constraint that yields equation 47; CRPS is a compatible auxiliary that can improve the practical mode-coverage signal without changing the desired global minimizers.

C Population-gap bounds under approximate forgetting

The exact-memory assumption in Appendix A ensures that the population gap can vanish. This appendix quantifies a nonzero gap when older observations retain a small influence on the filter. We first derive an exact identity for a finite-window oracle, and then bound the best attainable NFF population gap under quantitative forgetting assumptions. These results concern the recursive-ELBO objective; they do not assume that a finite-window filter is itself an exact full-history filter.

Setting and finite-window oracle. Throughout this appendix, the SSM is fixed at the data-generating model $\theta = \theta^\star$, including its initial-state prior. All conditional distributions are taken under $p^\star$. We assume that the required conditional densities exist and that the displayed expectations and KL divergences are finite. Integrals are replaced by sums for finite state spaces. For $1 \le \tau \le T$, write

$$W_t = Y_{\max\{1,t-\tau+1\}:t}, \qquad f_t = p^\star(X_t \mid Y_{1:t}), \qquad a_t^\tau = p^\star(X_t \mid W_t). \tag{68}$$

The masked-window implementation in the main text carries the same information as W_t. Define the backward oracle and the one-step-updated window filter by

$$b_t^\tau = p^\star(X_{t-1} \mid X_t, W_{t-1}), \qquad t \ge 2, \tag{69}$$

$$r_t^\tau = p^\star(X_t \mid W_{t-1}, Y_t). \tag{70}$$

For $t > \tau$, r_t^τ conditions on $\tau + 1$ observations, whereas a_t^τ conditions on the most recent τ observations. Both are conditionals of the true joint law: the distribution at the start of each window is the corresponding marginal of that law, not an independently reset copy of p_0.

Let $\Delta_\tau^{\mathrm{or}}$ denote the expected gap obtained by substituting $q_t^F = a_t^\tau$ and $q_t^B = b_t^\tau$ into the recursive objective, with the usual initial-time boundary term. This oracle is defined independently of a neural parameterization. To transfer its bound to the shared NFF family, we will explicitly require simultaneous representability of these factors. The time indices matter: for nonstationary

data, finite-window conditionals can depend on absolute time even when two numerical observation windows coincide. A shared network without an absolute-time input is not automatically able to represent all of them.

Proposition 1 (Exact finite-window oracle gap)**.** *For the oracle above,*

$$\Delta_\tau^{\mathrm{or}} = \sum_{t=\tau+1}^{T} \left(A_t^\tau + B_t^\tau\right), \tag{71}$$

where

$$A_t^\tau = I_{p^\star}(Y_t; Y_{1:t-\tau-1} \mid Y_{t-\tau:t-1}), \tag{72}$$

$$B_t^\tau = \mathbb{E}_{p^\star}\mathrm{KL}(p^\star(X_t \mid Y_{t-\tau+1:t}) \,\|\, p^\star(X_t \mid Y_{t-\tau:t})). \tag{73}$$

An empty history has zero conditional mutual information, and an empty sum is zero. Moreover,

$$A_t^\tau \le \mathbb{E}_{p^\star}\mathrm{KL}(f_{t-1} \| a_{t-1}^\tau). \tag{74}$$

Proof. Under the oracle, the induced predictor is

$$\begin{aligned}\widetilde{p}_t(Y_t \mid W_{t-1}) &= \iint p_{\theta^\star}^{\mathrm{obs}}(Y_t \mid x_t) p_{\theta^\star}^{\mathrm{dyn}}(x_t \mid x_{t-1}) a_{t-1}^\tau(x_{t-1})\, dx_{t-1}\, dx_t \\ &= p^\star(Y_t \mid W_{t-1}).\end{aligned} \tag{75}$$

Its adjacent-state posterior is

$$\widetilde{\pi}_t(x_t, x_{t-1}) = r_t^\tau(x_t) b_t^\tau(x_{t-1} \mid x_t). \tag{76}$$

Here the SSM conditional independence implies that conditioning additionally on Y_t does not change the distribution of X_{t-1} given (X_t, W_{t-1}). Consequently, the shared backward factor cancels in the KL divergence:

$$\mathrm{KL}(a_t^\tau b_t^\tau \| \widetilde{\pi}_t) = \mathrm{KL}(a_t^\tau \| r_t^\tau). \tag{77}$$

Substituting into the expected-gap identity of Appendix A gives the local gap

$$\mathbb{E}_{p^\star}\mathrm{KL}(p^\star(Y_t \mid Y_{1:t-1}) \,\|\, p^\star(Y_t \mid W_{t-1})) + \mathbb{E}_{p^\star}\mathrm{KL}(a_t^\tau \| r_t^\tau). \tag{78}$$

For $t > \tau$, the first term is the conditional mutual information in equation 72, and the second is equation 73. For $2 \le t \le \tau$, both windows contain the relevant complete histories, so both terms vanish. The initial-time term also vanishes because a_1^τ is the exact posterior under the correct prior. Summation proves equation 71.

For each fixed history $Y_{1:t-1}$, the two predictive distributions in equation 78 are the images of f_{t-1} and a_{t-1}^τ under the same Markov kernel

$$K(x_{t-1}, dy_t) = \int p_{\theta^\star}^{\mathrm{dyn}}(dx_t \mid x_{t-1}) p_{\theta^\star}^{\mathrm{obs}}(dy_t \mid x_t). \tag{79}$$

The data-processing inequality for KL, followed by averaging over histories, proves equation 74. □

The first term in equation 71 measures predictive information lost by discarding old observations. The second measures the inconsistency created when the window shifts and its oldest observation is removed. Importantly, the latter is a KL from the shorter-window filter to the longer-window filter. A forward-KL forgetting bound alone does not automatically control this term.

Corollary 1 (Exponential bound on the attainable population gap)**.** *Suppose a single shared recognition parameter vector ϕ_τ represents a_t^τ for every t and b_t^τ for every $t \ge 2$ on the relevant inputs. Suppose also that constants $C_F, C_R < \infty$ and $0 < \rho < 1$, uniform in t and the window lengths under consideration, satisfy*

$$\mathbb{E}_{p^\star}\mathrm{KL}(f_{t-1} \| a_{t-1}^\tau) \le C_F \rho^\tau, \tag{80}$$

$$\mathbb{E}_{p^\star}\mathrm{KL}(a_t^\tau \| r_t^\tau) \le C_R \rho^\tau, \tag{81}$$

for $t > \tau$. Then

$$0 \le \inf_\phi \Delta_{\mathrm{NFF}}(\theta^\star, \phi) \le \Delta_\tau^{\mathrm{or}} \le (T-\tau)(C_F + C_R)\rho^\tau. \tag{82}$$

In particular, the time-averaged gap obeys

$$\frac{1}{T} \inf_\phi \Delta_{\mathrm{NFF}}(\theta^\star, \phi) \le (C_F + C_R)\rho^\tau. \tag{83}$$

Proof. The population gap is nonnegative for every recognition model by Appendix A. Simultaneous representability provides the candidate ϕ_τ with gap $\Delta_{\mathrm{NFF}}(\theta^\star, \phi_\tau) = \Delta_\tau^{\mathrm{or}}$. Apply equation 74 and the two forgetting bounds to equation 71, and sum over $T - \tau$ times. □

The total gap accumulates with sequence length; the time-averaged gap decreases exponentially with the window length. This is an attainable-gap bound, not an upper bound for every trained network. It remains valid if the oracle factors can be approached by a sequence of shared recognition models whose population gaps converge to $\Delta_\tau^{\mathrm{or}}$; the existence of a single exact representation is then unnecessary. Ordinary universal approximation without control of the relevant log-density expectations does not by itself establish this convergence.

A finite-state sufficient condition. The following statement makes explicit how total-variation forgetting can yield the required KL control. It concerns the recursive objective on a finite state space with respect to counting measure; it does not assert that the continuous normalizing flows used in our experiments parameterize discrete distributions.

Corollary 2 (Conversion from total variation to KL)**.** *Let the state space be finite, and define* $\|P - Q\|_{\mathrm{TV}} = \frac{1}{2}\sum_x |P(x) - Q(x)|$. *Suppose that, almost surely for every* $t > \tau$,

$$\|f_{t-1} - a_{t-1}^\tau\|_{\mathrm{TV}} \le C\rho^\tau, \qquad \|a_t^\tau - r_t^\tau\|_{\mathrm{TV}} \le C\rho^\tau, \tag{84}$$

$$a_{t-1}^\tau(x) \ge \eta, \qquad r_t^\tau(x) \ge \eta \quad \textit{for every } x, \tag{85}$$

where $C < \infty$, $0 < \rho < 1$, *and* $\eta > 0$ *are uniform in time and the window lengths under consideration. Then*

$$\Delta_\tau^{\mathrm{or}} \le \frac{8C^2}{\eta}(T - \tau)\rho^{2\tau}. \tag{86}$$

If the simultaneous representability condition of Corollary 1 holds, the same expression also bounds the best attainable NFF population gap from above.

Proof. For probability mass functions with $Q(x) \ge \eta$, the inequality $\log u \le u - 1$ yields

$$\begin{aligned} \mathrm{KL}(P\|Q) &\le \sum_x \frac{(P(x) - Q(x))^2}{Q(x)} \\ &\le \frac{1}{\eta}\left(\sum_x |P(x) - Q(x)|\right)^2 = \frac{4}{\eta}\|P - Q\|_{\mathrm{TV}}^2. \end{aligned} \tag{87}$$

Use this inequality for $(P, Q) = (f_{t-1}, a_{t-1}^\tau)$ and $(P, Q) = (a_t^\tau, r_t^\tau)$, and apply Proposition 1. □

The positive-mass condition is a sufficient condition, not a necessary one. For example, let the finite-state transition probabilities obey $P(x' \mid x) \ge \alpha > 0$, let the initial prior obey $p_0(x) \ge \alpha_0 > 0$, and suppose the observation likelihoods satisfy

$$0 < \min_x g(y \mid x), \qquad \frac{\max_x g(y \mid x)}{\min_x g(y \mid x)} \le R < \infty \tag{88}$$

for all relevant observations. Prediction assigns at least α to each state, and each Bayes update then assigns at least α/R. At the initial time the analogous lower bound is α_0/R. Consequently, the filters in equation 85 have the common lower bound $\eta = \min\{\alpha, \alpha_0\}/R$. The exponential total-variation bounds in equation 84 must additionally be established for the filtering recursion. A uniform filter-stability bound over admissible window-start distributions can control both comparisons: each pair processes the same recent observations and differs in the information incorporated into its starting distribution.

Exponential forgetting of finite-state hidden Markov filters under appropriate assumptions is studied by Le Gland & Mevel (2000). The distinction between mixing of the unobserved Markov chain and stability of the observation-conditioned filter is discussed by Douc et al. (2007). Finite state cardinality alone does not establish the assumptions above, and total-variation contraction alone does not give the KL upper bound without an additional condition such as equation 85.

Scope for NFF. Proposition 1 and Corollary 1 apply to continuous state spaces as well, provided their integrability, forgetting, and representability assumptions hold. The finite-state corollary is an illustrative sufficient condition, not a verification of these assumptions for the Lorenz–96 or Kolmogorov experiments. In particular, the constants can depend on state dimension, transition noise, and the observation model, and need not give a useful numerical bound in high dimensions.

With approximate rather than exact forgetting, adding future-observation CRPS need not preserve the minimizer set: the zero-gap realizability argument of Appendix B no longer applies automatically. The bounds here isolate the recursive-ELBO gap at the correct SSM. Approximation by a restricted flow family, finite-data estimation, optimization error, and parameter learning require additional analysis before these bounds can be interpreted as guarantees for a particular fitted model or for its full-history posterior error.

D SETTINGS FOR THE EXPERIMENTS

D.1 OSCILLATOR

State-space model and data generation. The state is ordered as $X_t = (q_{1,t}, p_{1,t}, q_{2,t}, p_{2,t}) \in \mathbb{R}^4$. Between observations, it evolves according to

$$\begin{aligned}
\dot{q}_1 &= p_1, \\
\dot{p}_1 &= -\omega_1^2 q_1 - \gamma p_1 + \kappa q_2 + \alpha q_1 q_2 + f_1, \\
\dot{q}_2 &= p_2, \\
\dot{p}_2 &= -\omega_2^2 q_2 - \gamma p_2 + \kappa q_1 + \alpha \left(q_1^2 - \frac{1}{2} q_2^2 \right) + f_2,
\end{aligned} \tag{89}$$

where

$$(\omega_1, \omega_2, \gamma, \kappa, \alpha, f_1, f_2) = (1.0, 1.37, 0.08, 0.18, 0.08, 0.35, -0.20). \tag{90}$$

The unequal frequencies, nonlinear coupling, and nonzero forcing break the coordinatewise sign symmetries left unresolved by the squared observation operator.

We integrate equation 89 using fourth-order Runge–Kutta with internal step size 0.05. Ten internal steps define one observation interval $\Delta = 0.5$. Let Φ_Δ denote the resulting deterministic map. The discrete-time SSM is

$$\begin{aligned}
&X_1 \sim \mathcal{N}(0, I_4), && \\
&X_t = \Phi_\Delta(X_{t-1}) + \eta_t, && \eta_t \sim \mathcal{N}(0, 0.1^2 I_4), \\
&Y_t = X_t \odot X_t + \epsilon_t, && \epsilon_t \sim \mathcal{N}(0, 0.1^2 I_4),
\end{aligned} \tag{91}$$

with all noise draws independent across coordinates, times, and trajectories. Each sequence contains $T = 128$ states, and no spin-up period is used. The implementation clips intermediate RK4 states and noisy states coordinatewise to $[-12, 12]$ as a numerical safeguard.

