Diffusion Model's Generalization Can Be Characterized by Inductive Biases toward a Data-Dependent Ridge Manifold

Ye He
University of Colorado, Boulder
ye.he@colorado.edu

Yitong Qiu
Columbia University
yq2463@columbia.edu

Molei Tao
Georgia Institute of Technology
mtao@gatech.edu

Abstract

We study a data-dependent notion of diffusion-model generalization: when a model does not memorize the training set, where do its generated samples go relative to the geometry induced by the data? To answer this, we introduce a time-dependent family of log-density ridge manifolds constructed from the smoothed empirical distribution, and use it to characterize reverse-time inference. Our main results describe a reach–align–slide mechanism: samples enter a neighborhood of the ridge, after which normal alignment and tangential sliding are controlled by the corresponding components of training error. These post-entry effects may occur simultaneously within the regular ridge regime of the analysis. We further connect this geometric picture to training dynamics through directional decompositions of the learned error, and make this link explicit for random feature models, where architectural bias and optimization contributions can be separated quantitatively. Experiments on synthetic multimodal data and MNIST latent diffusion support the predicted geometric behavior in both low and high dimensions.

1 Introduction

Diffusion models [20, 31, 33] now achieve state-of-the-art sample quality across a wide range of generative tasks, making them a central tool for image, audio, and video generation [6, 14, 23]. At the same time, it is increasingly important to understand how innovative these models actually are, that is, what they generate beyond the training data. A central concern is memorization: in some regimes, diffusion models can behave like stochastic parrots that simply reproduce training data, raising both scientific and practical concerns, including privacy and safety risks [7, 15, 27, 32]. Recent theory and empirical evidence suggest that non-memorizing behavior arises from various sources of error inside the learned diffusion model [e.g., 37], and in the literature the term generalization is often used in precisely this sense of non-memorization [e.g., 21, 38].

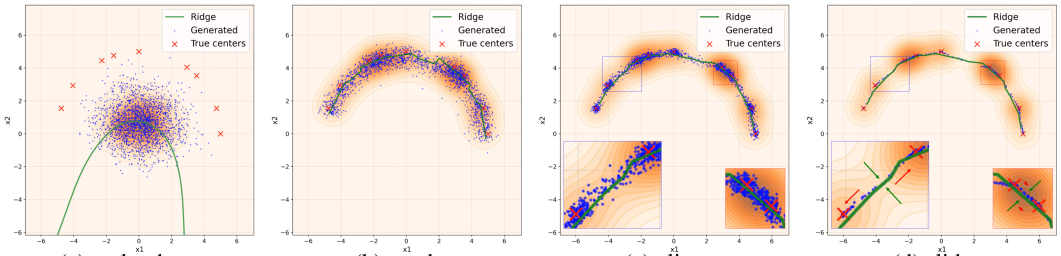
This leads to the central question of the paper:

When a diffusion model does not memorize the training set, where do its generated samples go?

Equivalently, once non-memorization has occurred, what geometric structure organizes the new samples produced by reverse-time inference? Our goal is to answer this question explicitly and quantitatively. In particular, we seek not merely to say that generated samples differ from the training data, but to describe where they are located relative to the geometry induced by the data.

This question should be understood in a fully data-dependent sense. Rather than taking an unknown population distribution as the primary reference, we take the finite training dataset itself as the object that defines the relevant geometry, and ask how generation departs from that geometry. Our

40th Conference on Neural Information Processing Systems (NeurIPS 2026).



(a) early phase (b) reach (c) align (d) slide

Figure 1: **Reach–align–slide on a semi-circular dataset.** Nine unevenly spaced training points (red crosses) lie on a semi-circle of radius 4. Generated samples (blue) evolve relative to the log-density ridge (green). In the zoomed regions, red arrows indicate sliding and green arrows indicate continued alignment, with lengths indicating intensity. The panels illustrate relative dominance, not a theorem asserting three disjoint stages.

focus is different from classical population-level generalization: we are not primarily asking how close the generated distribution is to an unknown population law, but how the model generates new samples relative to the observed data. This viewpoint is especially natural for structured intermediate generations between training samples, since the issue is not only distributional discrepancy, but also spatial organization. A more detailed comparison is given in Appendix A. Non-memorization and ridge proximity alone do not certify population recovery or semantic sample quality.

Our answer is geometric. We construct a time-indexed family of log-density ridge manifolds from the smoothed empirical distribution and use it as the reference geometry for reverse-time inference. Relative to these ridges, generated samples follow a reach–align–slide mechanism: they first enter a ridge neighborhood and thereafter align toward it in normal directions and slide along it in tangent directions. Alignment and sliding describe two components of the same post-entry trajectory, not disjoint temporal intervals. The reach and align results explain where generation concentrates, while the slide result gives a local directional description relative to nearby data-induced centers; it does not fully characterize the generated distribution within the tangent space. Crucially, normal training error controls alignment and tangential training error controls sliding. Figure 1 illustrates these effects on a semi-circular dataset.

A tractable setting where the training-to-geometry link becomes explicit is random feature neural networks (RFNNs) trained by gradient descent. There we decompose training error into directional components and show how finite-width approximation error and incomplete optimization translate into different alignment and sliding behaviors. This analysis is not intended as a full model of modern architectures, but as a nonasymptotic example showing how architecture and optimization jointly shape diffusion generation through direction-dependent geometric effects.

Our perspective is most closely related to recent theoretical work that seeks to explain why diffusion models generate non-memorizing samples. One line studies whether generalization can already arise from the stochasticity or structure of the finite training target itself [4, 34]; in contrast, we take the empirical training set as given and ask how the learned model generates relative to its induced geometry. Another direction analyzes training-induced bias, focusing on how model class, feature learning, or optimization shape learned scores [5, 22, 30, 36]. Our contribution is complementary: we quantify how that bias appears during inference through distinct normal and tangential effects relative to a data-dependent ridge family. Finally, several works study inference-time behavior under structured settings or geometry-adaptive smoothing [2, 16, 26]. Our work is closest in spirit to this direction, but makes the geometric object explicit from empirical data, describes both concentration and part of the tangential organization of generation, and connects these effects to directional training error.

Taken together, our framework links data geometry, inference dynamics, and training error: (1) construct a time-dependent log-density ridge geometry from the smoothed empirical distribution (Section 3.1); (2) establish tube entry and analyze post-entry normal alignment and tangential sliding (Sections 3.2–3.4); and (3) connect these quantities to directional training losses, with a nonasymptotic RFNN+GD instantiation separating architecture and optimization effects (Section 4). Section 5 and Appendices J–K provide experiments.

2 Preliminaries

SDE-based diffusion models. To generate samples in $\mathbb{R}^d$ from data $x_0^{(1)}, \dots, x_0^{(n)} \in \mathbb{R}^d$, we consider the variance-preserving (VP) forward process $dX_t = -X_t\,dt + \sqrt{2}\,dB_t$, $t \in [0, T]$, with marginals $p_t = \mathrm{Law}(X_t)$ and $p_0 = p$. The corresponding reverse process is

$$dY_t = \big(Y_t + 2\nabla \log p_{T-t}(Y_t)\big)dt + \sqrt{2}\,d\bar{B}_t. \tag{1}$$

Write $a_t = e^{-t}$ and $h_t = 1 - e^{-2t}$, so $X_t = a_t X_0 + \sqrt{h_t} z$ in distribution, with $z \sim \mathcal{N}(0, I_d)$ independent of X_0. We introduce an early stopping time $0 < \delta \ll 1$, so the score is used only on $[\delta, T]$.

Denoising mean matching loss. We consider learning the posterior mean instead of the score. By Tweedie's formula,

$$\nabla \log p_t(x) = -\frac{x}{h_t} + \frac{1}{h_t}\mathbb{E}_{x_0 \sim p_{0|t}(\cdot|x)}[a_t x_0], \tag{2}$$

where $p_{0|t}(\cdot|x)$ denotes the law of X_0 given $X_t = x$. Define

$$m(t, x) := \mathbb{E}[a_t X_0 \mid X_t = x]. \tag{3}$$

Thus learning the score is equivalent to learning this posterior mean. We measure approximation error by

$$\mathcal{L}_{\mathrm{MM}}(A) = \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}\big[\|m_A(t, X_t) - m(t, X_t)\|^2\big]dt, \tag{4}$$

where m_A is the learned posterior mean. In practice, we train using its denoising version

$$\mathcal{L}_{\mathrm{DMM}}(A) = \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}\big[\|m_A(t, X_t) - a_t X_0\|^2\big]dt. \tag{5}$$

These losses differ by an A-independent conditional-variance term (Appendix B). The simulated reverse process, initialized at $\mathcal{N}(0, I_d)$, is

$$d\widetilde{Y}_t = \left(\widetilde{Y}_t + \frac{2(m_A(T-t, \widetilde{Y}_t) - \widetilde{Y}_t)}{h_{T-t}}\right)dt + \sqrt{2}\,d\widetilde{B}_t. \tag{6}$$

Throughout, $e_A = m_A - m$; the learned reverse drift therefore differs from the exact drift by $+2e_A/h_t$ at diffusion time t.

Random feature neural network and gradient descent. The inference analysis in Section 3 is architecture-agnostic. To make the training-to-geometry link explicit in Section 4.2, we use the RFNN parametrization

$$m_A(t, x) = \frac{A}{\sqrt{p}}\sigma\left(\frac{W_x x}{\sqrt{d}} + \frac{W_t \varphi_t}{\sqrt{2K_t}} + b\right) =: \frac{A}{\sqrt{p}}\sigma_t(x).$$

Here W_x, W_t are fixed Gaussian random matrices, φ_t is a Fourier time embedding, $b \sim \mathcal{N}(0, I_p)$, and only $A \in \mathbb{R}^{d \times p}$ is trained. The loss is quadratic in A, and full-batch gradient descent with step size η satisfies

$$A_{k+1} - A_k = -2\eta A_k \widetilde{U} + 2\eta \widetilde{V}, \tag{7}$$

where

$$\widetilde{U} = \int_\delta^T \frac{w(t)}{h_t^2}\frac{\mathbb{E}[\sigma_t(X_t)\sigma_t(X_t)^\top]}{p}\,dt, \qquad \widetilde{V} = \int_\delta^T \frac{w(t)}{h_t^2}\frac{\mathbb{E}[a_t X_0 \sigma_t(X_t)^\top]}{\sqrt{p}}\,dt.$$

Further RFNN details are given in Appendix I.

3 Geometric Properties of the Inference Process

We introduce the data-dependent ridge geometry and study reverse-time inference relative to it. The logical ordering is tube entry followed by two potentially simultaneous directional effects: normal alignment and tangential sliding. We work under the following empirical-data setting.

Assumption 3.1. *The data points $\{x_0^{(i)}\}_{i=1}^n$ are distinct and bounded: $\Delta := \min_{i \neq j}\|x_0^{(i)} - x_0^{(j)}\| > 0$ and $R := \max_i \|x_0^{(i)}\| < \infty$. The data distribution is $p = \frac{1}{n}\sum_{i=1}^n \delta_{x_0^{(i)}}$.*

3.1 Data-dependent Manifolds—Log-density Ridge Sets

For the smoothed empirical distribution p_t, the log-density ridge reflects structure induced by the training data and serves as the reference object for describing generation. Intuitively, ridges generalize local maxima by imposing stationarity and negative curvature in selected directions.

Definition 3.2 (Log-density ridge sets)**.** *For a smooth positive density q on $\mathbb{R}^d$ and an integer $0 \le d^* < d$, the d^*-dimensional log-density ridge set with threshold $\beta > 0$ is*

$$\mathcal{R}_{d^*}(q;\beta) = \{x \in \mathbb{R}^d : E(x)E(x)^\top \nabla \log q(x) = 0,\ \lambda_{d^*+1}(x) \le -\beta\},$$

where $E = (v_{d^*+1}, \dots, v_d)$ consists of the bottom eigenvectors of $\nabla^2 \log q$, ordered by $\lambda_1 \ge \cdots \ge \lambda_d$. For $d^* > 0$, the strict gap $\lambda_{d^*} > \lambda_{d^*+1}$ is required wherever the ridge is considered. For $d^* = 0$, take $E = I_d$.

At a nondegenerate local maximum, stationarity and negative curvature hold in every direction. On a positive-dimensional ridge, the defining conditions constrain only the bottom Hessian eigenspace and allow variation in the remaining directions. Thus a zero-dimensional ridge consists of modes; a one-dimensional ridge traces curves of directional local maxima. For the empirical Gaussian mixture, small-noise modes lie near transported training points $a_t x_0^{(i)}$ (Lemma E.2), but positive-dimensional ridges need not pass through every such point. Unlike classical density ridges [10, 17], the definition uses log-density, which is natural because inference is driven by $\nabla \log p_t$.

For each $t \in [\delta, T]$, write $\mathcal{R}_t = \mathcal{R}_{d^*}(p_t; \beta_t)$. In the theoretical estimates below, use the same threshold $\beta_t = c/h_t$, with fixed $c \in [1/2, 1)$; see Remark D.6. As a simple analytical example already used in Section 4.2, for two points $(\pm\mu, 0)$,

$$m(t,x) = \left(a_t \mu \tanh \frac{a_t \mu x_1}{h_t}, 0\right), \qquad \mathcal{R}_t = \{x_2 = 0\}.$$

The line is a ridge because the score is stationary in the vertical direction and the vertical curvature is $-1/h_t$. The modes on that line need not coincide exactly with the transported data points.

Tube neighborhood and projection map. We analyze inference in a tube where the nearest-point projection is well-defined.

Assumption 3.3 (Smoothness and positive reach)**.** *For $t \in [\delta, T]$, $\mathcal{R}_t$ is a (piecewise) C^2-embedded submanifold in $\mathbb{R}^d$ with reach at least $r_t > 0$: points within distance r_t have a unique nearest point on $\mathcal{R}_t$.*

Proposition 3.4. *Under Assumption 3.1, the ridge family satisfies Assumption 3.3. As $t \to \delta^+ \ll 1$, $r_t = \Omega(h_t^2 \theta_t^{-1} R^{-3})$ for $\theta_t = \exp(-o(h_t^{-1}))$. For $0 < \rho_t < r_t$, define*

$$\mathcal{T}_t(\rho_t) := \{x : \operatorname{dist}(x, \mathcal{R}_t) \le \rho_t\}. \tag{8}$$

Then $\Pi_t : \mathcal{T}_t(\rho_t) \to \mathcal{R}_t$ is well-defined; $x - \Pi_t(x) \in N_{\Pi_t(x)}\mathcal{R}_t$; Π_t is C^1 with $\sup_x \|\nabla \Pi_t(x)\| \le (1 - \rho_t/r_t)^{-1}$; and for $\rho_t = \Theta(r_t)$, $\sup_x \|\partial_t \Pi_t(x)\| = O(R)$.

For a regular ridge point z, let $P_t^\perp(z)$ and $P_t^\parallel(z) = I - P_t^\perp(z)$ denote the orthogonal projectors onto its geometric normal and tangent spaces. In the tube, $P_t^\dagger(x)$ abbreviates $P_t^\dagger(\Pi_t(x))$. These geometric projectors and the Hessian eigenspaces in Definition 3.2 have distinct definitions; their use in the local estimates is discussed in Appendices C–G. The post-entry analysis is understood on regular ridge pieces where the required projection and local estimates remain valid, not across a loss of regularity or an eigengap degeneracy.

3.2 Reaching the Tube Neighborhood

The local displacement analysis begins after entry into the projection tube. Define

$$\widetilde{t}_{\text{in}} := \inf\{0 \le t \le T - \delta : \widetilde{Y}_t \in \mathcal{T}_{T-t}(\rho_{T-t})\}.$$

Theorem 3.5 (Informal; formal statement in Theorem E.1)**.** *Under Assumption 3.1,*

$$\mathbb{P}(\widetilde{t}_{\text{in}} \le T - \delta) \ge 1 - e_\delta - \varepsilon(T) - \sqrt{\varepsilon_A(T,\delta)/2},$$

where $e_\delta \to 0$ as $\delta \to 0$, $\varepsilon(T) \to 0$ as $T \to \infty$, and $\varepsilon_A(T,\delta) = \int_\delta^T h_t^{-2} \mathbb{E}[\|m_A(t, X_t) - m(t, X_t)\|^2] dt$.

This is entry *by the terminal time*, not a bound locating entry near the start of inference or guaranteeing a minimum duration afterwards. The proof lower-bounds terminal tube occupancy, which is sufficient but not necessary for first entry. The error term transfers this property from the exact reverse process; approaching the ridge does not by itself imply recovery of individual training samples. The bound is informative when its error terms are sufficiently small.

3.3 Post-entry Alignment in Normal Directions

The squared off-ridge displacement is $D_{T-t}(x) := \|x - \Pi_{T-t}(x)\|^2$. The local differential estimate in Appendix F has the form

$$\frac{d}{dt}\mathbb{E}D_{T-t}(\widetilde{Y}_t) \le -2(\beta_{T-t}-1)\mathbb{E}D_{T-t}(\widetilde{Y}_t) + C(\rho_{T-t}R + d) + \frac{C\mathbb{E}\|e_A^{\perp}(T-t,\widetilde{Y}_t)\|^2}{h_{T-t}^2\beta_{T-t}},$$

with $e_A^{\perp}(t,x) = P_t^{\perp}(\Pi_t(x))e_A(t,x)$. The coefficient $2(\beta_{T-t}-1)$ is positive in the late-time regime. This identifies a normal restoring term, opposed by learning error, noise, and geometric remainders; it does not assert monotone decrease of every stochastic trajectory.

Theorem 3.6. *Under Assumption 3.1, with $\beta_t = c/h_t$, $c \in [1/2, 1)$, the terminal estimate for the regular post-entry regime is*

$$\mathbb{E}[D_\delta(\widetilde{Y}_{T-\delta})] = O\left(d\delta^c + \delta^c \int_{\widetilde{t}_{\mathrm{in}}}^{T-\delta} h_{T-u}^{-1-c}\mathbb{E}\|e_A^{\perp}(T-u,\widetilde{Y}_u)\|^2\,du\right).$$

The final squared off-ridge displacement is controlled by a training-independent geometric term and the accumulated normal learning residual. The full time-resolved estimate is in Theorem F.1. Thus small error transverse to the ridge supports good normal alignment, within the validity regime of the local analysis.

3.4 Post-entry Sliding in Tangent Directions

Sliding concerns a different component of the same trajectory. Near the end of inference, p_s is a Gaussian mixture centered at $m_s^{(i)} := a_s x_0^{(i)}$. Define the center-dominant region

$$B_s^{(i)}(\theta_s) = \left\{x : \mathrm{Softmax}\left(-\frac{\|x - m_s^{(j)}\|^2}{2h_s}\right)_i \ge 1-\theta_s\right\}, \qquad \theta_s = \exp(-o(h_s^{-1})).$$

In a regular center-dominant chart, define

$$\widetilde{u}_t^{(i)} := (U_{T-t}^{(i)})^\top(\widetilde{Y}_t - m_{T-t}^{(i)}) \in \mathbb{R}^{d^*}, \tag{9}$$

where $U_s^{(i)}$ contains the top d^* eigenvectors of $\nabla^2 \log p_s(m_s^{(i)})$. This is a local tangent-coordinate proxy relative to the nearby center (Appendix G), not a complete description of the along-ridge distribution. With $e_A^{\parallel,i}(s,x) = (U_s^{(i)})^\top e_A(s,x)$, Appendix G gives

$$\frac{d}{dt}\mathbb{E}\|\widetilde{u}_t^{(i)}\|^2 \le -\left(h_{T-t}^{-1} - 2\right)\mathbb{E}\|\widetilde{u}_t^{(i)}\|^2 + O(d) + \frac{4}{h_{T-t}}\mathbb{E}\|e_A^{\parallel,i}(T-t,\widetilde{Y}_t)\|^2.$$

The contraction coefficient is positive at sufficiently small diffusion time. Integrating yields the terminal estimate.