Training and test sequences are generated online. Each dataset index determines an independent random-number stream and returns a block of independently generated trajectories. The latent trajectories are retained only for evaluation and visualization. They are not supplied to the training objective.

Finite-window representation. We train separate NFF models for $\tau \in \{1, 4, 16\}$. For each target time t, observations are folded into a right-aligned tensor $\bar{Y}_t \in \mathbb{R}^{\tau \times 4}$ and a validity mask $M_t \in \{0, 1\}^{\tau \times 4}$. Missing observations near the beginning of a sequence are filled with zero and assigned mask value zero. Each lag is represented by the 12-dimensional feature

$$\psi(\bar{Y}_t, M_t) = \left[M_t \odot \frac{\bar{Y}_t}{4},\ M_t \odot \frac{\operatorname{sign}(\bar{Y}_t)\sqrt{|\bar{Y}_t| + 10^{-6}}}{2},\ M_t \right]. \tag{92}$$

The first component retains the noisy squared observations, while the second provides the encoder with an approximate amplitude scale.

Context encoders. Both recognition factors use pre-normalized MLP-Mixer encoders. Each Mixer block applies a residual token-mixing MLP over the τ temporal lags, followed by a residual channel-mixing MLP. The token-mixing hidden width is $\max\{4, 2\tau\}$, and the channel-mixing hidden width is four times the Mixer width. After the Mixer stack, we concatenate the representation of the current-time token, the masked temporal mean, and an eight-dimensional embedding of the valid window length.

For the filtering factor $q_\phi^F(X_t \mid W_t)$, the Mixer has width 128 and six blocks. A two-layer head maps the pooled representation to a 64-dimensional flow context. For the backward factor $q_\phi^B(X_{t-1} \mid X_t, W_{t-1})$, the observation Mixer has width 64 and three blocks. The current state is represented using

$$[X_t/2,\ X_t \odot X_t/4]\,, \tag{93}$$

embedded into 64 dimensions, and fused with the observation representation to produce a second 64-dimensional flow context. The filtering and backward encoders do not share parameters.

Conditional normalizing flows. Both recognition densities are represented by conditional RealNVP flows with standard Gaussian base distributions. The filtering flow uses eight affine coupling layers with conditioner width 256, while the backward flow uses six coupling layers with conditioner width 128. The coupling-layer log scale and translation are smoothly bounded by 2 and 3, respectively.

For the filtering flow, the unconstrained flow output u is mapped to the physical state through

$$x = 4\tanh(u/4). \tag{94}$$

This defines an exact density on $(-4, 4)^4$, including the Jacobian of equation 94.

The backward factor uses a residual parameterization adapted to the short transition interval. Given its unconstrained flow output u_B and context c_B, we define

$$\begin{aligned} r &= 1.5\tanh(u_B/1.5), \\ \delta(c_B) &= 2\tanh\left(\widetilde{\delta}(c_B)/2\right), \\ X_{t-1} &= 4\tanh\left(\frac{X_t + \delta(c_B) + r}{4}\right), \end{aligned} \tag{95}$$

where $\widetilde{\delta}$ is a learned two-layer context network. All Jacobians in equation 95 are included in $\log q_\phi^B$. The physical dynamics parameters and the process and observation noise variances are fixed at their data-generating values.

Training objective. The minibatch objective is

$$\widehat{\mathcal{J}}(\phi) = -\widehat{\mathcal{L}}_{\mathrm{NFF}}(\phi) + 100\,\widehat{\mathcal{R}}_{\mathrm{CRPS}}(\phi). \tag{96}$$

We use one Monte Carlo sample for each local recursive-ELBO term. For the future-observation score, four independent samples are drawn from $q_\phi^F(X_t \mid W_t)$ and propagated through the prescribed stochastic transition for horizons $h = 1, \ldots, 8$. Independent process-noise draws are used for all members and rollout steps. The future observations are verification targets only and are never provided to the filtering encoder.

The expectation over additive Gaussian observation noise in the fair finite-ensemble CRPS is evaluated analytically using folded-normal expectations. This Rao–Blackwellization retains the expected CRPS while reducing Monte Carlo variance. We use forecast origins $t = 1, \ldots, T - 8$, so all eight horizons are available. Writing $\widehat{c}^{\mathrm{RB}}_{b,t,h,j}$ for the resulting scalar score, we use

$$\widehat{R}_{\mathrm{CRPS}} = \frac{1}{BT}\sum_{b=1}^{B}\sum_{t=1}^{T-8}\sum_{h=1}^{8}\sum_{j=1}^{4}\widehat{c}^{\mathrm{RB}}_{b,t,h,j},$$

where B is the trajectory batch size. The recursive-ELBO estimate is likewise averaged over trajectories and target times. Since the generative model is fixed in this experiment, both objective terms optimize only the recognition-model parameters.

Table 3: NFF training settings for the coupled-oscillator experiment. The batch column gives the number of independently generated trajectories in one optimizer update.

τ	batch	updates	peak lr	CRPS horizons	CRPS members	Train duration
1	512	10^5	3×10^{-4}	8	4	5 hours on an RTX5090
4	512	10^5	3×10^{-4}	8	4	7 hours on an RTX5090
16	128	10^5	3×10^{-4}	8	4	6 hours on an RTX5090

We use Adam with zero weight decay and clip the global gradient norm at 1. The first 1% of the 100,000 updates linearly warms the learning rate from zero to 3×10^{-4}. The learning rate is then set to 3×10^{-4}, 10^{-4}, 3×10^{-5}, and 10^{-5} over successive training intervals with boundaries at 25%, 50%, and 75% of the total updates. Mixed-precision arithmetic is not used. Five independently initialized models are trained for each value of τ.

Window-reset evaluation protocol. For a window length τ, define

$$s_\tau = T - \tau + 1. \tag{97}$$

All methods use only the observations $Y_{s_\tau:T}$. The corresponding window model is initialized at its first time from $p_0(X_{s_\tau}) = \mathcal{N}(0, I_4)$:

$$p_\tau(X_{s_\tau:T}, Y_{s_\tau:T}) = p_0(X_{s_\tau}) \prod_{t=s_\tau+1}^{T} p^{\mathrm{dyn}}(X_t \mid X_{t-1}) \prod_{t=s_\tau}^{T} p^{\mathrm{obs}}(Y_t \mid X_t). \tag{98}$$

The window target is $p_\tau(X_T \mid Y_{T-\tau+1:T})$. The numerical PF reference additionally restricts the state to $[-4, 4]^4$ at every observation time in the window, as detailed below.

NFF is queried with the folded window containing exactly $Y_{s_\tau:T}$. The PF and classical filters use $\mathcal{N}(0, I_4)$ as the window-start prior and assimilate Y_{s_τ} immediately, without an artificial transition before the first analysis. Each later step consists of one stochastic forecast over $\Delta = 0.5$, followed by assimilation of the next observation. All methods are evaluated against the same numerical PF reference using exactly $Y_{s_\tau:T}$. PF and the classical filters explicitly restart from the prescribed window-start prior, whereas NFF is trained across full sequences with the shared finite-window recursive ELBO and future-observation CRPS. This objective does not enforce an independent prior reset at each window boundary, so agreement with the reference is assessed empirically rather than imposed by the training construction. Figure 1 shows close agreement nonetheless: the mean (ED, W_1) is $(0.029, 0.040)$ at $\tau = 4$ and $(0.003, 0.012)$ at $\tau = 16$. These are total discrepancies from the common reference, including any departure associated with the training construction.

Particle-filter reference. For each held-out trajectory and each τ, the reference distribution is computed using 10^6 particles. At the first window time s_τ, the reset prior and squared likelihood factorize across coordinates:

$$p(x_d \mid y_d) \propto \exp\left[-\frac{x_d^2}{2} - \frac{(y_d - x_d^2)^2}{2(0.1)^2}\right]. \tag{99}$$

We evaluate each one-dimensional density in equation 99 on a 40,001-point grid over $[-4, 4]$ and sample from its numerical inverse CDF. This coordinate range matches the physical support of the NFF flows. The same coordinate restriction is enforced after each subsequent forecast: particles outside $[-4, 4]^4$ receive zero weight before normalization and resampling.

For every later observation, particles are propagated using the same RK4 map and process-noise distribution as the data-generating model, weighted by

$$p(Y_t \mid X_t) = \mathcal{N}\left(Y_t; X_t \odot X_t, 0.1^2 I_4\right), \tag{100}$$

and systematically resampled. Resampling is performed after every subsequent analysis, producing an equal-weight final particle cloud. For each held-out trajectory and τ, the same PF realization is used when evaluating NFF, EnKF, PFF, and EnSF.

Ensemble Kalman filter. We use a stochastic perturbed-observation EnKF with 1000 members. At each analysis, the nonlinear forecast observations are

$$H_t^{(i)} = X_{t,\mathrm{f}}^{(i)} \odot X_{t,\mathrm{f}}^{(i)}. \tag{101}$$

The Kalman gain is computed from the sample state-observation cross-covariance and forecast-observation covariance. We add $10^{-7} I_4$ to the innovation covariance before inversion. Independent $\mathcal{N}(0, 0.1^2 I_4)$ perturbations are added to the observed vector in the analysis update.

Ensemble score filter. EnSF also uses 1000 forecast members. At diffusion time $\lambda \in [0, 1]$, its forecast score is evaluated as the exact score of the equally weighted Gaussian mixture

$$p_\lambda(x) = \frac{1}{1000} \sum_{i=1}^{1000} \mathcal{N}\left(x; a(\lambda) X_{\mathrm{f}}^{(i)}, b(\lambda)^2 I_4\right), \tag{102}$$

where

$$a(\lambda) = 1 - \lambda(1 - 0.5), \qquad b(\lambda)^2 = 0.025 + \lambda(1 - 0.025). \tag{103}$$

The reverse process is integrated from $\lambda = 1$ to $\lambda = 0$ using 20 uniform Euler–Maruyama steps. The squared observation operator gives the analytic likelihood score

$$\nabla_x \log p(Y_t \mid x) = \frac{2x \odot (Y_t - x \odot x)}{0.1^2}, \tag{104}$$

which is multiplied by $1 - \lambda$ along the reverse path. Each score coordinate is clipped to $[-1000, 1000]$ for numerical stability.

Particle flow filter. The PFF baseline is a localized exact Daum–Huang flow with 1000 particles. For each particle $x^{(i)}$, the squared observation operator is linearized locally as

$$h(x) \simeq H_i x + e_i, \qquad H_i = \mathrm{diag}(2x^{(i)}), \qquad e_i = -x^{(i)} \odot x^{(i)}. \tag{105}$$

The forecast mean and covariance are held fixed during each analysis, while the linearization in equation 105 is particle-specific. The pseudo-time flow is integrated using 64 uniform steps. We add $10^{-5} I_4$ to the forecast covariance, use covariance inflation 1, and cap the Euclidean norm of each particle drift at 100. The transported particles retain equal weights, with no subsequent importance correction or resampling.

Distributional metrics. For NFF, we draw 10^5 independent samples from $q_\phi^F(X_T \mid W_T)$. The classical methods return 1000 final analysis members, while the PF reference contains 10^6 particles. Let $\{x_i\}_{i=1}^n$ denote samples from a tested method and $\{z_j\}_{j=1}^m$ samples from the corresponding PF reference. The reported multivariate energy distance is

$$\widehat{\mathrm{ED}} = \frac{2}{nm} \sum_{i=1}^{n} \sum_{j=1}^{m} \|x_i - z_j\|_2 - \frac{1}{n^2} \sum_{i=1}^{n} \sum_{i'=1}^{n} \|x_i - x_{i'}\|_2 - \frac{1}{m^2} \sum_{j=1}^{m} \sum_{j'=1}^{m} \|z_j - z_{j'}\|_2. \tag{106}$$

It is computed in the full four-dimensional state space using at most 4096 randomly selected samples from each distribution.

Marginal agreement is measured by

$$W_1 = \frac{1}{4} \sum_{d=1}^{4} W_1\left(\widehat{P}_d, \widehat{P}_d^{\mathrm{PF}}\right), \tag{107}$$

where $\widehat{P}_d$ and $\widehat{P}_d^{\mathrm{PF}}$ are the empirical distributions of coordinate d. Up to 10^5 samples from each distribution are used for equation 107; all available samples are used when a method returns fewer members. Metric subsampling seeds are held fixed across methods.

Aggregation and visualization. The quantitative results use ten prespecified held-out trajectories. For each NFF training seed or classical-filter random seed, ED and W_1 are first averaged over the ten trajectories. The table in Fig. 1 reports the mean and standard deviation of these trajectory-averaged values over five independent seeds.

The marginal curves in Fig. 1 use one prespecified held-out trajectory and 180 equal-width bins on $[-3, 3]$. The vertical truth marker is included only to orient the reader. Latent states are not used to train NFF, initialize any filter, perform an analysis update, or compute ED and W_1.

Table 4: Main NFF settings for the 40-dimensional Lorenz–96 experiment.

Quantity	Setting
State dimension, sequence length	40, 512
F, Δt, internal step	8, 0.03, 0.01
σ_{dyn}, σ_{obs}	1, 1
Prior	$\mathcal{N}(0, 5^2 I_{40})$
Memory length	$\tau = 16$
Spatial tokens per time	$m = 4$
Transformer	width 256, 6 blocks, 4 heads
Filtering flow	8 RealNVP layers, hidden width 1024
Backward flow	4 RealNVP layers, hidden width 512
Future-observation CRPS	horizons 1:4, 4 particles, weight 10
Training batch and updates	32 trajectories; 160,000 updates (nonlinear), 80,000 updates (linear)
Nonlinear-observation learning rate	3×10^{-4}; ELBO-only uses 10^{-4}
Linear-observation learning rate	10^{-4} for every ablation
Train duration	20 hours (nonlinear), 10 hours (linear) on an NVIDIA GH200 Superchip

D.2 LORENZ96

Data generation and observation models. We use the 40-dimensional Lorenz–96 system with cyclic indexing and forcing $F = 8$. One observation interval has length $\Delta t = 0.03$ and is integrated with fourth-order Runge–Kutta using an internal step of 0.01. After each observation interval, independent Gaussian process noise with standard deviation $\sigma_{\mathrm{dyn}} = 1$ is added to every state coordinate. The raw initial condition is sampled coordinatewise from $\mathcal{U}[-10, 10]$ and advanced for 50 observation intervals before storage, including process noise during spin-up. We use the first $T = 512$ stored states after this spin-up. We consider

$$Y_t = h(X_t) + \epsilon_t, \qquad \epsilon_t \sim \mathcal{N}(0, I_{40}), \tag{108}$$

with either $h(x) = x$ or the componentwise nonlinear operator $h(x) = |x|$. The nonlinear operator preserves the observation dimension but leaves the sign of every coordinate ambiguous at an individual time. For each of five runs, the training set contains 10,000 NPZ files with 512 independent trajectories per file. The five runs use disjoint file-index ranges. Files are processed in chunks of 32 trajectories, so a complete pass through a training set contains 160,000 gradient updates. Each run is evaluated on one independent file containing 512 trajectories. Simulated states are used only for evaluation. The initial prior in the prescribed state-space model and in every classical filter is $p_0(X_1) = \mathcal{N}(0, 5^2 I_{40})$; no filtering method is initialized from the simulated state. This prescribed inference prior is not assumed to equal the distribution of the first stored state after spin-up.

NFF architecture. NFF uses memory length $\tau = 16$. For each target time, the observations are folded into a tensor of shape $(B, \tau, 40)$ together with a binary validity mask. The spatial Transformer divides each 40-dimensional observation into $m = 4$ contiguous blocks of 10 coordinates, producing $4\tau = 64$ tokens per window. It applies full bidirectional self-attention with model width 256, six Transformer blocks, four attention heads, feed-forward width 512, sinusoidal time and space encodings, zero dropout, and output context dimension 640. Value and mask tokens are embedded separately. The observation encoder is shared by the filtering and backward factors. The filtering factor $q_\phi^F(X_t \mid W_t)$ is an eight-layer conditional RealNVP with hidden width 1024. The backward factor $q_\phi^B(X_{t-1} \mid X_t, W_{t-1})$ uses four RealNVP layers with hidden width 512. Both use alternating coupling masks, learned LU mixing, base standard deviation 5, and scale clamp 2. The Gaussian ablation replaces both flows by diagonal-Gaussian conditionals while retaining the same observation encoder and training objective.

Numerical safeguards. The NFF transition mean is evaluated in float32. Its input and each internal RK4 result are clipped coordinatewise to $[-30, 30]$, with non-finite values replaced for numerical stability. Gaussian process noise is added after the complete observation interval. These safeguards modify the numerical transition mean, not the support of the filtering density.

Objectives and optimization. We compare the combined recursive-ELBO and CRPS objective with a flow posterior (E+C, F), the same objective with diagonal-Gaussian conditionals (E+C, G), ELBO-only flow (E, F), and CRPS-only flow (C, F). Each update uses $B = 32$ trajectories of length $T = 512$. Let $\widehat{L}_{b,t}$ denote the one-sample local recursive-ELBO estimate, including the initial-time boundary term at $t = 1$. For predictive CRPS, $K = 4$ independent filtering samples are propagated with independent process noise over horizons $h = 1, \dots, 4$. Only origins $t = 1, \dots, T-4$ are used, so all four verification horizons are available at every scored origin.

Let $\widehat{c}^{\mathrm{RB}}_{b,t,h,j}$ denote the fair scalar CRPS estimate for observation coordinate j. Additive Gaussian observation noise is integrated analytically through folded-normal expectations, including variance $2\sigma^2_{\mathrm{obs}}$ in member-pair differences. The implemented minibatch objective is

$$\widehat{J} = -\frac{\lambda_E}{BT}\sum_{b=1}^{B}\sum_{t=1}^{T}\widehat{L}_{b,t} + \frac{\lambda_{\mathrm{CRPS}}}{BT}\sum_{b=1}^{B}\sum_{t=1}^{T-4}\sum_{h=1}^{4}\sum_{j=1}^{40}\widehat{c}^{\mathrm{RB}}_{b,t,h,j}. \tag{109}$$

The weights $(\lambda_E, \lambda_{\mathrm{CRPS}})$ are $(1, 10)$ for both combined variants, $(1, 0)$ for ELBO-only, and $(0, 10)$ for CRPS-only. Thus, the ELBO is averaged over trajectories and times without state-dimension normalization. Predictive CRPS is summed over coordinates and horizons and divided by BT, rather than averaged over all scored scalar entries. Future observations are verification targets only. The physical model and noise levels are fixed, and rollout gradients update the recognition parameters.

For nonlinear observations, the three CRPS-containing variants use peak learning rate 3×10^{-4} and ELBO-only uses 10^{-4}. For linear observations, all four variants use 10^{-4}. We use AdamW with zero weight decay and global gradient-norm clipping at 1000. The learning rate increases linearly from zero to its peak during the first 1% of updates and remains constant thereafter. Neural-network computations use bfloat16 and TF32; numerical dynamics use float32. All nonlinear-observation variants are trained for 160,000 updates, whereas all linear-observation variants are trained for 80,000 updates. Reported results use the model at the end of the corresponding training budget.

Amortized ensemble filter. AmEnF uses the public `ConvEnAF2d` analysis core of McCabe and Brown (2021), pinned at commit `2565288`, with the prescribed Lorenz–96 simulator and observation model. The network has six periodic convolutions with kernel size 1×5, hidden width 64, non-affine LayerNorm over channel and spatial dimensions, LeakyReLU with slope 0.01, and channel-wise dropout probability 0.2. An input projection is followed by two two-layer residual blocks and a readout. Each member has six recurrent-memory channels. The 12 input channels contain the signed forecast state, raw observation, instantaneous Lorenz–96 tendency, three local covariance bands, and memory. The covariance bands contain the pointwise variance and covariances with the two nearest neighbors, computed from ensemble anomalies with denominator $M-1$. Predicted observations and innovations are not supplied as additional network inputs. The readout produces a state adjustment a, memory proposal p, and gates g, c, giving

$$X_a = \mathrm{sigmoid}(g) \odot X_f + a, \qquad m' = \mathrm{sigmoid}(c) \odot m + p.$$

Training uses $M = 16$ members and batches of 32 trajectories of length 512. Each sequence starts from $\mathcal{N}(0, 5^2 I_{40})$ and zero memory, independently of the simulated state. After assimilating Y_t, each member is propagated through the prescribed stochastic transition, and Y_{t+1} is scored before being assimilated. Forecasts use three RK4 substeps of length 0.01, internal-step state clipping at ± 30, and independent Gaussian process noise of standard deviation 1 per observation interval. For the nonlinear experiment, the predicted observation mean is

$$\widehat{Y}_{t+1,j} = \frac{1}{M}\sum_{i=1}^{M}\left|X^{(i)}_{t+1,f,j}\right|.$$

The loss is the squared error between this prediction and $Y_{t+1,j}$, normalized by $\sigma^2_{\mathrm{obs}} = 1$ and averaged over coordinates, batch trajectories, and forecast origins within each optimization segment. Thus, the absolute-value operator is applied before ensemble averaging. No latent-state loss, ELBO, or CRPS penalty is used. Forcing $F = 8$, both noise levels, and the zero sensor bias remain fixed; only the analysis network is learned.

We reuse the NFF data-loading and observation-noise routines. For each run, the highest-index file in its 10,000-file training pool is held out for monitoring. The remaining 9,999 files are visited in

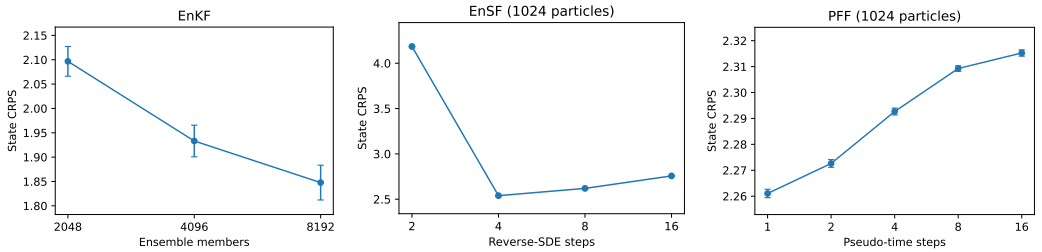


Figure 4: Classical-filter sensitivity for nonlinear Lorenz–96 observations. EnKF varies the ensemble size; EnSF and PFF use 1024 particles while varying the reverse-SDE and pseudo-time step counts. Points show the mean state CRPS and error bars show one standard deviation over five seed-level means. Each seed-level mean first averages 512 trajectories.

file-balanced order, taking one 32-trajectory chunk per file before revisiting a file. Backpropagation is truncated after at most 40 forecast–analysis steps; the ensemble and memory continue across truncation boundaries with their computational graphs detached. We train five independent models per observation setting, using 160,000 AdamW updates for nonlinear observations and 80,000 updates for linear observations, with zero weight decay, gradient-norm clipping at 1, and bfloat16 autocast for the network while the simulator runs in float32. The learning rate warms up linearly from 8×10^{-7} to 8×10^{-4} over the first 1% of updates, then decays exponentially to 8×10^{-5}.

At test time, AmEnF recursively assimilates all 512 observations with 128 members and active Monte Carlo dropout. Each run is evaluated on its paired 512-trajectory test file. State CRPS and calibration are computed over $t = 257, \ldots, 512$. At $T = 512$, the same ten supplementary signed-linear measurements with noise standard deviation 3 are assimilated by likelihood reweighting of the 128 analysis members, and post-update CRPS is scored on the remaining 30 coordinates. Results are aggregated over the five runs. Filtering NLL is reported as N/A because AmEnF returns an empirical ensemble rather than an evaluable normalized filtering density.

Classical filters. EnKF, EnSF, and PFF recursively assimilate all 512 observations. Each ensemble is initialized once from $\mathcal{N}(0, 5^2 I_{40})$, assimilates Y_1, and then alternates forecast and analysis steps through Y_{512}. The transition and observation noise match the data-generating model. EnKF uses the stochastic perturbed-observation update and evaluates the nonlinear observation operator on every forecast member. EnSF uses the empirical forecast ensemble in its prior score and integrates the analysis reverse SDE by Euler–Maruyama. PFF uses a local exact Daum–Huang flow specialized to the componentwise absolute-value observation, with forecast-covariance shrinkage 0.05 and geometrically spaced pseudo-time increments. We use 8192 members for EnKF. EnSF and PFF use 1024 particles, with four reverse-SDE steps for EnSF and one pseudo-time step for PFF. These settings were chosen from the disclosed sensitivity study in Figure 4, using state CRPS as the primary criterion. Four EnSF steps also minimize every other reported metric among the tested step counts. For PFF, one pseudo-time step minimizes state and future-observation CRPS and coverage error; additional steps slightly improve the KDE log score but worsen the primary CRPS metrics and increase runtime. EnKF continues to improve over 2048, 4096, and 8192 members. On the subsampled initial conditions used for the sensitivity study, the final doubling reduces state CRPS from 1.933 to 1.848. We therefore use 8192 members for the main comparison without claiming ensemble-size convergence. Table 1 evaluates this configuration on the full set of initial conditions, yielding 1.746 ± 0.053; the different mean reflects the different evaluation set.

Evaluation. Reported learned-model results use a fixed post-hoc evaluation protocol rather than the diagnostic settings embedded in the training configurations. Each run is evaluated on all 512 trajectories in its paired test file. State CRPS, calibration, and filtering NLL use every time $t = 257, \ldots, 512$. Selecting this reporting interval does not reset inference or discard the preceding causal observations.

State CRPS and calibration use 128 independent NFF posterior samples or 128 AmEnF analysis members per filtering distribution. State CRPS uses the fair finite-ensemble scalar formula, averaged

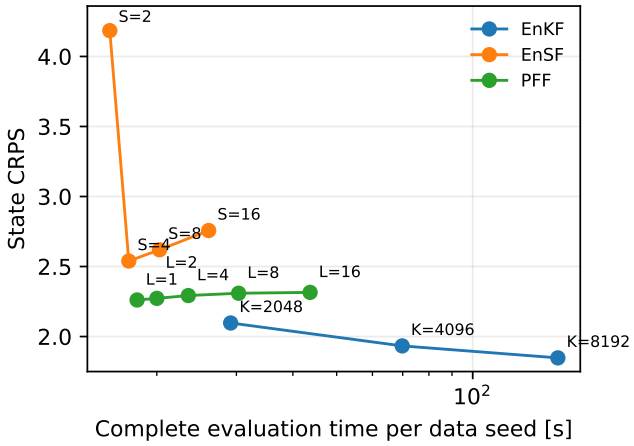


Figure 5: State CRPS against complete evaluation time for the classical-filter sensitivity study. Each timing covers 512 trajectories of length 512 and all reported metrics, using trajectory batch size 64 on one GPU. It measures batched throughput for this implementation and is not batch-one assimilation latency.

over state coordinates, scored times, and trajectories. For AmEnF, this specifies the scoring formula without assuming that interacting ensemble members are independent draws.

Calibration uses central marginal intervals at nominal levels $0.05, 0.10, \ldots, 0.95$. Coverage is pooled over trajectories, scored times, and coordinates within each file before computing

$$\text{Calib. error} = \frac{100}{0.95 - 0.05} \int_{0.05}^{0.95} \left| \widehat{C}(\alpha) - \alpha \right| \, d\alpha$$

by trapezoidal integration, giving percentage points. NFF and its Gaussian ablation evaluate $-\log q_\phi^F(X_t \mid W_t)/40$ directly in physical coordinates. AmEnF's filtering NLL is reported as N/A because its ensemble does not provide an evaluable normalized density. Each metric is computed within the paired test file before aggregation across the five runs.