Theorem 3.7. *Under Assumption 3.1, in a regular center-dominant ridge chart where the local analysis applies, for $\delta \ll 1$,*

$$\mathbb{E}\|\widetilde{u}_{T-\delta}^{(i)}\|^2 = O\left(d\sqrt{\delta} + \sqrt{\delta}\int_{\widetilde{t}_{\mathrm{in}}}^{T-\delta} h_{T-u}^{-3/2}\mathbb{E}\|e_A^{\parallel,i}(T-u,\widetilde{Y}_u)\|^2\,du\right).$$

The tangential residual is controlled by a training-independent term and accumulated tangential error. Normal alignment can therefore coexist with an upper bound allowing substantial along-ridge spread. A nonvanishing upper bound permits, but does not by itself prove, non-memorizing generation. No theorem here requires normal alignment to finish before sliding starts.

Remark 3.8 (Training weights and small-noise scope)**.** *If the per-time weighted mean-square residual is uniformly controlled, the accumulated normal and tangential terms are bounded, respectively, by constants times*

$$\delta^c \int_\delta^{T-\widetilde{t}_{\mathrm{in}}} \frac{h_s^{1-c}}{w(s)}\, ds, \qquad \sqrt{\delta} \int_\delta^{T-\widetilde{t}_{\mathrm{in}}} \frac{h_s^{1/2}}{w(s)}\, ds.$$

Larger weights at small diffusion times reduce the permitted late-time residual. On a fixed late-time interval, these bounds vanish for $w = 1$ and $w = h_s$, whereas $w = h_s^2$ can leave a nonvanishing contribution. This statement is conditional on the error control and regular-chart regime, not an unconditional guarantee of memorization. The analysis stops at $\delta > 0$ and does not extend a tangent frame through an eigengap closure or a singular ridge transition. For a single point mass, $p_t = \mathcal{N}(a_t x_0, h_t I)$ has a zero-dimensional ridge at its mode: there is no nontrivial tangent space and no slide component.

4 From Training Error to Generation Geometry

Section 3 reduces generation geometry to directional errors: normal error controls alignment to the ridge, while tangential error controls a local center-relative residual along it. We now connect these quantities to training, first through projected posterior-mean matching losses and then through an RFNN+GD example.

4.1 Directional Decomposition of Training Loss

For $\dagger \in \{\perp, \|\}$, define

$$\mathcal{L}_{\mathrm{MM}}^{\dagger}(A) := \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E}\|P_t^{\dagger}(X_t) e_A(t, X_t)\|^2\, dt, \qquad \mathcal{L}_{\mathrm{MM}} = \mathcal{L}_{\mathrm{MM}}^{\perp} + \mathcal{L}_{\mathrm{MM}}^{\|}.$$

The training-to-inference transfer uses the change-of-measure, moment, and bounded weighted-error conditions in Assumption H.1; it is not an automatic consequence of a small aggregate training loss. The directional fields are those used in Appendix H, with the local geometric interpretation from Section 3.

Theorem 4.1 (Directional training losses control generation geometry)**.** *Under Assumption H.1 and the geometric conditions of Section 3, the normal and tangential bounds satisfy*

$$\begin{aligned} \textit{normal-error bound} &\lesssim C_\delta^{\perp} \mathcal{L}_{\mathrm{MM}}^{\perp} + d\delta^c + C_\delta^{\perp}(\sqrt{d} + R) e^{-T}, \\ \textit{tangent-error bound} &\lesssim C_\delta^{\|} \mathcal{L}_{\mathrm{MM}}^{\|} + d\sqrt{\delta} + C_\delta^{\|}(\sqrt{d} + R) e^{-T}, \end{aligned}$$

where $C_\delta^{\perp} = \delta^c\big(1 \vee \delta^{1-c}/w(\delta)\big)$ and $C_\delta^{\|} = \sqrt{\delta}\big(1 \vee \sqrt{\delta}/w(\delta)\big)$, with $c \in [1/2, 1)$.

Theorem 4.1, proved in Appendix H, converts the inference-path error terms into training-side quantities. Normal loss predicts alignment and tangential loss predicts the local sliding residual, with sensitivity coefficients and remainders that depend on the stopping time and training weight.

4.2 RFNN: Architecture and Optimization Effects on Generation

For the iteration (7), let $\lambda_1 = \lambda_{\max}(\widetilde{U})$ and assume $0 < \eta < 1/\lambda_1$. The limiting readout is

$$A_\infty = A_0(I - \widetilde{U}\widetilde{U}^{+}) + \widetilde{V}\widetilde{U}^{+}, \qquad A_k - A_\infty = (A_0 - A_\infty)(I - 2\eta\widetilde{U})^k.$$

Here $\widetilde{U}^{+}$ is the pseudoinverse. To state the directional decomposition explicitly, write $e_\infty = m_{A_\infty} - m$ and $r_k = m_{A_k} - m_{A_\infty}$. The quantities already derived in Appendix I are

$$\begin{aligned} \mathrm{Err}_{\mathrm{arc}}^{\dagger} &= \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E}\|P_t^{\dagger}(X_t) e_\infty(t, X_t)\|^2\, dt, \\ \mathrm{Err}_{\mathrm{train}}^{\dagger}(k) &= \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E}\left[\|P_t^{\dagger}(X_t) r_k(t, X_t)\|^2 + 2\langle P_t^{\dagger}(X_t) e_\infty(t, X_t), P_t^{\dagger}(X_t) r_k(t, X_t)\rangle\right] dt. \end{aligned}$$

Thus $\mathcal{L}^{\dagger}_{\mathrm{MM}}(A_k) = \mathrm{Err}^{\dagger}_{\mathrm{arc}} + \mathrm{Err}^{\dagger}_{\mathrm{train}}(k)$. The first term is the limiting finite-feature residual in that direction. The second is the finite-iteration correction, which tends to zero through the factors $(1 - 2\eta\lambda_j)^k$ for positive eigenvalues of $\widetilde{U}$. It includes a cross term and need not be nonnegative or monotone separately in each direction. Their sum over both directions is the nonnegative total optimization error (Proposition I.1).

Theorem 4.2 (Architecture and optimization control geometry)**.** *Under the conditions of Theorem 4.1, let $\{A_k\}$ follow (7) with $0 < \eta < 1/\lambda_1$. Up to the remainders in Theorem 4.1, the normal and tangential bounds are controlled by*

$$C^{\perp}_{\delta}\big(\mathrm{Err}^{\perp}_{\mathrm{arc}} + \mathrm{Err}^{\perp}_{\mathrm{train}}(k)\big), \qquad C^{\parallel}_{\delta}\big(\mathrm{Err}^{\parallel}_{\mathrm{arc}} + \mathrm{Err}^{\parallel}_{\mathrm{train}}(k)\big),$$

respectively. Explicit spectral formulas and the derivation are given in Appendix I.

This separates how finite representational capacity and finite training affect the two directional quantities. It does not assert that normal error is always easier to learn: the comparison depends on the data, features, and initialization.

Fully explicit results for two-point data. Consider $x_0^{(1)} = (-\mu, 0)$ and $x_0^{(2)} = (\mu, 0)$. The ridge and posterior mean are given in Section 3.1. Write the rows of A_0 as $A_{0,1}^{\top}, A_{0,2}^{\top}$. If $\{(\lambda_i, u_i)\}_{i=1}^{r}$ is the positive spectrum of $\widetilde{U}$ and $\widetilde{v}$ is the first row of $\widetilde{V}$, then

$$\mathrm{Err}^{\parallel}_{\mathrm{train}}(k) = \sum_{i=1}^{r} \lambda_i (1 - 2\eta\lambda_i)^{2k} \left(A_{0,1}^{\top} u_i - \frac{\widetilde{v} u_i}{\lambda_i} \right)^2,$$

$$\mathrm{Err}^{\perp}_{\mathrm{train}}(k) = \sum_{i=1}^{r} \lambda_i (1 - 2\eta\lambda_i)^{2k} (A_{0,2}^{\top} u_i)^2.$$

In particular, $A_{0,2} = 0$ gives $\mathrm{Err}^{\perp}_{\mathrm{train}}(k) = 0$: there is no training-induced normal error. This preserves the normal restoring dynamics rather than making all stochastic samples instantaneously lie on the ridge. If $A_{0,2}$ emphasizes slow spectral modes, normal optimization error decays slowly and can delay alignment.

The architecture residual is purely tangential: $\mathrm{Err}^{\perp}_{\mathrm{arc}} = 0$, while

$$\mathrm{Err}^{\parallel}_{\mathrm{arc}} = \int_{\delta}^{T} \frac{w(t)}{h_t^2} \mathbb{E}\left[a_t^2 \mu^2 \tanh^2\left(\frac{a_t \mu X_{t,1}}{h_t} \right) \right] dt - \widetilde{v}\widetilde{U}^{+}\widetilde{v}^{\top}.$$

A positive finite-feature tangential residual permits close normal alignment together with along-ridge spread. Increasing representational accuracy reduces this source of bias, while initialization controls the optimization contribution. This example isolates the two mechanisms without asserting a universal normal-before-tangent learning order. Numerical illustrations are in Appendix K.1.

5 Experiments

We evaluate the framework through 2D geometric illustrations, the explicit two-point problem, and MNIST latent diffusion. The first shows curved and edge-like generation, the second compares directional inference and training quantities, and the third tests the picture in higher dimension. Details are in Appendix J; additional results, including the supplied lattice experiments, are in Appendix K.

5.1 2D Illustrations of the Role of Ridge

We begin with qualitative 2D examples. These illustrate data-induced structures along which generated samples concentrate, rather than provide quantitative verification of all geometric hypotheses. For four training points at $(\pm 1, \pm 1)$, Figure 2 shows concentration along edge-like structures that are not themselves training samples. The moving ridge tracks these structures. For $(0, 0), (3, 0), (0, 5)$, Figure 3 shows a bent ridge and samples following this curved geometry, rather than arbitrary straight-line interpolation between all pairs of points.