Supplementary-observation update. At the final filtering time $T = 512$, ten fixed supplementary sensors provide signed linear measurements

$$z_j = X_{T,j} + \epsilon_j^{\text{add}}, \qquad \epsilon_j^{\text{add}} \sim \mathcal{N}(0, 3^2), \qquad j \in S,$$

where $S = \{0, 4, 8, \ldots, 36\}$ uses zero-based coordinate indices. The sensor offset is fixed at zero rather than averaged over cyclic shifts. Supplementary noise uses seed 20260826 and is keyed by trajectory and final-time indices, giving identical measurements across methods and trajectory batchings. These measurements are withheld from the filtering encoders and original analysis recursions.

NFF uses $M = 2048$ independently drawn filtering samples for this update, while AmEnF uses all 128 genuine analysis members. These supplementary-update budgets are distinct from the 128-sample budget used for marginal scoring. Members are not duplicated to equalize the budgets.

Given final filtering samples $\{x^{(i)}\}_{i=1}^M$, we assimilate the supplementary measurements by likelihood reweighting:

$$w_i = \frac{\exp\left[-\frac{1}{2(3^2)} \sum_{j \in S} (z_j - x_j^{(i)})^2\right]}{\sum_{\ell=1}^M \exp\left[-\frac{1}{2(3^2)} \sum_{j \in S} (z_j - x_j^{(\ell)})^2\right]}.$$

No resampling is performed. Let $U = \{0, \ldots, 39\} \setminus S$ denote the remaining 30 coordinates. The final column of Table 1 reports the post-update weighted empirical CRPS

$$\text{CRPS}_{\text{after},U} = \frac{1}{30} \sum_{j \in U} \left[\sum_{i=1}^M w_i |x_j^{(i)} - X_{T,j}| - \frac{1}{2} \sum_{i,\ell=1}^M w_i w_\ell |x_j^{(i)} - x_j^{(\ell)}| \right].$$

Table 5: Main NFF settings for the 16,384-dimensional Kolmogorov-flow experiment (E+C, F). Flow-block counts are listed from fine to coarse scales.

Quantity	Setting
State grid, sequence length	128×128 ($d = 16{,}384$), $T = 512$
True F_0, Re	1, 1000
Δt, internal step	0.05, 0.005 (RK4)
σ_{dyn}, σ_{obs}	0.1, 1
Observation network	64×64 dense linear; stride-4 sparse square
Initial-state prior	Learnable spectral Gaussian
Memory length	$\tau = 64$
Observation encoder	7 periodic residual blocks, width 96
Encoder kernels and context	3×3 kernels, 64 context channels
Filtering flow	4 scales, $(2, 3, 4, 4)$ RealNVP coupling blocks
Backward flow	4 scales, $(2, 2, 3, 3)$ RealNVP coupling blocks
Backward observation window	W_{t-1} (previous window)
Flow conditioners	Width 96, 2 residual blocks, 3×3 kernels
Observation and state scaling	$Y/10$, $X/5$; standard Gaussian flow bases
ELBO estimation	1 sample per target; state-dimension normalization
Future-observation CRPS	$h = 1, \ldots, 12$, 2 members, $\lambda_{\mathrm{CRPS}} = 30$
Targets per update	128 ELBO times; 32 sampled CRPS origins
Training batch and updates	1 trajectory, 80,000 updates
Initial $(F_0, \log \mathrm{Re}, b)$	$(0.1, \log 100, 0)$
Network, prior, and bias peak LR	3×10^{-4}
Physical-parameter peak LR	3×10^{-3}
Optimizer, weight decay, clipping	AdamW, 0, global gradient norm 1
Learning-rate schedule	5% warm-up; exponential decay to $0.1\times$ peak
Train duration	30 hours on an NVIDIA GH200 Grace Hopper Superchip

This is an absolute post-update score, not a CRPS gain, and uses weighted empirical CRPS rather than the fair finite-ensemble formula used for state CRPS. Here, "unobserved" means not directly measured by the supplementary sensors; all 40 coordinates are observed through the original absolute-value operator. For the coordinatewise gain plots in Appendix E.1.2, the pre-update score uses equal weights on the same filtering samples, and gains are defined as pre-update minus post-update CRPS.

D.3 Kolmogorov flow

State-space model and data generation. The state $X_t \in \mathbb{R}^{128\times 128}$ is the vorticity field on a periodic square of side 2π, giving $d = 16{,}384$ state coordinates. The prescribed simulator advances

$$\partial_s \omega = -\nabla \cdot (\mathbf{v}\omega) + \nu \Delta \omega - \nu_4 \Delta^2 \omega - 0.1\omega + F_0 G, \qquad \nu = \max\{\exp(-\log \mathrm{Re}), 10^{-8}\}, \quad (110)$$

where velocity is reconstructed from vorticity in Fourier space, with zero velocity at the zero Fourier mode. In zero-based array coordinates, the forcing template is $G_{r,c} = -4\cos(4 \cdot 2\pi r/128)$. Advection uses a flux-form pseudo-spectral discretization with two-thirds de-aliasing. The hyperviscosity is fixed at $\nu_4 = 0.1/(\delta t\, k_{\mathrm{cut}}^4)$, where $k_{\mathrm{cut}} = (2/3)\, 64$ and $\delta t = 0.005$. Ten fourth-order Runge–Kutta steps define one observation interval $\Delta t = 0.05$. The data-generating parameters are $F_0 = 1$ and $\mathrm{Re} = 1000$.

Writing the resulting deterministic map as $\Phi_{\Delta t,\theta}$, the discrete transition is

$$X_t = \Phi_{\Delta t,\theta}(X_{t-1}) + \eta_t, \qquad \eta_t \sim \mathcal{N}(0, 0.1^2 I_d). \quad (111)$$

Process noise is independent across grid cells and observation intervals and is added after the complete interval, not after each internal RK4 step. Initial velocities use the JAX-CFD filtered-velocity initializer with maximum velocity 3 and peak wavenumber 4, and are converted to vorticity by centered periodic differences. Each trajectory is advanced for 400 observation intervals before storage, including process noise during spin-up. The optional process-noise filtering and state clipping are disabled in data generation. The experiments use the first $T = 512$ saved states, stored in float32.

The training pool contains 2500 files, indexed $0, \ldots, 2499$, with 128 trajectories per file. All five NFF runs use this pool with different initialization and data-order seeds. The five independent

test files have indices $1000000, \ldots, 1000004$, each containing 128 trajectories. Learned-model run $s \in \{1, \ldots, 5\}$ is evaluated on file $1000000+s-1$, with the classical methods evaluated on the same files. Simulated states are excluded from training batches and are used only to construct observations and to evaluate or visualize the inferred distributions. No filter is initialized from the simulated test state.

Observation network and sensor bias. Let $\mathcal{D} = \{(r, c) : 0 \le r, c < 64\}$ be the densely observed square. The sparse network consists of the lattice $r, c \in \{3, 7, \ldots, 127\}$ outside $\mathcal{D}$; denote these sparse-only locations by $\mathcal{S}$. The observations are

$$Y_{t,r,c} = \begin{cases} X_{t,r,c} + b_{r,c} + \epsilon_{t,r,c}, & (r,c) \in \mathcal{D}, \\ X_{t,r,c}^2 + b_{r,c} + \epsilon_{t,r,c}, & (r,c) \in \mathcal{S}, \end{cases} \qquad \epsilon_{t,r,c} \sim \mathcal{N}(0,1). \tag{112}$$

Thus, there are 4096 linear sensors and 768 sparse-only square sensors, giving 4864 observations per time. The linear operator takes precedence where the sparse lattice overlaps the dense square. Bias and noise are added after the observation operator. Negative noisy square observations are retained without clipping, square-root preprocessing, or sign assignment. Missing cells are excluded from observation-space losses.

The bias field is fixed in time and shared across trajectories and runs. Its entries are drawn independently from $\mathcal{N}(0,1)$ using NumPy's `default_rng(42)`. Original observation noise is deterministically keyed by $(314159265, \text{file index}, \text{trajectory index}, \text{time index})$ using NumPy `SeedSequence`, so overlapping windows and the learned methods use identical measurements. The classical filters use the corresponding oracle bias-corrected measurements, as described below.

Learned parameters and initial-state prior. The learned methods jointly estimate F_0, $\log \mathrm{Re}$, and the sensor-bias field, initialized at 0.1, $\log 100$, and zero, respectively. These are point estimates. The process and observation noise scales are fixed, and no variational physical-parameter distribution or physical-parameter KL penalty is introduced.

For NFF and AmEnF, the initial-state prior is a zero-mean periodic Gaussian with Fourier precision

$$P(k) = \left(a_0 + a_1 \ell_k + a_2 \ell_k^2\right) \left[1 + (B_0 - 1)\,\mathrm{sigmoid}\left(12(\|k\|_2 - 16)\right)\right], \tag{113}$$

$$\ell_k = 4 \sin^2 \frac{\pi k_r}{128} + 4 \sin^2 \frac{\pi k_c}{128}. \tag{114}$$

Using an orthonormal FFT, its log density is

$$\log p_0(X) = -\frac{1}{2} \sum_k P(k) |\widehat{X}_k|^2 + \frac{1}{2} \sum_k \log P(k) - \frac{d}{2} \log(2\pi). \tag{115}$$

The parameters (a_0, a_1, a_2, B_0) are initialized at $(10^{-3}, 1, 0.1, 6)$ and represented by trainable $\log a_0$, $\log a_1$, $\log a_2$, and $\log(B_0 - 1)$. The cutoff and sharpness remain fixed. In NFF, this prior enters only the initial-time boundary term; sampling an interior loss block does not reset the prior. The CRPS-only ablation omits the boundary-prior and backward-density terms, so parameters used exclusively by those terms receive no training signal. The spectral Gaussian is the prescribed prior family, not an assertion that the spun-up data distribution is exactly Gaussian.

NFF observation encoder. NFF uses memory length $\tau = 64$. Each observation history and validity mask have shape $(B, 64, 128, 128)$; the 64 value channels and 64 mask channels are concatenated before spatial encoding. Values are divided by 10, and unavailable entries are zero-filled. The mask distinguishes missing spatial measurements and temporal padding near the beginning of a sequence. The shared encoder has seven residual blocks of width 96, each using a periodic 3×3 convolution, one-group GroupNorm, and ReLU. Its two-channel output passes through a zero-initialized scaled-tanh head with a learned bound initialized at 10. Two 1×1 convolutions with widths $2 \to 64 \to 64$ and an intermediate SiLU activation produce the context field. Temporal lags are input channels; convolutions operate in space. No observation after the target filtering time enters the encoder.

Conditional densities and Gaussian ablation. The flow variants use multiscale conditional RealNVP densities. Four lossless 2×2 squeezing operations give spatial resolutions 64^2, 32^2, 16^2, and 8^2. The filtering flow uses $(2, 3, 4, 4)$ affine coupling blocks across these levels, and the backward

flow uses $(2, 2, 3, 3)$. Couplings use alternating channel masks and learned invertible 1×1 mixing. Each conditioner has width 96, two periodic residual blocks, 3×3 convolutions, GroupNorm, and SiLU activations. Log scales and translations are smoothly bounded by 1 and 10, respectively. Half of the channels are factored into the Gaussian base at each nonterminal level; no state coordinates are discarded. Observation contexts are average-pooled to each resolution, while the current state supplied to the backward flow is rearranged by lossless squeezing. Both flows use standard Gaussian bases and physical-state normalization $Z = X/5$, including the affine correction $-d \log 5$ in physical-space log densities.

The Gaussian ablation replaces both flows with conditional diagonal Gaussians while retaining the encoder. Each Gaussian conditioner has width 96 and two periodic residual blocks. Means are smoothly bounded by 20 in physical coordinates, and log standard deviations in normalized coordinates are constrained to $[-4, 2]$. These bounds restrict distribution parameters, not Gaussian support. The initial Gaussian conditionals are $\mathcal{N}(0, 25 I_d)$.

Backward-conditioning convention. We use the factorization of Section 3:

$$q_{\phi,t}(X_t, X_{t-1}) = q^F_\phi(X_t \mid W_t)\, q^B_\phi(X_{t-1} \mid X_t, W_{t-1}). \tag{116}$$

The shared observation encoder is applied to W_t for the filtering factor and to W_{t-1} for the backward factor and preceding filtering-density term $q^F_\phi(X_{t-1} \mid W_{t-1})$. The backward factor additionally receives X_t. The Gaussian ablation uses the same conditioning inputs.

NFF objectives and optimization. We compare combined flow (E+C, F), combined diagonal Gaussian (E+C, G), ELBO-only flow (E, F), and CRPS-only flow (C, F). One adjacent-state Monte Carlo sample is used per ELBO target time. For a sampled target block $\mathcal{B}$, the combined objective is

$$\widehat{J} = -\frac{1}{d|\mathcal{B}|} \sum_{t \in \mathcal{B}} \widehat{L}_t + 30 \widehat{R}_{\mathrm{CRPS}}. \tag{117}$$

The corresponding term is omitted in each single-objective ablation. Future-observation CRPS uses two independent filtering members and all horizons $h = 1, \ldots, 12$, with independent process-noise draws across members and rollout steps. One filtering draw is reused from the ELBO computation when available. Additive Gaussian observation noise is integrated analytically through folded-normal expectations in the fair CRPS estimator. Gradients propagate through the prescribed simulator and observation model to the recognition, physical, and bias parameters.

Each update uses one trajectory and samples one of the four aligned 128-time blocks starting at zero-based indices $0, 128, 256, 384$. All 128 target times enter the ELBO, while 32 origins are sampled uniformly without replacement for predictive CRPS. The full observation sequence remains available for preceding histories and valid future verification targets. Inverse-inclusion-probability weighting accounts for block and origin sampling, retaining the full-sequence average over valid origin–horizon–sensor entries. Random horizon truncation is disabled.

All four variants use 80,000 AdamW updates, zero weight decay, and global gradient-norm clipping at 1. Peak learning rates are 3×10^{-4} for recognition and spectral-prior parameters, 3×10^{-3} for $(F_0, \log \mathrm{Re})$, and 3×10^{-4} for sensor bias. The first 5% of updates linearly warm the learning rates from 10^{-3} times their peaks; exponential decay then reaches one tenth of the peak at the final update. Neural-network computations use bfloat16 autocast and TF32, while the differentiable FFT/RK4 simulator uses float32. Checkpoints are recorded every 25,000 updates and at completion. Reported results use the final 80,000-update checkpoint, not a checkpoint selected by state-reconstruction accuracy. This budget processes 80,000 one-trajectory training items rather than a complete pass through the 320,000-trajectory pool.

Amortized ensemble filter. AmEnF adapts the public `ConvEnAF2d` analysis core of McCabe and Brown (2021), pinned at commit `2565288`. The Kolmogorov network has six periodic 5×5 convolutions, hidden width 48, non-affine LayerNorm over channel and spatial dimensions, LeakyReLU with slope 0.01, spatial dropout probability 0.1, and four recurrent-memory channels per member. One input projection is followed by two two-layer residual blocks and a readout. Its 14 input channels contain the signed forecast state, observation, availability mask, instantaneous prescribed-model tendency, five local covariance channels, four memory channels, and one fixed

Table 6: Frozen classical-filter settings for Kolmogorov flow. Block sizes are spatial block side lengths in grid cells, not localization radii.

Method	Members	Analysis settings
EnKF	32	Block side 1; posterior std multiplier 1.04
EnSF	32	16 reverse-SDE steps
PFF	32	Block side 8; 16 pseudo-time steps

sensor-type channel. The covariance stencil contains the pointwise variance and covariances with the four nearest neighbors, computed directly from ensemble anomalies with denominator $M - 1$.

Availability is encoded as $+0.1$ at sensors and -0.1 elsewhere. The observation-input channel at missing cells is filled using a detached permutation of forecast members; these entries are excluded from the loss. The sensor-type channel distinguishes dense linear and sparse square sensors together with the availability mask. The readout produces a state adjustment a, memory proposal p, state gate g, and memory gate c, giving

$$X_a = \mathrm{sigmoid}(g) \odot X_f + a, \qquad m' = \mathrm{sigmoid}(c) \odot m + p. \tag{118}$$

The two-dimensional kernels, covariance stencil, and sensor metadata are explicit benchmark adaptations, not upstream Kolmogorov defaults.

Training uses eight members, one trajectory per batch, and truncated backpropagation through 16 forecast-analysis steps. Ensembles and memory continue across truncation boundaries, but their computational graphs are detached. Each sequence starts from the learned spectral prior and zero memory. After assimilating Y_t, members are propagated through the prescribed stochastic transition and Y_{t+1} is scored before being assimilated. The loss is the observed-coordinate mean squared error of

$$\widehat{Y}_{t+1,j} = \frac{1}{M} \sum_{i=1}^{M} h_j(X_{t+1,f}^{(i)}) + b_j, \tag{119}$$

normalized by σ_{obs}^2. In particular, the nonlinear operator is applied before averaging over members. No latent-state loss, ELBO, or CRPS training penalty is added.

AmEnF uses the same data-generation and observation routines, physical-parameter initial values, and spectral-prior family as NFF. Files $499, 999, 1499, 1999, 2499$ are reserved for observation-only monitoring. The remaining files are visited in shard-balanced order, taking one trajectory per file before revisiting a file. Training uses 100,000 AdamW updates, zero weight decay, gradient clipping at 1, and peak learning rates 3×10^{-4} for the analysis network and prior, 5×10^{-3} for physical parameters, and 5×10^{-4} for bias. Warm-up and decay use the same fractions as NFF. Training uses bfloat16 and activation checkpointing with preserved random-number state. Hyperparameters are fixed without a pilot search, and the final completed checkpoint is evaluated. These budgets do not imply equal data exposure or computation across methods. At test time, AmEnF recursively assimilates all 512 observations with 128 genuine members and active Monte Carlo dropout.

Classical filters and initialization. EnKF, EnSF, and PFF are evaluated as oracle baselines with the true physical parameters and sensor bias. After exact bias removal, their observations are $h(X_t) + \epsilon_t$. In the main comparison, all three methods use 32 members or particles and assimilate all 512 observations sequentially. Each starts from the first 32 fields of a common, separately prepared initial-ensemble bank, independent of the test-state realization. The same bank is reused across methods and test trajectories. The first observation is assimilated immediately, without an artificial forecast before the first analysis. Subsequent forecasts advance the full vorticity field and add independent process noise before the next analysis.

EnKF uses the block-local ensemble-space analysis with observation covariance $R = I$. Localization partitions the grid into non-overlapping blocks; it is not a distance-dependent covariance taper. Observations update only their own block. The selected block side is 1. After the analysis, anomalies about the analysis mean are multiplied by 1.04 before state clipping. This is a posterior standard-deviation adjustment, not inflation of the forecast covariance.

EnSF uses 16 reverse-SDE steps with the archived paired forecast-component approximation for the forecast score. It does not evaluate the exact score of the full empirical Gaussian mixture. The dense linear and sparse squared observation operators are passed to the analysis through their respective observed-coordinate groups.

PFF uses non-overlapping 8×8 analysis blocks and 16 pseudo-time steps. At each particle and pseudo-time step, a squared observation at coordinate j is linearized as

$$h_j(x_j) \simeq J_{i,j} x_j + e_{i,j}, \qquad J_{i,j} = 2x_j^{(i)}, \qquad e_{i,j} = -(x_j^{(i)})^2. \tag{120}$$

The corresponding linearized measurement is $Y_{t,j} - e_{i,j} = Y_{t,j} + (x_j^{(i)})^2$. Dense linear sensors retain $J_{i,j} = 1$ and $e_{i,j} = 0$. The returned particles have equal weights before the supplementary-observation diagnostic.

For the main comparison, the final launcher freezes the settings in Table 6 and executes only the 15 selected method–file tasks, without repeating the sensitivity search. Reproduction uses the final-run overlay together with its base classical-filter package. The runner serializes the inherited `Protocol`, including the remaining score-bridge, covariance-regularization, integration, and clipping settings, and records the initial-bank checksum. These inherited settings are not replaced by those of the Lorenz–96 implementation. The default final execution uses float32, trajectory batch size 4, and TF32 disabled. Diagnostic random-number streams are separate from filtering streams. Finite state clipping is retained and counted; a task fails if any pre-sanitization NaN or infinity is encountered rather than accepting a repaired result.

Classical-filter parameter sensitivity. We examine the classical-filter settings for the mixed observation operator with dense linear observations and sparse squared observations on the $\times 4$ sparse sensor grid. Unless ensemble size is varied explicitly, all configurations use 32 ensemble members or particles. The oracle physical parameters, exact sensor-bias removal, and truth-independent initialization are retained throughout.

Each configuration assimilates all 512 observations without resetting. To limit the search cost, we evaluate the same 16 trajectories from each of five test files, using zero-based trajectory indices $0, 8, \ldots, 120$. State CRPS averages over all state coordinates and the final 256 filtering times; calibration uses every fourth time within that interval. We first aggregate within each file and then report the mean and sample standard deviation across the five file-level results. State CRPS uses the finite-ensemble formula with pairwise denominator $M(M-1)$. This is a sensitivity analysis on a fixed subset of the test data, rather than a separate validation study. The main comparison instead uses all 128 trajectories per file and evaluates calibration at every reporting time.

EnKF localization and inflation. With 32 ensemble members, we evaluate the full Cartesian product of localization block sides $b \in \{1, 2, 4, 8\}$ and posterior standard-deviation multipliers $\alpha_\sigma \in \{1.00, 1.02, 1.04, 1.06\}$. The localization and inflation conventions are those described above; in particular, the posterior covariance multiplier before clipping is α_σ^2.

Figure 6 shows that block side 1 gives the lowest mean state CRPS at every tested multiplier. We retain $\alpha_\sigma = 1.04$ as an accuracy–calibration compromise. At $b = 1$, increasing the multiplier from 1.02 to 1.04 changes mean CRPS from 0.992 to 1.035, while reducing calibration error from 9.455 to 3.815 percentage points. Increasing it further to 1.06 reduces calibration error to 2.427 percentage points but raises CRPS to 1.173. Thus, $b = 1, \alpha_\sigma = 1.04$ balances state accuracy and calibration rather than minimizing either metric alone.

EnKF ensemble size. We compare $M \in \{8, 16, 32, 64, 128\}$ at the retained localization block side $b = 1$ and posterior standard-deviation multiplier $\alpha_\sigma = 1.04$, keeping all other filter and evaluation settings fixed. The M-member initial ensemble consists of the first M fields of the same truth-independent bank. Figure 8 (a) shows a strong reduction in state CRPS from $M = 8$ to $M = 32$, whereas increasing the ensemble to $M = 64$ or 128 does not further improve the mean state CRPS at these fixed settings and substantially increases filtering time. We therefore retain $M = 32$ for the main comparison as a favorable accuracy–runtime tradeoff. Localization and inflation are not retuned for each ensemble size, so this sweep assesses ensemble-size sensitivity rather than separately optimized EnKF configurations.

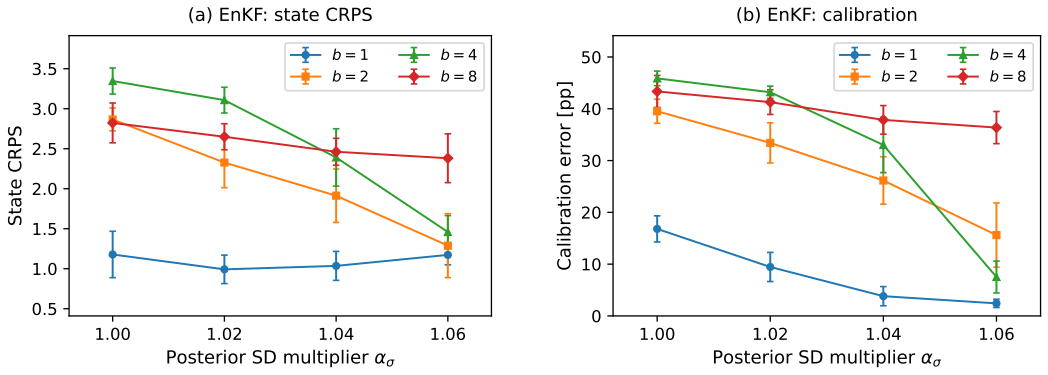


Figure 6: Joint EnKF localization/inflation search for Kolmogorov flow with 32 ensemble members. Left: state CRPS. Right: calibration error in percentage points. Both panels report the complete 4×4 grid of block sides and posterior standard-deviation multipliers on the fixed trajectory subset. The main comparison uses $b = 1, \alpha_\sigma = 1.04$.

EnSF reverse-SDE steps. We vary the reverse-SDE step count over $S \in \{2, 4, 8, 16\}$, with all remaining settings fixed. Figure 7 shows that $S = 16$ gives the lowest mean state CRPS among the tested step counts. The dependence is not monotone, since increasing from four to eight steps does not improve CRPS. Four steps give lower calibration error than sixteen, so retaining $S = 16$ prioritizes state CRPS rather than calibration alone.

PFF localization and pseudo-time steps. We vary the pseudo-time step count over $L \in \{1, 2, 4, 8, 16\}$ at block side $b = 8$, and separately compare block sides $b \in \{4, 8, 16\}$ at $L = 8$. At $b = 8$, mean state CRPS decreases throughout the tested step-count range, while calibration error remains low at both $L = 8$ and $L = 16$ (Figure 7).

The block-size comparison shows a complementary tradeoff. At $L = 8$, block side 8 improves both state CRPS and calibration relative to block side 4. Increasing the block side to 16 further reduces CRPS but substantially worsens calibration. We therefore retain $b = 8, L = 16$, combining the lowest tested CRPS at $b = 8$ with low calibration error. The block-size comparison is conditional on $L = 8$; these separate sweeps do not establish joint optimality over block side and pseudo-time step count.

Computational cost. Figure 8 shows EnKF runtime as a function of ensemble size at $b = 1, \alpha_\sigma = 1.04$, together with the runtime–accuracy tradeoffs for the EnSF and PFF step-count sweeps. Timing includes forecasting, process-noise sampling, analysis, and numerical-safety instrumentation, but excludes prior preparation, data loading, observation construction, and diagnostic evaluation. The reported milliseconds per trajectory and assimilation cycle measure batched throughput rather than single-trajectory latency. These measurements describe the evaluated implementations and configurations; the sweeps do not establish ensemble-size or solver-step convergence beyond the tested ranges.

Filtering metrics. All methods are evaluated on all 128 trajectories in each paired test file. State CRPS, marginal calibration, and available filter NLL/dim use every time $t = 257, \ldots, 512$, corresponding to zero-based indices $256, \ldots, 511$. In particular, the final classical calibration stride is 1. NFF uses its 64-step observation history, whereas AmEnF and the classical filters retain their full causal recursions; selecting the reporting interval does not reset any filter.

Marginal scores use 64 NFF posterior draws, the first 64 of AmEnF's 128 propagated members, or all 32 classical analysis members. Post-hoc NFF inference uses bfloat16 autocast, while AmEnF and classical evaluation use float32. State CRPS uses the fair finite-ensemble scalar formula and is averaged over all 16,384 state coordinates and scored times. For interacting ensembles, this specifies the scoring formula without asserting that the members are independent draws.