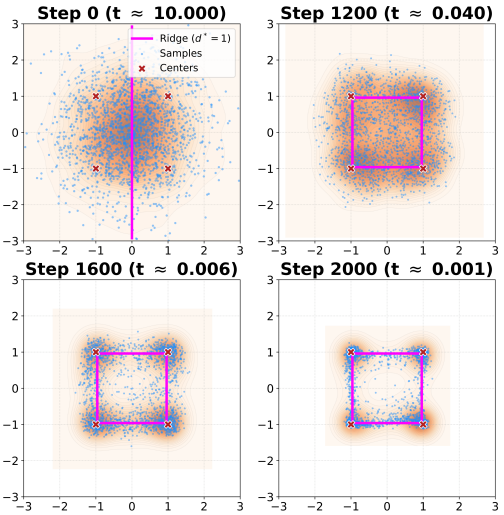


Figure 2: Generation from four training points $(\pm 1, \pm 1)$.

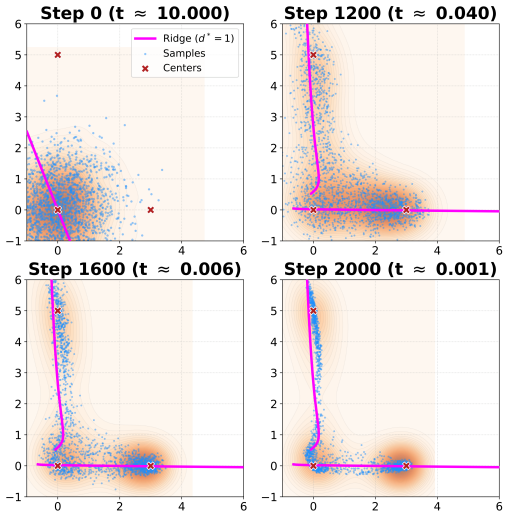
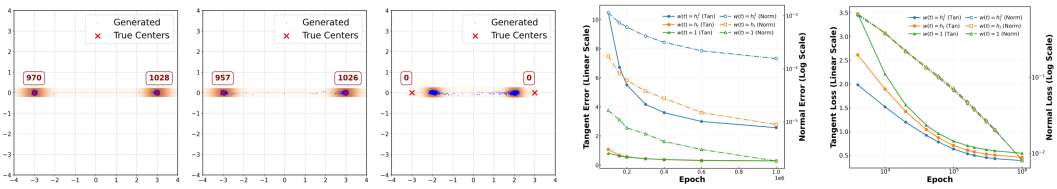


Figure 3: Generation from three training points $(0, 0), (3, 0), (0, 5)$.



(a) $w(t) = 1$ (b) $w(t) = h_t$ (c) $w(t) = h_t^2$ (d) Error Dynamics (e) Loss Decomposition

Figure 4: **RFNN generation, directional inference quantities, and training losses.** Left: samples under $w(t) = 1, h_t, h_t^2$; boxed counts are within radius 0.5 of the target modes, and shading shows a KDE. Right: tangential quantities (solid, linear scale) and normal quantities (dash-dot, logarithmic scale), under the same weighting schedules.

The examples also illustrate why ridge conditions matter. On the square diagonals, symmetry gives a repeated Hessian eigenvalue, so the strict one-dimensional eigengap fails. For the triangle, the third component contributes a nonzero score normal to the straight hypotenuse; this explains why pairwise line interpolation is not generally the appropriate geometric reference. These observations concern the displayed configurations and do not classify all branches of their small-noise ridges.

Appendix K.5 adds the existing 21 lattice configurations, varying angle, size, and translation. The supplied sample panels show structured generation and direction-dependent posterior-mean errors. Their error decomposition uses nearest straight lattice segments as proxies, not exact projections onto the curved log-density ridge; the appendix states this distinction explicitly.

5.2 Two Points in the 2D Plane

For the dataset $\{(\pm 3, 0)\}$, the ridge is the horizontal axis, with tangent $e_1 = (1, 0)$ and normal $e_2 = (0, 1)$. We compare generated sample geometry, directional inference quantities from Section 3, and directional training losses from Section 4. We report RFNN results here; MLP experiments and initialization effects are in Appendix K.

Directional geometry and its training origin. Across the three schedules in Figure 4, normal error is small and generated samples remain near $y = 0$. Tangential behavior changes substantially: $w = 1$ produces the smallest tangential residual and strongest concentration near the two training points; $w = h_t^2$ produces the largest residual and most pronounced edge-like spread; $w = h_t$ is intermediate. These observations are consistent with close normal alignment and a larger residual along the ridge.

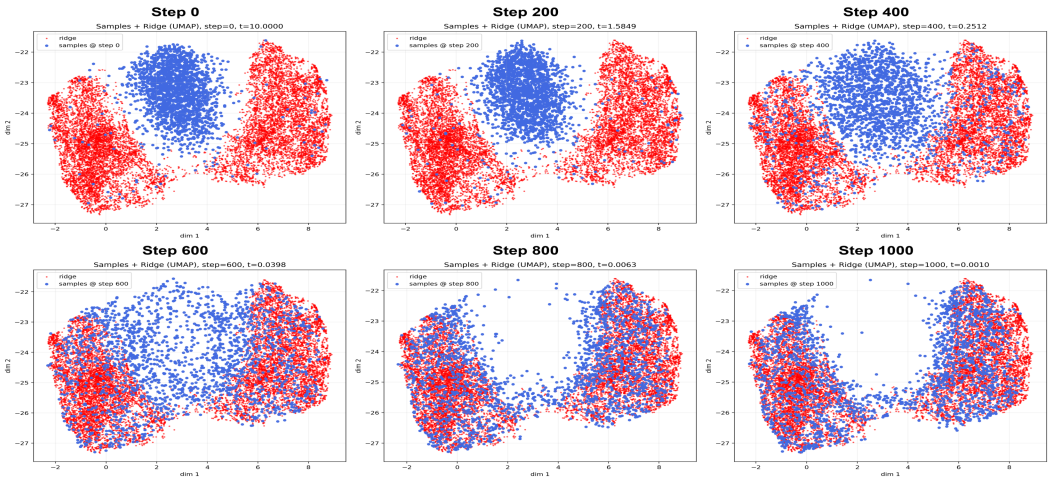


Figure 5: **UMAP visualization of generated samples and the reference ridge.** Red points show the computed reference ridge displayed at $t = 0.001$ in the supplied visualization; blue points show generated samples at successive inference steps. This is a qualitative embedding and does not preserve ridge distances or the normal/tangent decomposition.

The directional training losses provide the corresponding training-side comparison. Normal loss is small for all schedules. Raw tangential loss alone, however, does not determine final geometry: $w = h_t^2$ can have relatively small tangential training loss while producing substantial spread, because $C_\delta^{\parallel}$ amplifies late-time error when $w(\delta)$ is small. With $w = 1$, this amplification is weaker. The comparison therefore supports using directional losses together with their sensitivity coefficients rather than interpreting unweighted loss magnitudes in isolation.

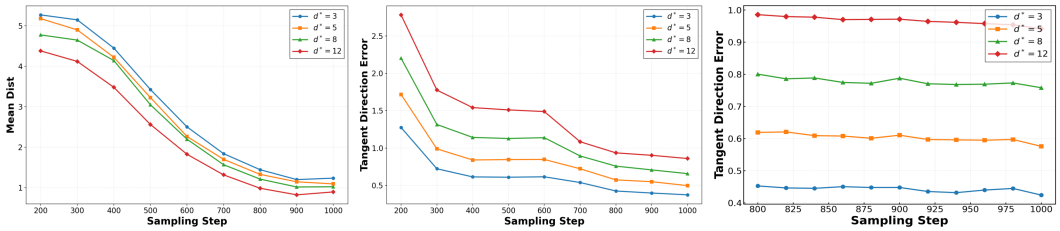
5.3 Higher-dimensional Data—MNIST

We study a binary MNIST problem in latent space and ask whether generated samples exhibit normal alignment and a persistent local tangential residual relative to training data.

Experimental setup. We use digits 4 and 8. A VAE embeds images into a 32-dimensional latent space; a time-conditioned MLP is then trained in that space with $w(t) = 1$. Architectural and training details are in Appendix J.3.

Qualitative visualization. In Figure 5, later generated samples move toward the global structure represented by the reference ridge. This provides a visual comparison, not a quantitative verification: UMAP need not preserve normal and tangent directions or Euclidean projection distances.

Quantitative normal and tangential behavior. We estimate ridge distance by solving the constrained projection problem in Appendix J.4. These are numerical projection estimates, not certificates of a globally nearest ridge point. Appendix J.5 reports convergence and eigengap diagnostics. The empirical projection calculation uses the fixed threshold reported there, $\beta = 10^{-3}$, rather than the theoretical scaling c/h_t. The earliest high-noise measurements are omitted because near-isotropy makes the eigenspace split poorly conditioned. Figure 6(a) shows decreasing mean distance through most of inference, followed by a small floor around step 900. Tangential residuals behave differently and remain nonzero late in inference (panels (b)–(c)). The apparent temporal imbalance is consistent with strong alignment and incomplete center-relative tangential contraction; it does not establish an exact switching time between mechanisms. The image trajectories in Appendix K.4 provide a complementary qualitative view. Interpreting these quantities requires the spectral and solver checks in Appendix J.5, particularly when the eigengap becomes small.



(a) distance to ridge (b) tangent error (full) (c) tangent error (zoom)

Figure 6: **Ridge-related metrics during MNIST inference.** (a) Mean numerical ridge-distance estimate across 200 trajectories, for $d^* = 3, 5, 8, 12$, starting at step 200. (b) Tangential residuals from step 200. (c) The final 200 steps. A residual plateau is not a measurement of instantaneous tangential velocity or pathwise stationarity.