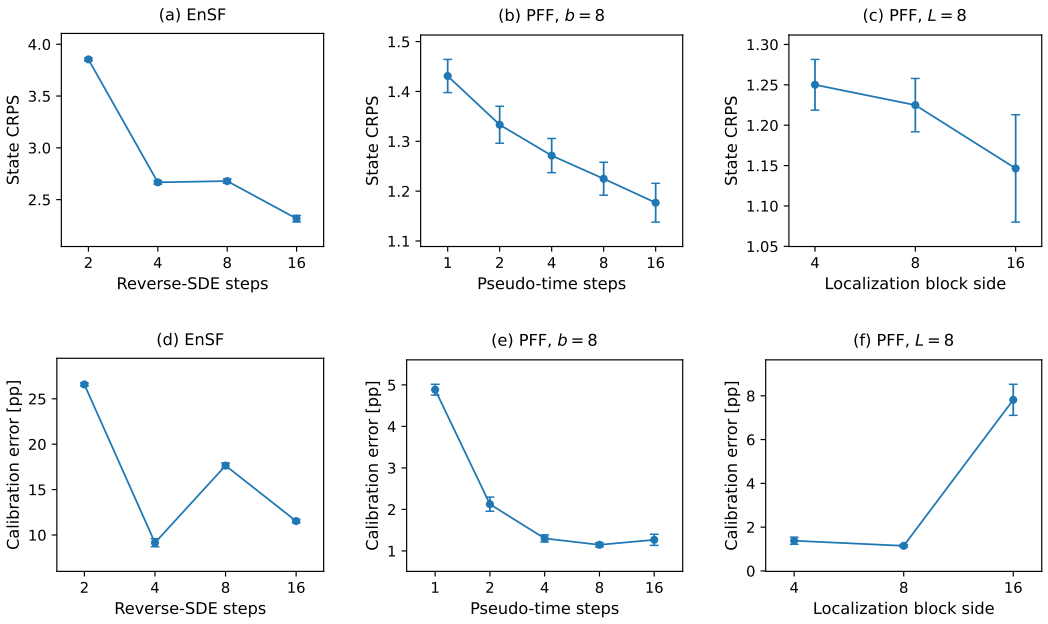


Figure 7: EnSF and PFF parameter sensitivity for Kolmogorov flow with 32 particles. Columns vary the EnSF reverse-SDE steps, PFF pseudo-time steps at $b = 8$, and PFF block side at $L = 8$, respectively. The upper row reports state CRPS and the lower row reports calibration error. Points and error bars show means and one sample standard deviation across five file-level aggregates. The main comparison uses $S = 16$ for EnSF and $b = 8, L = 16$ for PFF.

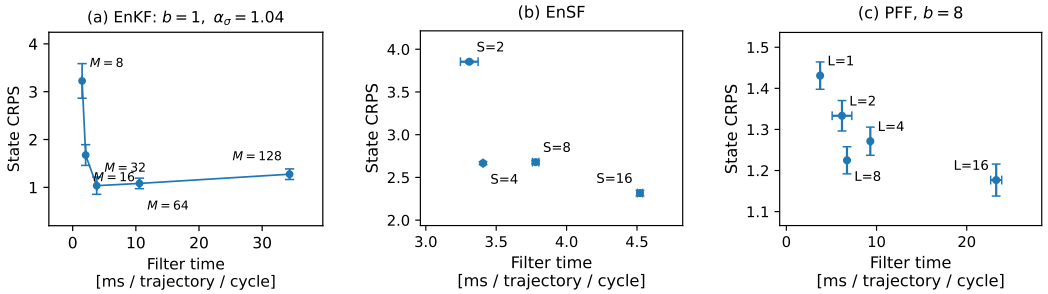


Figure 8: Computational cost of the classical filters in Kolmogorov flow. Left: EnKF filter-only runtime versus ensemble size at $b = 1, \alpha_\sigma = 1.04$. Middle: state CRPS versus runtime for the EnSF reverse-SDE step sweep. Right: state CRPS versus runtime for the PFF pseudo-time step sweep at $b = 8$. EnSF and PFF use 32 particles. Runtime is measured in milliseconds per trajectory and assimilation cycle, excluding diagnostic scoring. Points and error bars denote means and one sample standard deviation across five file-level aggregates. Separately executed configurations need not have monotonically increasing measured runtimes.

Calibration uses central marginal intervals at nominal levels $0.05, 0.10, \ldots, 0.95$. Coverage is pooled over trajectories, scored times, and coordinates within each file before computing

$$\text{Calib. error} = \frac{100}{0.95 - 0.05} \int_{0.05}^{0.95} |\widehat{C}(\alpha) - \alpha| \, d\alpha \tag{121}$$

by trapezoidal integration. The result is reported in percentage points. NFF and the Gaussian ablation evaluate $-\log q_\phi^F(X_t \mid W_t)/d$ directly in physical coordinates. AmEnF and the classical filters return empirical ensembles rather than normalized physical-space densities; their filter NLL/dim is reported as N/A, without fitting a replacement density estimator.

Supplementary-observation update. At $t = 512$, four previously unmeasured cells receive bias-free, signed linear observations with independent Gaussian noise of standard deviation 1. The fixed zero-based $(\text{row}, \text{column})$ coordinates are

$$\mathcal{A} = \{(27, 88), (77, 58), (64, 62), (57, 119)\}. \tag{122}$$

These are the first four entries of the fixed ordered square-x4 coordinate list, selected before row-major sorting. They lie outside the original observation network. All methods use the same coordinate-keyed noise realizations with base seed 20260826 and replicate 0. These measurements are withheld from the filtering encoders and base analysis recursions and do not update model parameters.

For final filtering samples $x^{(i)}$ and supplementary measurements z_j, the diagnostic uses only likelihood reweighting,

$$w_i \propto \exp\left[-\frac{1}{2}\sum_{j\in\mathcal{A}}(z_j - x_j^{(i)})^2\right], \qquad \sum_i w_i = 1. \tag{123}$$

The support consists of 4096 independent NFF samples, all 128 genuine AmEnF members, or all 32 genuine classical members. No members are duplicated to equalize these budgets. Weights are normalized stably without clipping, tempering, or resampling. Effective sample size is recorded even when the weights collapse.

The post-update score at coordinate j is

$$\mathrm{CRPS}_w(j) = \sum_i w_i |x_j^{(i)} - X_{512,j}| - \frac{1}{2}\sum_{i,\ell} w_i w_\ell |x_j^{(i)} - x_j^{(\ell)}|. \tag{124}$$

This is weighted empirical CRPS, not the fair finite-ensemble formula used for state CRPS. The reported value averages the 16,380 coordinates outside $\mathcal{A}$. Here, "unobserved" means not directly measured by the supplementary sensors, and includes cells observed by the original network. The pre-update score uses equal weights on the same final-time support and coordinate set; gain maps report pre-update minus post-update CRPS. This final-time score is distinct from state CRPS averaged over the last 256 filtering times.

Parameter errors and aggregation. Physical-parameter errors are $|\widehat{F}_0 - 1|$ and $|\widehat{\log \mathrm{Re}} - \log 1000|$. Sensor-bias error is the mean absolute error over the 4864 original sensor locations,

$$\mathrm{MAE}_b = \frac{1}{4864}\sum_{j\in\mathcal{D}\cup\mathcal{S}} |\widehat{b}_j - b_j|. \tag{125}$$

State and supplementary-update scores are first averaged over the 128 trajectories within each paired file; calibration is pooled within each file as described above. Results are then summarized by the mean and standard deviation over the five seed-level values. Parameter errors are computed once per trained checkpoint and summarized across the five runs. The final classical exporter uses the sample standard deviation with denominator $5 - 1$.

D.4 NFF POSTERIOR-SAMPLING RUNTIME

We benchmark posterior-ensemble generation using one trained ELBO+CRPS flow checkpoint per system. Each measurement generates $M = 128$ independent samples from $q_\phi^F(X_t \mid W_t)$ for one trajectory at one filtering cycle, with batch size $B = 1$. Checkpoint loading and warm-up are excluded. Each value is the mean $\pm$ sample standard deviation over five timing repeats, with each repeat averaging 32 target times. The benchmark measures posterior-ensemble generation rather than diagnostic scoring. The Lorenz–96 and Kolmogorov-flow measurements include observation encoding at every cycle.

Table 7: NFF posterior-ensemble generation time. Time is measured in milliseconds per trajectory and filtering cycle. The sampling-chunk column gives the maximum number of posterior samples generated simultaneously.

System	Hardware / arithmetic	τ	Sampling chunk	Time [ms]
Coupled oscillator	RTX 5090 / float32	16	128	3.234 ± 0.003
Lorenz–96	GH200 120GB / bfloat16	16	128	9.442 ± 0.015
Kolmogorov flow	GH200 120GB / bfloat16	64	8	331.2 ± 0.8

Table 8: Linear Lorenz–96 filtering after $80{,}000$ updates for NFF and AmEnF. Values are means $\pm$ standard deviations over successful runs; parentheses give successful/attempted seed counts for this evaluation. Failed runs are excluded from the numerical summaries. E/C denote recursive ELBO/future-observation CRPS; F/G denote flow/diagonal Gaussian. The final column scores coordinates not directly measured by supplementary sensors and reflects both posterior quality and finite-sample reweighting effects.

	Marginal posterior		Joint density and update	
Method	State CRPS ↓	Calib. error [pp] ↓	Filter NLL/dim ↓	Unobserved state CRPS after additional obs. ↓
NFF				
E+C, F $(5/5)$	0.449 ± 0.000	0.828 ± 0.101	1.192 ± 0.001	0.448 ± 0.002
E+C, G $(5/5)$	0.448 ± 0.000	1.099 ± 0.095	1.190 ± 0.001	0.448 ± 0.003
E, F $(3/5)$	0.445 ± 0.000	0.824 ± 0.023	1.181 ± 0.001	0.443 ± 0.002
C, F $(5/5)$	0.449 ± 0.000	0.965 ± 0.239	$(2.054 \pm 1.246) \times 10^7$	0.460 ± 0.004
EnKF $(5/5)$	0.448 ± 0.000	1.570 ± 0.001	N/A	0.452 ± 0.005
PFF $(5/5)$	0.474 ± 0.000	8.232 ± 0.015	N/A	0.480 ± 0.005
EnSF $(5/5)$	1.114 ± 0.001	10.116 ± 0.028	N/A	1.235 ± 0.002
AmEnF $(5/5)$	0.557 ± 0.000	39.668 ± 0.088	N/A	0.557 ± 0.004

E ADDITIONAL RESULTS OF THE EXPERIMENTS

E.1 LORENZ96

E.1.1 LINEAR OBSERVATIONS

Table 8 evaluates $h(x) = x$ as a linear-observation control for the nonlinear Lorenz–96 experiment. The identity operator removes the coordinatewise sign ambiguity of absolute-value observations while retaining the nonlinear stochastic dynamics and noise levels $\sigma_{\mathrm{dyn}} = \sigma_{\mathrm{obs}} = 1$. Among successful runs, EnKF and the NFF variants attain closely clustered marginal state CRPS of 0.445–0.449. PFF and AmEnF obtain 0.474 and 0.557, respectively, while EnSF remains less competitive at 1.114. In particular, the agreement between EnKF and NFF provides an implementation sanity check in a setting without observation-induced sign ambiguity, complementing the nonlinear-observation comparison.

Within ELBO+CRPS training, the flow and diagonal-Gaussian variants have nearly identical state CRPS and filtering NLL, and the same mean post-update CRPS to the reported precision. The flow has lower mean calibration error (0.828 versus 1.099 pp), but the diagonal-Gaussian approximation is otherwise competitive on these metrics. Thus, the substantial advantage of the flow family under ambiguous observations is not reproduced in this linear-observation control.

Under the 80,000-update training budget, the two CRPS-containing flow variants and the ELBO+CRPS diagonal-Gaussian variant succeed for all five seeds. The ELBO-only flow succeeds in three of five runs and attains the lowest reported mean in each column among those retained runs. The linear-observation results therefore do not show a uniform accuracy improvement from adding predictive CRPS. However, the ELBO-only summaries exclude two failed runs and use a different subset of seeds, so their small numerical advantages must be interpreted together with the failure counts.

The CRPS-only flow achieves competitive marginal state CRPS and low marginal calibration error, yet its filtering NLL/dim is $(2.054 \pm 1.246) \times 10^7$, compared with mean values of 1.181–1.192 for the ELBO-containing variants. Thus, good marginal scores can coexist with a very poor joint filtering log score even when the observation operator has no sign ambiguity. These diagnostics alone do not determine whether the density error arises primarily from dependence structure, marginal shape, or tail behavior.

The ELBO-containing variants also achieve lower absolute CRPS after supplementary-observation reweighting than the CRPS-only flow (0.443–0.448 versus 0.460). The final column is not a paired conditioning gain relative to the first column: it scores the final filtering time and excludes the supplementary sensor coordinates, whereas the first column averages over different times and coordinates. Post-update scores also reflect finite-sample reweighting effects. Overall, this control shows that the benefits of posterior flexibility and auxiliary CRPS depend on the observation setting, while marginal accuracy alone remains insufficient to assess the learned filtering density.

E.1.2 FIGURES FOR THE ADDITIONAL OBSERVATION

Figure 9 compares the ELBO+CRPS flow (upper) with the CRPS-only flow (lower). Both models show positive CRPS gains at the ten supplementary sensor coordinates. Away from these sensors, the ELBO+CRPS flow yields mostly positive or near-zero gains, with a few negative values, whereas the CRPS-only flow deteriorates at many coordinates.