6 Conclusions and Limitations

This work gives a data-dependent geometric perspective on diffusion generation. A time-dependent log-density ridge induced by the training data provides a reference for tube entry and for the subsequent, potentially simultaneous normal and tangential effects. The analysis clarifies how learning error enters these effects: normal residuals control alignment, while tangential residuals control local displacement along the ridge. RFNN+GD makes the architecture and finite-iteration contributions explicit.

The theory concerns continuous-time inference and regular ridge neighborhoods, with positive stopping noise. It does not establish disjoint temporal stages, global validity across singular ridge transitions, or the complete probability law along the ridge. Our explicit training decomposition is for RFNNs, not a full account of modern architectures; the direction that is learned more accurately depends on the setting. We do not analyze discretization error here. Existing sampler analyses [3, 8, 9, 12, 13, 24, 35] motivate studying its geometric effects, but no discretized reach–align–slide theorem is claimed. Finally, high-dimensional ridge estimation and projection are computationally demanding, and the supplied lattice diagnostics use segment proxies. Ridge organization alone does not distinguish useful semantic generalization from hallucination.

Acknowledgments and Disclosure of Funding

MT thanks Hongkai Zhao, Mikhail Belkin and Peter L. Bartlett for insightful discussions. MT is grateful for partial supports by NSF Grant DMS-2513699, DOE Grant SC0026274, and Richard Duke Fellowship.

Impact Statement

This paper presents work whose goal is to advance the field of machine learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

References

[1] E. Aamari, J. Kim, F. Chazal, B. Michel, A. Rinaldo, and L. Wasserman. Estimating the reach of a manifold, 2019.

[2] R. Baptista, A. Dasgupta, N. B. Kovachki, A. Oberai, and A. M. Stuart. Memorization and regularization in generative diffusion models, 2025. arXiv:2501.15785.

[3] J. Benton, V. De Bortoli, A. Doucet, and G. Deligiannidis. Nearly d-linear convergence bounds for diffusion models via stochastic localization, 2024. International Conference on Learning Representations.

[4] Q. Bertrand, A. Gagneux, M. Massias, and R. Emonet. On the closed-form of flow matching: Generalization does not arise from target stochasticity, 2025. arXiv:2506.03719.

[5] T. Bonnaire, R. Urfin, G. Biroli, and M. Mëzard. Why diffusion models don't memorize: The role of implicit dynamical regularization in training, 2025. arXiv:2505.17638.

[6] T. Brooks, B. Peebles, C. Holmes, W. DePue, Y. Guo, L. Jing, D. Schnurr, J. Taylor, T. Luhman, E. Luhman, et al. Video generation models as world simulators, 2024. OpenAI Blog, 1(8):1.

[7] N. Carlini, J. Hayes, M. Nasr, M. Jagielski, V. Sehwag, F. Tramer, B. Balle, D. Ippolito, and E. Wallace. Extracting training data from diffusion models, 2023. 32nd USENIX Security Symposium, pp. 5253–5270.

[8] H. Chen, H. Lee, and J. Lu. Improved analysis of score-based generative modeling: User-friendly bounds under minimal smoothness assumptions, 2023. International Conference on Machine Learning, pp. 4735–4763.

[9] S. Chen, S. Chewi, J. Li, Y. Li, A. Salim, and A. R. Zhang. Sampling is as easy as learning the score: Theory for diffusion models with minimal data assumptions, 2023. International Conference on Learning Representations.

[10] Y.-C. Chen, C. R. Genovese, and L. Wasserman. Asymptotic theory for density ridges, 2015. The Annals of Statistics, pp. 1896–1928.

[11] Z. Chen. On the interpolation effect of score smoothing, 2025. arXiv:2502.19499.

[12] G. Conforti, A. Durmus, and M. G. Silveri. Score diffusion models without early stopping: Finite Fisher information is all you need, 2023. arXiv:2308.12240.

[13] V. De Bortoli. Convergence of denoising diffusion models under the manifold hypothesis, 2022. Transactions on Machine Learning Research.

[14] P. Dhariwal and A. Nichol. Diffusion models beat GANs on image synthesis, 2021. Advances in Neural Information Processing Systems, 34:8780–8794.

[15] J. Duan, F. Kong, S. Wang, X. Shi, and K. Xu. Are diffusion models vulnerable to membership inference attacks?, 2023. International Conference on Machine Learning, pp. 8717–8730.

[16] T. Farghly, P. Potaptchik, S. Howard, G. Deligiannidis, and J. Pidstrigach. Diffusion models and the manifold hypothesis: Log-domain smoothing is geometry adaptive, 2025. arXiv:2510.02305.

[17] C. R. Genovese, M. Perone-Pacifico, I. Verdinelli, and L. Wasserman. Nonparametric ridge estimation, 2014.

[18] A. J. George, R. Veiga, and N. Macris. Denoising score matching with random features: Insights on diffusion models from precise learning curves, 2025. arXiv:2502.00336.

[19] Y. He, K. Rojas, and M. Tao. Zeroth-order sampling methods for non-log-concave distributions: Alleviating metastability by denoising diffusion, 2024. Advances in Neural Information Processing Systems, 37:71122–71161.

[20] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models, 2020. Advances in Neural Information Processing Systems, 33:6840–6851.

[21] Z. Kadkhodaie, F. Guth, E. P. Simoncelli, and S. Mallat. Generalization in diffusion models arises from geometry-adaptive harmonic representations, 2023. arXiv:2310.02557.

[22] M. Kamb and S. Ganguli. An analytic theory of creativity in convolutional diffusion models, 2024. arXiv:2412.20292.

[23] Z. Kong, W. Ping, J. Huang, K. Zhao, and B. Catanzaro. DiffWave: A versatile diffusion model for audio synthesis, 2021. International Conference on Learning Representations.

[24] H. Lee, J. Lu, and Y. Tan. Convergence for score-based generative modeling with polynomial complexity, 2022. Advances in Neural Information Processing Systems, 35:22870–22882.

[25] G. Leobacher and A. Steinicke. Existence, uniqueness and regularity of the projection onto differentiable manifolds, 2021. Annals of Global Analysis and Geometry, 60(3):559–587.

[26] X. Li, Z. Shen, Y.-P. Hsieh, and N. He. When scores learn geometry: Rate separations under the manifold hypothesis, 2025. arXiv:2509.24912.

[27] Y. Liu, J. Huang, Y. Li, D. Wang, and B. Xiao. Generative AI model privacy: A survey, 2024. Artificial Intelligence Review, 58(1):33.

[28] K. Moshksar. Refining concentration for Gaussian quadratic chaos, 2024. arXiv:2412.03774.

[29] P. Niyogi, S. Smale, and S. Weinberger. Finding the homology of submanifolds with high confidence from random samples, 2008. Discrete & Computational Geometry, 39(1):419–441.

[30] K. Shah, A. Kalavasis, A. R. Klivans, and G. Daras. Does generation require memorization? Creative diffusion models using ambient diffusion, 2025. arXiv:2502.21278.

[31] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics, 2015. International Conference on Machine Learning, pp. 2256–2265.

[32] G. Somepalli, V. Singla, M. Goldblum, J. Geiping, and T. Goldstein. Understanding and mitigating copying in diffusion models, 2023. Advances in Neural Information Processing Systems, 36:47783–47803.

[33] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. Score-based generative modeling through stochastic differential equations, 2021. International Conference on Learning Representations.

[34] J. J. Vastola. Generalization through variance: How noise shapes inductive biases in diffusion models, 2025. arXiv:2504.12532.

[35] Y. Wang, Y. He, and M. Tao. Evaluating the design space of diffusion-based generative models, 2024. Advances in Neural Information Processing Systems, 37:19307–19352.

[36] Y.-H. Wu, P. Marion, G. Biau, and C. Boyer. Taking a big step: Large learning rates in denoising score matching prevent memorization, 2025. arXiv:2502.03435.

[37] Z. Ye, Q. Zhu, M. Tao, and M. Chen. Provable separations between memorization and generalization in diffusion models, 2025. arXiv:2511.03202.

[38] H. Zhang, J. Zhou, Y. Lu, M. Guo, P. Wang, L. Shen, and Q. Qu. The emergence of reproducibility and consistency in diffusion models, 2023. arXiv:2310.05264.

A Related Work

This appendix expands the brief discussion in the introduction and clarifies how our notion of generalization and our ridge-based analysis relate to several nearby directions in the diffusion-model literature. Our goal is not to survey the entire area, but to explain more precisely which question our paper addresses and how it connects to existing viewpoints.

Population-level generalization versus our data-dependent question. A substantial line of work [e.g., 4, 5, 35, 37] studies diffusion-model generalization by comparing the generated distribution to an unknown population distribution and deriving bounds in global discrepancy metrics. This perspective is natural when the goal is population-level recovery or distributional approximation. Our paper addresses a different question. We take the finite training dataset itself as the primary reference object and ask where generated samples go relative to the geometry induced by that dataset. In this sense, our focus is not primarily on global closeness to an unknown population law, but on the geometric organization of non-memorizing generations relative to the observed data. This viewpoint is especially useful when the phenomenon of interest is structured intermediate generation between training samples, since the relevant issue is not only how different two distributions are, but also how generated samples are spatially arranged.

Target-side stochasticity and finite-data target structure. One line of work [e.g., 4, 34] asks whether generalization can already arise from the stochasticity or structure of the finite-data training target itself. From this viewpoint, the learned diffusion model may generate non-memorizing samples not only because of imperfections in training or inference, but also because the empirical target differs from a population-level object in a structured way. Our framework is related to this direction in that it also adopts a fully finite-data viewpoint. However, our emphasis is different: rather than analyzing the stochastic gap between empirical and population targets, we take the empirical dataset as given and study how reverse-time inference organizes samples relative to the geometry induced by that dataset.

Training-induced bias. A closely related direction studies the inductive bias created during training, for example through model class, feature learning, optimization dynamics, or finite training time [e.g., 5, 22, 30, 36]. This literature explains how the learned score or posterior mean differs from the ideal one and how such differences depend on architecture and optimization. Our contribution is complementary in two ways. First, rather than stopping at aggregate training or test error, we identify directional components of training error relative to a data-dependent geometric object. Second, in the RFNN setting, building on random-feature analyses of diffusion training [5, 18], we show how limited expressivity due to finite width and incomplete optimization translate into different geometric effects during inference: normal components of error control alignment to the ridge, while tangential components control spreading along it. In this sense, our framework explains not only what bias training creates, but also how that bias appears geometrically during sampling.