This contrast complements the aggregate results in Table 1. Despite similar time-averaged marginal state CRPS, the ELBO+CRPS flow achieves lower post-update CRPS on the 30 coordinates not directly measured by the supplementary sensors: 0.814 ± 0.004, compared with 0.876 ± 0.007 for CRPS-only. Because the supplementary likelihood depends only on sensor coordinates, its influence on the remaining coordinates probes the cross-coordinate structure of the filtering distribution. These results support retaining the recursive ELBO alongside predictive CRPS to learn distributions useful for further conditioning, rather than assessing them through marginal scores alone. The diagnostic also depends on finite-sample reweighting and does not by itself establish exact joint-posterior recovery.

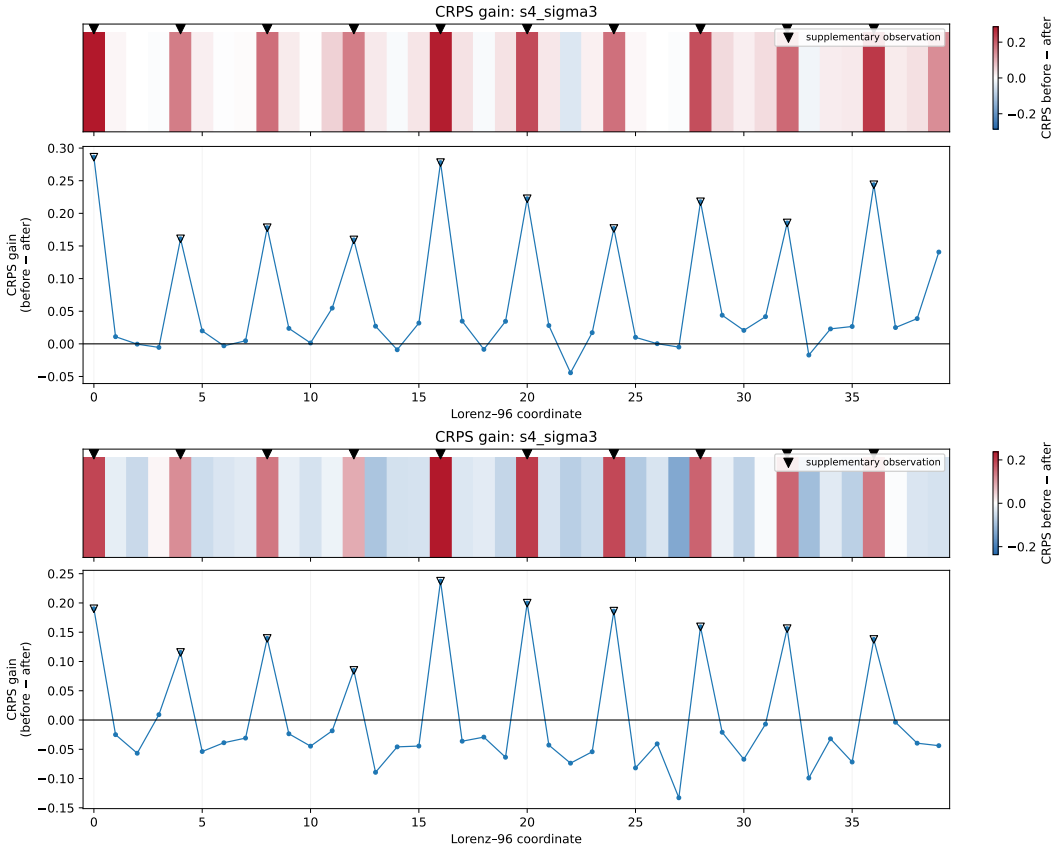


Figure 9: CRPS improvement after additional observations for the $h(x) = |x|$ experiments in the main text.

E.2 KOLMOGOROV FLOW

E.2.1 PARAMETER IDENTIFIABILITY TEST

We examine how spatial observation density affects parameter recovery in Kolmogorov flow. We use uniform sensor grids covering the full domain, with linear measurements $h(x) = x$ at the observed locations. This removes the pointwise sign ambiguity of squared observations while retaining the difficulty of spatially incomplete measurements. The physical parameters F_0 and $\log \mathrm{Re}$ are learned jointly with the NFF inference network from observations alone.

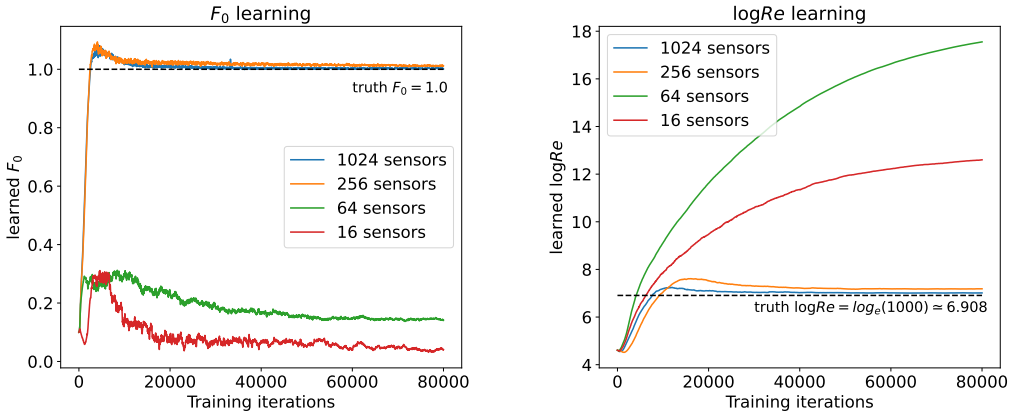


Figure 10: Learning of F_0 (left) and $\log \mathrm{Re}$ (right) under varying observation density. Linear observations are collected on uniform 32×32, 16×16, 8×8, and 4×4 sensor grids over the 128×128 state grid, corresponding to 1,024, 256, 64, and 16 sensors. Each curve shows a single training run. Dashed lines indicate the true values, $F_0 = 1$ and $\log \mathrm{Re} = \log 1000$.

Figure 10 shows a clear separation between the two denser and two sparser observation networks. With 1,024 or 256 sensors, F_0 approaches its true value and $\log \mathrm{Re}$ stabilizes near, but slightly above, the truth. With 64 or 16 sensors, F_0 is substantially underestimated, while $\log \mathrm{Re}$ continues to increase far beyond its true value over the plotted training interval.

These results are consistent with limited practical identifiability under sparse observations, although the learning curves alone cannot distinguish insufficient observational information from optimization or variational-approximation effects. We therefore interpret this experiment as evidence that observation-only parameter recovery is sensitive to spatial observation density even with linear measurements, rather than as a proof of structural non-identifiability.

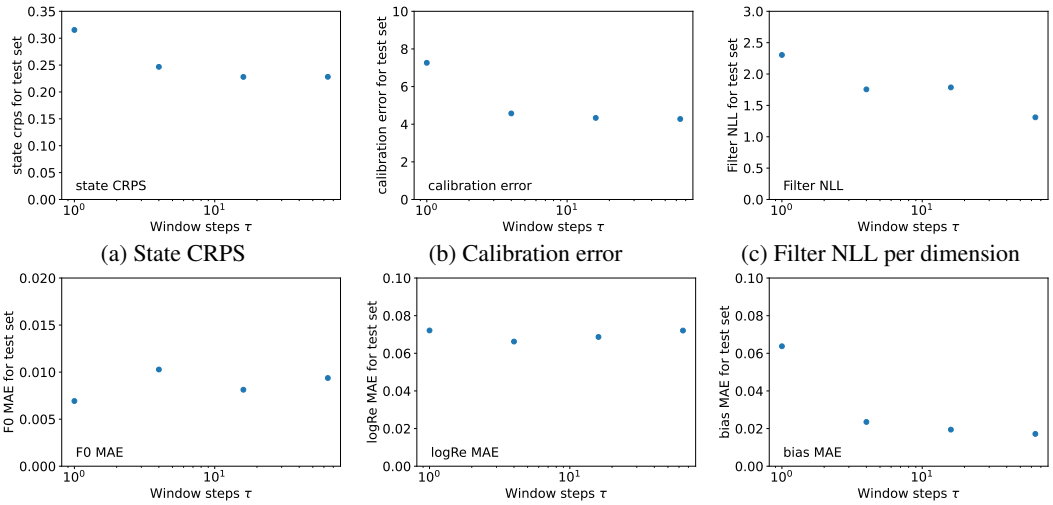
E.2.2 SENSITIVITY TO CONDITIONING-WINDOW LENGTH AND TRAINING-SET SIZE

We evaluate the ELBO+CRPS flow under the mixed observation network of Section 5.3, using training seed 1 and a fixed budget of 80,000 gradient updates per run. We vary either the conditioning-window length or the number of distinct training observation sequences, retaining the remaining settings and held-out evaluation protocol. All results use the final checkpoint.

Conditioning-window length. We vary $\tau \in \{1, 4, 16, 64\}$ using the reference training run's 80,000 distinct observation sequences (Figure 11). Increasing τ from 1 to 64 reduces state CRPS from approximately 0.32 to 0.23 and improves calibration, filter NLL, and sensor-bias estimation. State CRPS and calibration change little between $\tau = 16$ and 64, whereas filter NLL improves further, supporting the longer window for density estimation. The errors in F_0 and $\log \mathrm{Re}$ show no clear monotonic trend across window lengths.

NFF nevertheless provides useful state inference with $\tau = 1$, when $q_\phi^F(X_t \mid Y_t)$ conditions only on the current spatial observation field. This does not remove temporal information from training: for $t \geq 2$, the recursive ELBO uses $q_\phi^B(X_{t-1} \mid X_t, Y_{t-1})$ and couples adjacent states through the transition density, while future-observation CRPS scores predictions obtained by propagating

filtering samples. Both terms use the physical dynamics model with jointly learned parameters. The useful single-snapshot performance is therefore consistent with the amortized filter exploiting spatial and dynamical regularities learned across training sequences; longer conditioning windows additionally provide the particular test trajectory's observation history.



(a) State CRPS (b) Calibration error (c) Filter NLL per dimension

(d) Absolute error in F_0 (e) Absolute error in $\log \mathrm{Re}$ (f) Sensor-bias MAE

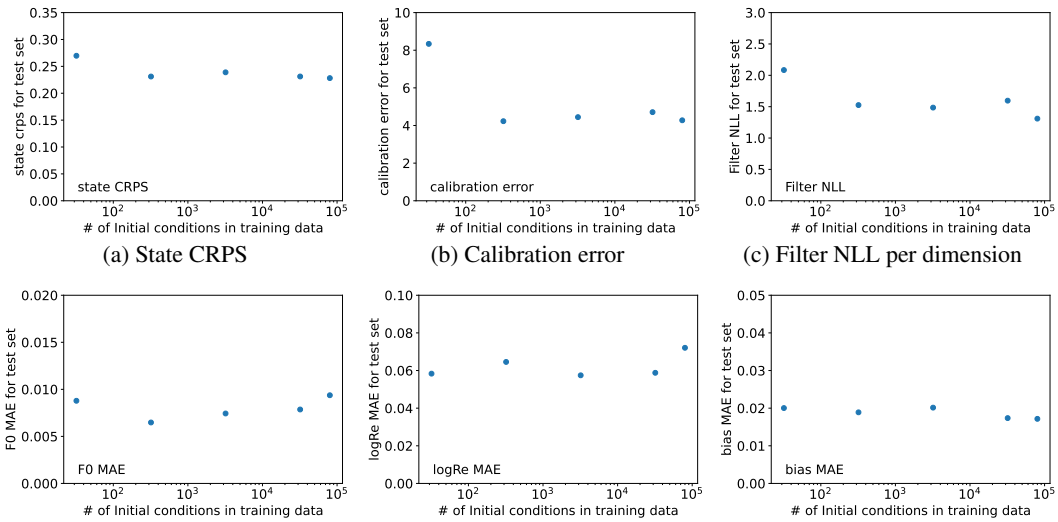
Figure 11: **Sensitivity to conditioning-window length in Kolmogorov flow.** The ELBO+CRPS flow is trained separately with $\tau \in \{1, 4, 16, 64\}$, using $80{,}000$ distinct training observation sequences and $80{,}000$ gradient updates per run. Panels report (a) state CRPS, (b) calibration error in percentage points, (c) filter NLL per state dimension, (d,e) absolute errors in the learned physical parameters, and (f) sensor-bias MAE over observed locations. Lower values are better in all panels. Each setting uses training seed 1; the points are not averages over independent training runs.

Training-set size. At fixed $\tau = 64$, we vary the number of distinct training observation sequences over $N \in \{32, 320, 3{,}200, 32{,}000, 80{,}000\}$ (Figure 12). Each sequence corresponds to a distinct initial condition, but only observations are used for training. The reference count of $80{,}000$ reflects the sequences actually consumed during training, rather than the nominal $320{,}000$-trajectory pool. Smaller datasets are reused to retain the same update budget. This experiment therefore measures sensitivity to training-data availability rather than a reduction in optimization cost.

Notably, NFF retains useful held-out state inference when trained on observations from only 32 trajectories. Despite a $2{,}500$-fold reduction in distinct training trajectories relative to the reference run, state CRPS increases only from approximately 0.23 to 0.27, while physical-parameter and sensor-bias errors remain comparable to those obtained with larger datasets. State CRPS and calibration are already close to the reference values with 320 trajectories. Filter NLL remains more sensitive to dataset size, and both calibration error and filter NLL deteriorate more noticeably with only 32 trajectories. Thus, additional data improve uncertainty quality even when marginal state accuracy is largely retained. These single-seed results suggest that useful observation-only inference need not require a large training archive in this setting, supporting NFF's potential for data-limited applications.

E.2.3 Figures for the additional observation

Supplementary-observation protocol. We apply the same final-time reweighting diagnostic to the 16,384-dimensional vorticity field. Four supplementary sensors measure signed vorticity at the fixed locations marked in the figures, using $h_{\mathrm{add}}(x) = x$ and independent Gaussian observation noise with standard deviation $\sigma_{\mathrm{add}} = 1.0$. These measurements are not supplied to the filtering encoder. Instead, each method's final filtering samples are reweighted using equation 123. The unobserved-state metric in Table 2 excludes these four locations and averages the post-update CRPS over the remaining 16,380 state coordinates.



(a) State CRPS (b) Calibration error (c) Filter NLL per dimension

(d) Absolute error in F_0 (e) Absolute error in $\log \mathrm{Re}$ (f) Sensor-bias MAE

Figure 12: **Sensitivity to training-set size in Kolmogorov flow.** At fixed $\tau = 64$, the ELBO+CRPS flow is trained with observations from $N \in \{32, 320, 3{,}200, 32{,}000, 80{,}000\}$ distinct trajectories. All runs use training seed 1 and 80,000 gradient updates, reusing smaller datasets. The horizontal axis counts distinct training initial conditions actually used, not gradient updates or individual observation times; the reference run consumes 80,000 of the 320,000 available trajectories. Panel definitions and units match Figure 11; lower is better throughout. Each point represents one training run.

Spatial CRPS maps. Figures 13 and 14 show the spatial distribution of state CRPS before and after assimilating the supplementary measurements. For each method, panels (a) and (b) show the before- and after-update scores, while panel (c) shows their difference, $\Delta\mathrm{CRPS} = \mathrm{CRPS}^{\mathrm{before}} - \mathrm{CRPS}^{\mathrm{after}}$. Red and blue in the difference maps indicate improvement and deterioration, respectively. The numbered square markers identify the supplementary sensor locations. The common color scales allow direct comparison across methods, while the separate difference scale makes small update effects visible.

Interpretation. The combined flow retains low CRPS over most of the field after the update, with modest, spatially structured gains and losses. In contrast, the CRPS-only flow exhibits broadly negative gains in the displayed example. The ELBO-only flow and AmEnF show spatially mixed changes on top of substantially larger initial errors, while the classical ensemble baselines also exhibit substantial regions of deterioration under the same reweighting diagnostic.

The nearly unchanged scores of the diagonal-Gaussian ablation have a different interpretation. Conditional on the original observation window, this posterior factorizes across state coordinates. Consequently, exact conditioning on a likelihood involving only the four sensor coordinates leaves every other marginal unchanged. Small nonzero changes away from the sensors can nevertheless arise from finite-sample reweighting. A small difference map therefore does not by itself demonstrate successful information transfer; it must be considered together with the absolute before- and after-update errors.

The aggregate results in Table 2 support this distinction. The combined flow achieves an unobserved-coordinate post-update CRPS of 0.232 ± 0.007, compared with 2.213 ± 0.262 for the CRPS-only flow and 2.022 ± 0.056 for PFF, the best-performing baseline on this metric. Together, the maps and aggregate scores show that the combined flow remains substantially more accurate after supplementary assimilation. This is an operational test of conditioning the represented filtering distributions, rather than a proof of exact joint-posterior recovery: finite-sample reweighting can affect the observed gains, and individual noisy measurements need not improve every coordinate.

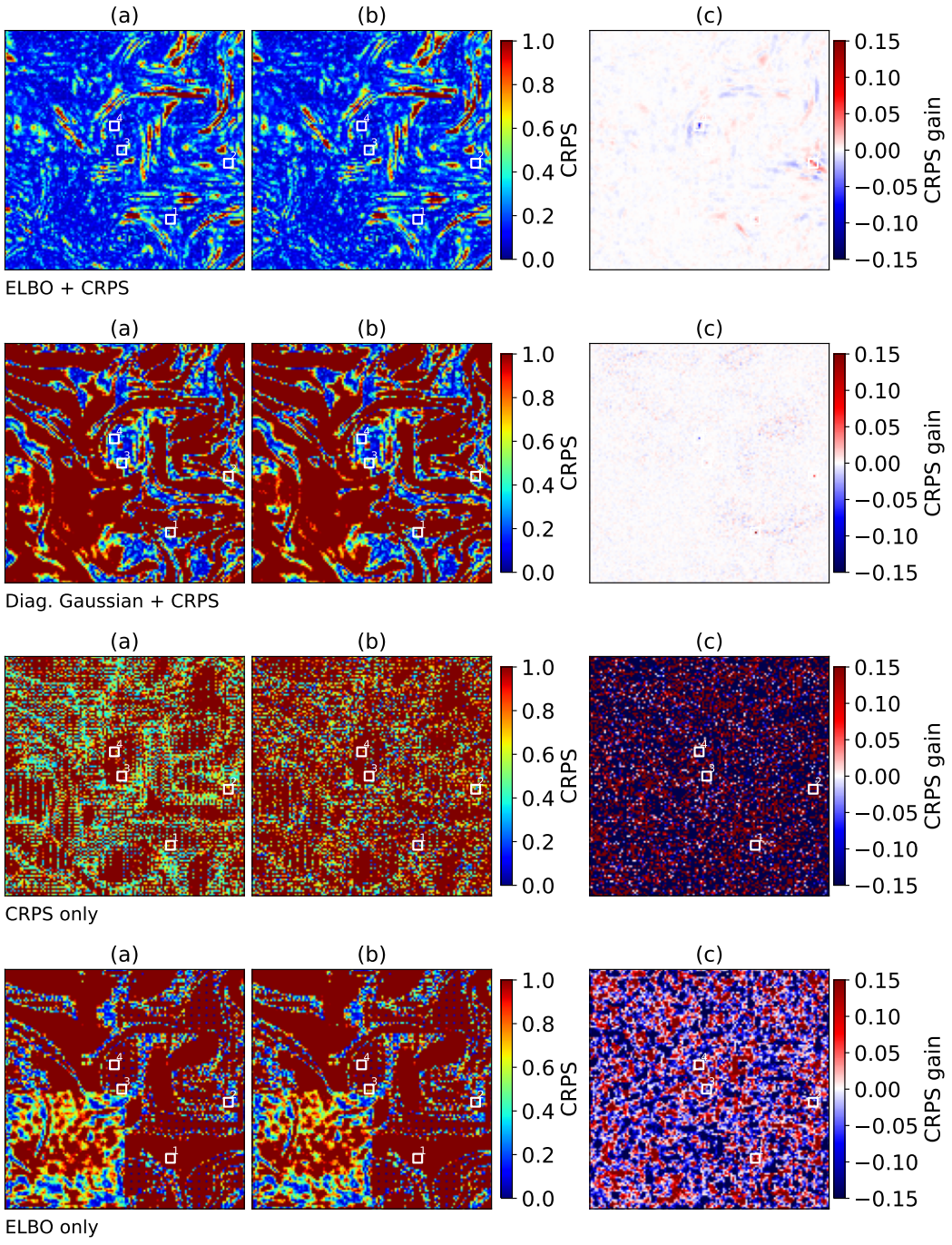


Figure 13: Kolmogorov supplementary-observation update. Columns show state CRPS (a) before and (b) after the update, and (c) the gain (before minus after). Numbered squares mark the four supplementary sensors.

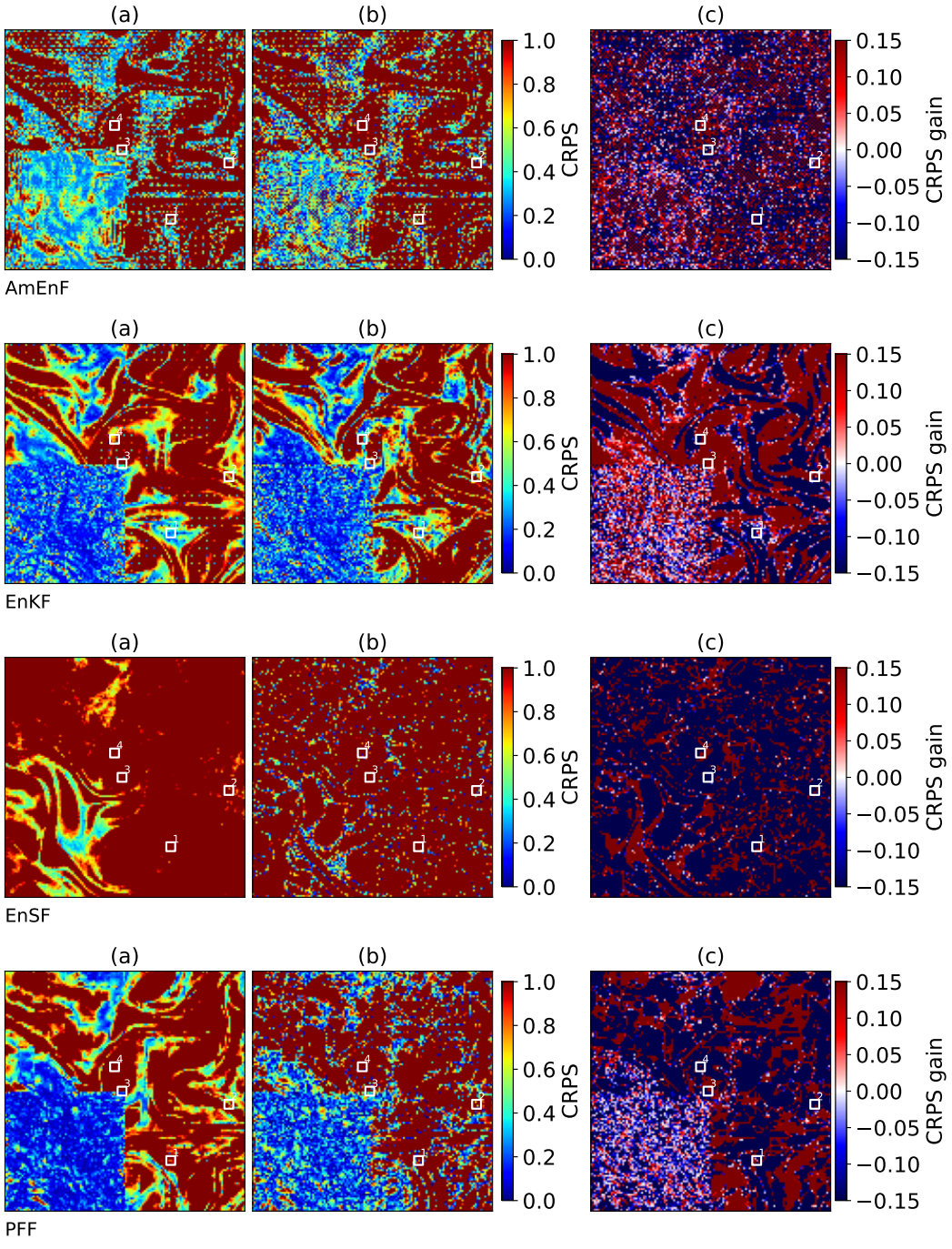


Figure 14: Continuation from Fig. 13.

E.3 NUMERICAL CHECKS OF FILTERING NLL

We screened 64 held-out state–time pairs per checkpoint for five CRPS-only checkpoints in each system and five ELBO+CRPS checkpoints in Lorenz–96. Detailed checks used eight prespecified pairs and the four largest screened NLLs per checkpoint, with duplicates removed, yielding 58 distinct pairs per CRPS-only cohort and 57 for the ELBO+CRPS control. The two Lorenz–96 cohorts used identical observations and the same 40 prespecified pairs. We compared the original density evaluator with an independently assembled change-of-variables calculation, repeated density evaluation with fixed and recomputed encoder contexts in FP64, and checked forward/inverse consistency at held-out truths and two generated samples per detailed context.

Table 9: Numerical NLL checks. Entries are maxima over the checked subsets pooled across five seeds, not full-test error bounds. Relative NLL discrepancies use the FP64 reference; reconstruction RMS is divided by $\max\{1, \mathrm{RMS}(x)\}$. Sampling-density discrepancies are absolute differences between generation-time and re-evaluated log densities per state dimension. The configured path uses BF16 encoder autocast and TF32; strict audit paths disable both.

	Lorenz–96		Kolmogorov
Diagnostic	ELBO+CRPS	CRPS-only	CRPS-only
Fixed-context FP32/FP64 NLL difference	9.98×10^{-7}	8.98×10^{-6}	2.57×10^{-6}
Configured/end-to-end FP64 NLL discrepancy	0.925%	43.9%	0.983%
Truth round-trip scaled RMS, FP32	1.80×10^{-6}	0.321	9.59×10^{-5}
Truth round-trip scaled RMS, FP64	3.27×10^{-15}	7.09×10^{-10}	1.24×10^{-13}
Sampling-density discrepancy, public FP32	1.72×10^{-6}	358	1.40×10^{-4}
Sampling-density discrepancy, audit FP64	2.66×10^{-15}	1.41×10^{-9}	1.73×10^{-13}

The extreme CRPS-only NLLs persisted under end-to-end FP64 evaluation: 1.06×10^{10}–4.97×10^{13} nats per dimension in Lorenz–96 and 2.35×10^{5}–2.63×10^{6} in Kolmogorov flow. Their Gaussian-base quadratic terms dominated the remaining NLL terms, supporting extremely low assigned density at the evaluated truths rather than an explanation based solely on low-precision arithmetic. However, the Lorenz–96 CRPS-only checkpoints exhibited substantial single-precision inversion and sampling-density inconsistency (Table 9); their exact NLL values are precision-sensitive. The ELBO+CRPS Lorenz–96 control instead had end-to-end FP64 NLLs between 1.17 and 3.50 nats per dimension and showed no comparable inversion inconsistency. These subset checks support the magnitude of the density contrast while identifying a numerical limitation of the CRPS-only ablation. They neither certify full-test numerical accuracy nor attribute filtering error exclusively to cross-coordinate dependence.