Inference-time bias accumulation: metric and geometric viewpoints. Another broad perspective studies how errors accumulate through inference. One version of this literature [e.g., 3, 8, 24, 35] characterizes the gap between exact and learned reverse processes through divergence-type quantities such as KL or TV. Such results are useful for quantifying distributional discrepancy, but by themselves they say relatively little about the geometry of generated samples or how those samples are organized relative to the training data beyond a global metric.

A second version [2, 11, 16, 26] takes a more geometric viewpoint, often under manifold-type assumptions on the data distribution. Our work is closest in spirit to this line, but differs in several important respects. First, we make the relevant data structure explicit by constructing a time-indexed family of log-density ridge sets directly from the smoothed empirical distribution, rather than assuming an underlying manifold a priori. This differs from Farghly et al. [16], which motivates log-density smoothing as a useful analytical lens but does not study simulated inference trajectories or provide a data-dependent ridge-manifold description. It is also complementary to Baptista et al. [2], which analyzes memorization through the reverse dynamics induced by the empirical-loss minimizer using Voronoi geometry, whereas our focus is the complementary non-memorizing regime, where we identify a time-dependent ridge geometry from the smoothed empirical distribution and analyze how inference evolves relative to it. Second, we analyze how inference trajectories evolve relative

to this geometry through a reach–align–slide mechanism. This goes beyond concentration near a low-dimensional set by capturing part of the tangential organization of generation relative to nearby data, while remaining intentionally partial: the analysis predicts tangential motion toward nearby data-induced centers rather than fully characterizing the generated configuration inside the tangent space. Third, our setting complements Li et al. [26], which studies the transition between sampling uniformly from a low-dimensional manifold and sampling a target distribution supported on that manifold. In contrast, we study full denoising diffusion, where the noise level is time-dependent and vanishes as sampling approaches the data, and we explicitly quantify how the learned score or posterior mean estimator drives inference around the geometric object governing generation. Finally, compared with analytical explanations of interpolation bias in stylized settings or under imposed error ansatz [e.g., 11], our conclusions are formulated under regularity conditions and connect the resulting geometric bias explicitly to both the dataset and the training parameters.

How our framework combines these perspectives. Viewed together, our framework combines several of the above viewpoints in a single data-driven analysis. We work directly with the empirical data distribution, avoiding any need to assume an underlying smooth population distribution or a fixed manifold known in advance. We then construct a time-dependent ridge family adapted to the diffusion noise level and analyze reverse-time inference relative to this family through a reach–align–slide mechanism. Finally, in the RFNN+GD setting, we decompose the relevant directional training errors into architecture-driven and optimization-driven terms, making explicit how model class, training procedure, and inference geometry interact. Throughout, the analysis is nonasymptotic: the dataset is finite, and in the RFNN example the data dimension, sample size, network width, and training time are all kept finite rather than taken to infinity.

Summary. Relative to nearby work, the main distinction of our paper is therefore not a single isolated technical improvement, but a shift in viewpoint. We study non-memorizing generation in a fully data-dependent setting, identify an explicit time-dependent geometric object from the empirical data, analyze reverse-time inference relative to that object through reach–align–slide, and connect the resulting geometry back to directional components of training error.

B Denoising Mean Matching Loss

In this appendix, we relate the posterior mean-matching loss $\mathcal{L}_{\mathrm{MM}}$ to its denoising version $\mathcal{L}_{\mathrm{DMM}}$. Recall that $X_t = a_t X_0 + \sqrt{h_t} Z$, where $Z \sim \mathcal{N}(0, I_d)$ is independent of X_0, and $m(t,x) = \mathbb{E}[a_t X_0 \mid X_t = x]$. By Tweedie's formula,

$$\nabla \log p_t(x) = \frac{m(t,x) - x}{h_t}, \qquad s_A(t,x) = \frac{m_A(t,x) - x}{h_t}.$$

Thus the posterior mean-matching loss is exactly the corresponding weighted score-matching loss:

$$\begin{aligned}\mathcal{L}_{\mathrm{MM}}(A) &= \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E}\big[\|m_A(t,X_t) - m(t,X_t)\|^2\big] dt \\ &= \int_\delta^T w(t) \mathbb{E}\big[\|s_A(t,X_t) - \nabla \log p_t(X_t)\|^2\big] dt.\end{aligned}$$

Although m is generally unavailable for training, its conditional-expectation property gives a denoising objective. Write

$$m_A(t,X_t) - a_t X_0 = \big(m_A(t,X_t) - m(t,X_t)\big) + \big(m(t,X_t) - a_t X_0\big).$$

The first summand is measurable with respect to X_t, whereas the second has conditional mean zero. Therefore,

$$\begin{aligned}&\mathbb{E}\big[\langle m_A(t,X_t) - m(t,X_t),\, m(t,X_t) - a_t X_0\rangle\big] \\ &\quad = \mathbb{E}\big[\langle m_A(t,X_t) - m(t,X_t),\, \mathbb{E}[m(t,X_t) - a_t X_0 \mid X_t]\rangle\big] = 0.\end{aligned}$$

Expanding the squared norm and integrating consequently yields

$$\mathcal{L}_{\mathrm{DMM}}(A) = \mathcal{L}_{\mathrm{MM}}(A) + C, \qquad C := \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E}\big[\|m(t,X_t) - a_t X_0\|^2\big] dt \geq 0.$$

Equivalently, $\mathcal{L}_{\mathrm{MM}}(A) = \mathcal{L}_{\mathrm{DMM}}(A) - C$. The constant C is independent of A, so the two objectives have the same minimizers and, whenever differentiable, the same gradient with respect to A.

The denoising objective can be evaluated using clean data and Gaussian corruptions:

$$\mathcal{L}_{\mathrm{DMM}}(A) = \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E}_{X_0} \mathbb{E}_Z \big[\|m_A(t, a_t X_0 + \sqrt{h_t} Z) - a_t X_0\|^2\big] dt.$$

Under Assumption 3.1, $p = \frac{1}{n} \sum_{i=1}^n \delta_{x_0^{(i)}}$, so the expectation over X_0 is an exact empirical average:

$$\mathcal{L}_{\mathrm{DMM}}(A) = \frac{1}{n} \sum_{i=1}^n \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E}_Z \big[\|m_A(t, a_t x_0^{(i)} + \sqrt{h_t} Z) - a_t x_0^{(i)}\|^2\big] dt. \tag{10}$$

Our analysis uses exact evaluation of the Gaussian expectation and the time integral in (10). In practice, both are approximated numerically. For example, with independent $Z^{(i,j)} \sim \mathcal{N}(0, I_d)$ and a grid $\delta = t_0 < t_1 < \cdots < t_N = T$, the right-endpoint approximation is

$$\mathcal{L}_{\mathrm{DMM}}^{m,N}(A) = \frac{1}{nm} \sum_{i=1}^n \sum_{j=1}^m \sum_{k=1}^N \frac{(t_k - t_{k-1}) w(t_k)}{h_{t_k}^2} \big\|m_A(t_k, a_{t_k} x_0^{(i)} + \sqrt{h_{t_k}} Z^{(i,j)}) - a_{t_k} x_0^{(i)}\big\|^2.$$

For the trapezoidal rule used in the experiments, replace the right-endpoint weights by the corresponding trapezoidal quadrature weights, including both endpoints. The factor $w(t_k)$ must be retained for a nonconstant training-weight schedule.

C Data-independent Properties of the Log-density Ridge Sets

In this section, we introduce properties of the log-density ridges that are independent of our data assumptions. We summarize them in the following proposition.

Proposition C.1. *Under Assumption 3.3, for any $\rho_t \in (0, r_t)$ and the tube neighborhood*

$$\mathcal{T}_t(\rho_t) := \{x \in \mathbb{R}^d : \operatorname{dist}(x, \mathcal{R}_t) \le \rho_t\}, \tag{11}$$

the nearest-point projection $\Pi_t : \mathcal{T}_t(\rho_t) \to \mathcal{R}_t$ is well-defined and satisfies:

1. *Π_t is C^1 on $\mathcal{T}_t(\rho_t)$;*
2. *$n_t(x) := x - \Pi_t(x)$ belongs to $N_{\Pi_t(x)}(\mathcal{R}_t)$;*
3. *$\sup_{x \in \mathcal{T}_t(\rho_t)} \|\nabla \Pi_t(x)\| \le (1 - \rho_t/r_t)^{-1}$;*
4. *for the normal velocity field $v_t(z) \in N_z(\mathcal{R}_t)$,*

$$\sup_{x \in \mathcal{T}_t(\rho_t)} \|\partial_t \Pi_t(x)\| \le V_t + \frac{\rho_t}{1 - \rho_t/r_t} W_t,$$

where

$$V_t := \sup_{z \in \mathcal{R}_t} \|v_t(z)\|, \qquad W_t := \sup_{z \in \mathcal{R}_t, \|u\|=1} \|P^{\parallel}(z)(\nabla v_t(z) u)\|.$$

Here $P^{\parallel}(z)$ and $P^{\perp}(z)$ denote the orthogonal projectors onto $T_z(\mathcal{R}_t)$ and $N_z(\mathcal{R}_t)$, respectively.

Proof of Proposition C.1. That Π_t is well-defined follows from Assumption 3.3. Its C^1 spatial regularity follows from Leobacher and Steinicke [25, Theorem 2]. For (2), $z = \Pi_t(x)$ minimizes $\|x - z'\|^2$ over $z' \in \mathcal{R}_t$. Differentiating along $u \in T_z(\mathcal{R}_t)$ gives

$$D(\|x - z'\|^2)[u]\big|_{z'=z} = -2\langle x - z, u\rangle = -2\langle n_t(x), u\rangle = 0.$$

Thus $n_t(x) \in N_z(\mathcal{R}_t)$. To prove (3), use the expression in Leobacher and Steinicke [25, Theorem C]:

$$\nabla \Pi_t(x) = \big(\mathrm{id}_{T_{\Pi_t(x)}(\mathcal{R}_t)} - \|x - \Pi_t(x)\| S_{t,\Pi_t(x),\theta}\big)^{-1} P^{\parallel}(\Pi_t(x)), \qquad \theta = \frac{x - \Pi_t(x)}{\|x - \Pi_t(x)\|}.$$

Here $S_{t,z,\theta}$ is the shape operator. By Lemma C.3 and its linearity in the normal argument,

$$\begin{aligned} \|\nabla \Pi_t(x)\| &\le \|\big(\mathrm{id}_{T_{\Pi_t(x)}(\mathcal{R}_t)} - S_{t,\Pi_t(x),x-\Pi_t(x)}\big)^{-1}\| \\ &\le \frac{1}{1 - \|x - \Pi_t(x)\|/r_t} \le \frac{1}{1 - \rho_t/r_t}. \end{aligned}$$

For (4), Lemma C.4 supplies the normal velocity field; the bound then follows from the definition of V_t and Lemma C.5. □

Definition C.2 (Shape operator)**.** *Let $\mathrm{II}_{t,z}$ be the second fundamental form of $\mathcal{R}_t$ at z. For $\theta \in N_z(\mathcal{R}_t)$, the shape operator $S_{t,z,\theta} : T_z(\mathcal{R}_t) \to T_z(\mathcal{R}_t)$ is defined by*

$$\langle S_{t,z,\theta} u, v\rangle = \langle \mathrm{II}_{t,z}(u, v), \theta\rangle, \qquad u, v \in T_z(\mathcal{R}_t).$$

Lemma C.3. *Under Assumption 3.3, for $z \in \mathcal{R}_t$ and $\theta \in N_z(\mathcal{R}_t)$:*

1. *$\|S_{t,z,\theta}\| \le \|\theta\|/r_t$;*
2. *if $\|\theta\| \le \rho_t < r_t$, then $L_{t,z,\theta} := \mathrm{id}_{T_z(\mathcal{R}_t)} - S_{t,z,\theta}$ is invertible and*

$$\|L_{t,z,\theta}^{-1}\| \le \frac{1}{1 - \|\theta\|/r_t}.$$

Proof. By Niyogi et al. [29, Proposition 6.1], $\|\mathrm{II}_{t,z}\| \le 1/r_t$. Therefore $\|S_{t,z,\theta}\| \le \|\theta\|/r_t$, and the inverse bound follows from the Neumann-series estimate. □

Lemma C.4. *Fix $t \in [\delta, T]$. Let $U \subset \mathbb{R}^{d^*}$ be open and $\widetilde{\Phi}_t : U \to \mathbb{R}^d$ a C^1 family of C^2 embeddings with $\widetilde{\Phi}_t(U) \subset \mathcal{R}_t$, parametrizing a given C^1 curve $z(t) \in \mathcal{R}_t$. Suppose $z(t) = \widetilde{\Phi}_t(\widetilde{u}(t))$ for a C^1 curve $\widetilde{u}(t) \in U$. There exists a C^1 family of local diffeomorphisms $\psi_t : U' \to U$ such that*

$$\Phi_t := \widetilde{\Phi}_t \circ \psi_t : U' \to \mathcal{R}_t, \qquad \partial_t \Phi_t(u) \perp T_{\Phi_t(u)}(\mathcal{R}_t).$$

Consequently, $v_t(z) := \partial_t \Phi_t(u)$, $z = \Phi_t(u)$, is a well-defined continuous normal velocity field. It equals the normal component of $\partial_t \widetilde{\Phi}_t$ and is independent of tangential reparametrization.

Proof. Put $\widetilde{v}_t(u) = \partial_t \widetilde{\Phi}_t(u)$. Since $\widetilde{\Phi}_t$ is an embedding, $\nabla \widetilde{\Phi}_t(u) : \mathbb{R}^{d^*} \to T_{\widetilde{\Phi}_t(u)}(\mathcal{R}_t)$ is an isomorphism. Define $a_t(u)$ by

$$\nabla \widetilde{\Phi}_t(u) a_t(u) = -P^{\parallel}(\widetilde{\Phi}_t(u)) \widetilde{v}_t(u). \tag{12}$$

This determines a_t uniquely; the assumed regularity gives continuity in time and smoothness in u. On a local domain U' containing the relevant curve, consider

$$\partial_t \psi_t(u) = a_t(\psi_t(u)), \qquad \psi_\delta(u) = u \in U'. \tag{13}$$

The local flow is a diffeomorphism. The chain rule gives

$$\begin{aligned}
\partial_t \Phi_t(u) &= \widetilde{v}_t(\psi_t(u)) + \nabla \widetilde{\Phi}_t(\psi_t(u)) \partial_t \psi_t(u) \\
&= \widetilde{v}_t(\psi_t(u)) + \nabla \widetilde{\Phi}_t(\psi_t(u)) a_t(\psi_t(u)) \\
&= \widetilde{v}_t(\psi_t(u)) - P^{\parallel}(\widetilde{\Phi}_t(\psi_t(u))) \widetilde{v}_t(\psi_t(u)) \\
&= P^{\perp}(\widetilde{\Phi}_t(\psi_t(u))) \widetilde{v}_t(\psi_t(u)).
\end{aligned}$$

Thus $\partial_t \Phi_t(u)$ is normal to $\mathcal{R}_t$. Its value is the normal part of the original velocity, independently of the choice of tangent coordinates. □

Lemma C.5. *Let $x \in \mathcal{T}_t(\rho_t)$ and $z(t) = \Pi_t(x)$. Under Lemma C.4 and Assumption 3.3, a normal-gauge chart Φ_t and a C^1 curve $u(t) \in U \subset \mathbb{R}^{d^*}$ satisfy $z(t) = \Phi_t(u(t))$ and*

$$\partial_t \Pi_t(x) = v_t(z(t)) + \tau_t, \qquad \tau_t := \nabla_u \Phi_t(u(t)) \partial_t u(t) \in T_{z(t)}(\mathcal{R}_t).$$

Furthermore, $\|\tau_t\| \le \rho_t (1 - \rho_t / r_t)^{-1} W_t$.

Proof. Differentiate $z(t) = \Phi_t(u(t))$ to obtain $\partial_t z = v_t(z) + \tau_t$. Let $E_i(t) = \partial_{u_i} \Phi_t(u(t))$ be the local tangent frame. Orthogonality gives $\langle n_t(x), E_i(t) \rangle = 0$; differentiating and using $v_t \perp E_i$ yields

$$\langle \tau_t, E_i(t) \rangle = \langle n_t(x), \partial_t E_i(t) \rangle. \tag{14}$$

The moving frame satisfies

$$\begin{aligned}
\partial_t E_i(t) &= \partial_{u_i} \partial_t \Phi_t(u(t)) + \sum_{j=1}^{d^*} \partial^2_{u_i u_j} \Phi_t(u(t)) \partial_t u_j(t) \\
&= \nabla v_t(z) E_i(t) + \sum_{j=1}^{d^*} \mathrm{II}_{t,z}(E_i(t), E_j(t)) \partial_t u_j(t) \\
&\quad + \sum_{j=1}^{d^*} P^{\parallel}(z) \partial^2_{u_i u_j} \Phi_t(u(t)) \partial_t u_j(t).
\end{aligned}$$

Since $n_t(x)$ is normal and $\tau_t = \sum_j \partial_t u_j(t) E_j(t)$,

$$\langle n_t(x), \partial_t E_i(t) \rangle = \langle n_t(x), \nabla v_t(z) E_i(t) \rangle + \langle S_{t,z,n_t(x)} \tau_t, E_i(t) \rangle.$$

Substituting in (14), restricted to $T_z(\mathcal{R}_t)$,

$$(I - S_{t,z,n_t(x)}) \tau_t = P^{\parallel}(z) (\nabla v_t(z))^{\top} n_t(x).$$

Using Lemma C.3 and the velocity-derivative bound gives

$$\|\tau_t\| \le \|L^{-1}_{t,z,n_t(x)}\| \|\nabla v_t(z)\| \|n_t(x)\| \le \frac{\rho_t}{1 - \rho_t / r_t} W_t.$$

□

D Data-dependent Ridge Motion Estimations

In this section, we provide data-dependent estimates for quantities related to the dynamical properties of the log-density ridges.

Proposition D.1. *Under Assumption 3.1, the log-density ridge sets satisfy Assumption 3.3. Furthermore, when $t \to \delta^+ \ll 1$,*

$$r_t = \Omega(\beta_t h_t^3 \theta_t^{-1} R^{-3}), \qquad V_t = O(\beta_t^{-1} h_t^{-1} R), \qquad W_t = O(\beta_t^{-1} h_t^{-1} R r_t^{-1}),$$

where θ_t is chosen with order $\theta_t = \exp(-o(h_t^{-1}))$.

Remark D.2 (Order estimation of ridge motion)**.** *Combining Propositions C.1 and D.1, taking $\rho_t = \Theta(r_t)$ and $\beta_t = \Theta(1/h_t)$ gives Proposition 3.4(3): $\sup_{x \in \mathcal{T}_t(\rho_t)} \|\partial_t \Pi_t(x)\| = O(R)$.*

To study the relation between Assumption 3.1 and the geometry of $\mathcal{R}_t$, we estimate derivatives of the score in regions dominated by centers $m_s^{(i)} := a_s x_0^{(i)}$. For $i \in [n]$ and $0 < \theta_s < 1$, define

$$B_s^{(i)}(\theta_s) := \left\{ x \in \mathbb{R}^d : \mathrm{Softmax}\left(-\frac{\|x - m_s^{(j)}\|^2}{2h_s} \right)_i \geq 1 - \theta_s \right\}. \tag{15}$$

Lemma D.3. *For $x \in \mathbb{R}^d$ and $\theta_s \in (0, 1/2)$, if*

$$\|x - m_s^{(j)}\|^2 - \|x - m_s^{(i)}\|^2 \geq 2h_s \log\left(\frac{(1-\theta_s)(n-1)}{\theta_s} \right), \qquad j \neq i, \tag{16}$$

then $x \in B_s^{(i)}(\theta_s)$.

Remark D.4. *As $t \to T^-$, if $h_{T-t} = o(1)$ and $\theta_{T-t} = \exp(-o(h_{T-t}^{-1}))$, the right-hand side of (16) is $o(1)$. The center-dominant analysis below is used on regions satisfying (16).*

Proof of Lemma D.3. Write $\gamma_j(s,x) = \mathrm{Softmax}(-\|x - m_s^{(\ell)}\|^2/(2h_s))_j$. For $j \neq i$, (16) implies

$$\frac{\gamma_j(s,x)}{\gamma_i(s,x)} = \exp\left(-\frac{\|x - m_s^{(j)}\|^2 - \|x - m_s^{(i)}\|^2}{2h_s} \right) \leq \frac{\theta_s}{(1-\theta_s)(n-1)}.$$

Consequently,

$$1 - \gamma_i(s,x) = \sum_{j \neq i} \gamma_j(s,x) \leq \frac{\theta_s}{1-\theta_s} \gamma_i(s,x),$$

which gives $\gamma_i(s,x) \geq 1 - \theta_s$. □

Lemma D.5. *For $t \in [\delta, T]$,*

$$\sup_{x \in \bigcup_i B_t^{(i)}(\theta_t)} \|\nabla m(t,x)\| \leq \frac{20 a_t^2 \theta_t R^2}{h_t}, \qquad \sup_{x \in \bigcup_i B_t^{(i)}(\theta_t)} \|\nabla^3 \log p_t(x)\| \leq \frac{80 a_t^3 \theta_t R^3}{h_t^3},$$

where the center-dominant regions are given by (15).

Remark D.6 (Choice of β_t)**.** *Lemma D.5 validates the choice $\beta_t = \Theta(1/h_t)$ in the center-dominant small-noise regime. Lemma D.7 gives $\nabla^2 \log p_t = -I_d/h_t + \Sigma(t,x)/h_t^2$. Thus every eigenvalue satisfies*

$$\lambda_j(t,x) \leq -\frac{1}{h_t} + \frac{20 a_t^2 \theta_t R^2}{h_t^2} \leq -\frac{c}{h_t}, \qquad \tfrac{1}{2} < c < 1,$$

under a suitable choice of $\theta_t = \exp(-o(h_t^{-1}))$. The negative-curvature condition in the ridge definition is then satisfied.

Proof of Lemma D.5. Let $x \in B_t^{(i)}(\theta_t)$ and abbreviate the posterior weights by γ_j. By Lemma D.7,

$$\|\nabla m(t,x)\| = \frac{1}{h_t} \|\Sigma(t,x)\| \leq \frac{1}{h_t} \sum_{j=1}^n \gamma_j \|a_t x_0^{(j)} - m(t,x)\|^2.$$

For the dominant center,

$$\|a_t x_0^{(i)} - m(t,x)\| = \left\| a_t x_0^{(i)} - \sum_j \gamma_j a_t x_0^{(j)} \right\| \\ \leq \sum_{j \neq i} \gamma_j a_t \|x_0^{(j)} - x_0^{(i)}\| \leq 2a_t\theta_t R.$$

For $j \neq i$,

$$\|a_t x_0^{(j)} - m(t,x)\| \leq \|a_t x_0^{(i)} - m(t,x)\| + a_t\|x_0^{(j)} - x_0^{(i)}\| \leq 2a_t(\theta_t + 1)R \leq 4a_t R.$$

Combining the estimates,

$$\|\nabla m(t,x)\| \leq \frac{1}{h_t}\big[\gamma_i(2a_t\theta_t R)^2 + \sum_{j\neq i}\gamma_j(4a_t R)^2\big] \leq \frac{20a_t^2\theta_t R^2}{h_t}.$$

Similarly,

$$\|\nabla^3 \log p_t(x)\| \leq \frac{1}{h_t^3}\sum_j \gamma_j \|a_t x_0^{(j)} - m(t,x)\|^3 \\ \leq \frac{1}{h_t^3}\big[\gamma_i(2a_t\theta_t R)^3 + \sum_{j\neq i}\gamma_j(4a_t R)^3\big] \leq \frac{80a_t^3\theta_t R^3}{h_t^3}.$$

□

Proof of Proposition D.1. The submitted argument begins with Lemma D.8:

$$\sup_{z\in\mathcal{R}_t}\|\mathrm{II}_{t,z}\| \leq \frac{1}{\beta_t}\sup_{z\in\mathcal{R}_t}\|\nabla^3 \log p_t(z)\|.$$

It then uses the reach comparison

$$r_t \gtrsim \beta_t \left(\sup_{z\in\mathcal{R}_t}\|\nabla^3 \log p_t(z)\|\right)^{-1}.$$

In the center-dominant regime, inserting the estimate from Lemma D.5 gives

$$r_t \gtrsim \beta_t \left(\sup_{x\in\bigcup_i B_t^{(i)}(\theta_t)}\|\nabla^3 \log p_t(x)\|\right)^{-1} = \Omega\left(\frac{\beta_t h_t^3}{\theta_t R^3}\right).$$

For V_t, the normal-gauge calculation of Lemmas D.9–D.10 gives the expression

$$v_t(z) = -\frac{1}{h_t}E_t(z)\big(E_t(z)^\top \nabla^2 \log p_t(z) E_t(z)\big)^{-1} E_t(z)^\top \partial_t m(t,z).$$

Consequently,

$$V_t \leq \frac{1}{h_t}\sup_{z\in\mathcal{R}_t}\left\|\big(E_t^\top \nabla^2 \log p_t E_t\big)^{-1}\right\| \|E_t^\top \partial_t m(t,z)\| \leq \frac{1}{\beta_t h_t}\sup_{z\in\mathcal{R}_t}\|E_t^\top \partial_t m(t,z)\|.$$

Lemma D.11 supplies the stated order $V_t = O(R/(h_t\beta_t))$.

Finally, $P^\perp(z)v_t(z) = v_t(z)$. Differentiating along a tangent vector u yields

$$(\nabla_u P^\perp)v_t + P^\perp \nabla_u v_t = \nabla_u v_t, \qquad (\nabla_u P^\perp)v_t = P^\parallel \nabla_u v_t.$$

Hence $W_t \leq (\sup\|\nabla_u P^\perp\|)V_t$. Since $P^\parallel + P^\perp = I_d$, $\|\nabla_u P^\parallel\| = \|\nabla_u P^\perp\|$. Differentiating the projector identity and multiplying by the tangent or normal projectors on both sides gives

$$P^\parallel(\nabla_u P^\parallel)P^\parallel = 0, \qquad P^\perp(\nabla_u P^\parallel)P^\perp = 0.$$

On $T_z(\mathcal{R}_t)\oplus N_z(\mathcal{R}_t)$ we may therefore write

$$\nabla_u P^\parallel(z) = \begin{pmatrix} 0 & B^\top \\ B & 0 \end{pmatrix}, \qquad B = P^\perp(\nabla_u P^\parallel)P^\parallel.$$

Using Aamari et al. [1, Theorem 3.4],

$$\|\nabla_u P^\perp(z)\| = \|B\| = \|\mathrm{II}_{t,z}(u,\cdot)\|_{T_z(\mathcal{R}_t)\to N_z(\mathcal{R}_t)} \lesssim r_t^{-1}.$$

Thus $W_t \lesssim V_t/r_t$, giving the final estimate. □

Lemma D.7 (Explicit formulas)**.** *Under Assumption 3.1,*

$$\nabla \log p_t(x) = -\frac{x}{h_t} + \frac{m(t,x)}{h_t}, \qquad \nabla^2 \log p_t(x) = -\frac{I_d}{h_t} + \frac{\Sigma(t,x)}{h_t^2},$$
$$\nabla^3 \log p_t(x) = \frac{\mathbb{E}[(U(t,x) - m(t,x))^{\otimes 3}]}{h_t^3}, \qquad \nabla m(t,x) = \frac{\Sigma(t,x)}{h_t}.$$

Here $U(t,x)$ takes the values $a_t x_0^{(i)}$ with probabilities $\gamma_i(t,x) = \mathrm{Softmax}(-\|x - a_t x_0^{(j)}\|^2/(2h_t))_i$, and $m(t,x) = \mathbb{E}U(t,x)$, $\Sigma(t,x) = \mathrm{Cov}(U(t,x))$.

Lemma D.8 (Second fundamental form bound)**.** *For $t \in [\delta, T]$ and $z \in \mathcal{R}_t$,*

$$\|\mathrm{II}_{t,z}\| \le \frac{1}{\beta_t} \|\nabla^3 \log p_t(z)\|.$$

Proof. Take orthonormal bases $T \in \mathbb{R}^{d \times d^*}$ and $N \in \mathbb{R}^{d \times (d-d^*)}$ of the tangent and normal spaces at z. In coordinates $x(u,v) = z + Tu + Nv$, define

$$F(u,v) := N^\top \nabla \log p_t(x(u,v)).$$

The submitted local calculation uses $F(0,0) = 0$ and

$$\partial_v F(u,v) = N^\top \nabla^2 \log p_t(x(u,v)) N, \qquad \|(\partial_v F(0,0))^{-1}\| \le \beta_t^{-1}.$$

The implicit function theorem then gives $v = \phi(u)$, with $\phi(0) = 0$, and the local parametrization $\gamma(u) = z + Tu + N\phi(u)$. Differentiating $F(u, \phi(u)) = 0$ at zero gives $\nabla\phi(0) = 0$. Differentiating twice gives

$$\partial_{uu} F(0,0) + \partial_v F(0,0) \nabla^2 \phi(0) = 0.$$

For $\xi \in \mathbb{R}^{d^*}$,

$$\partial_{uu} F(0,0)[\xi, \xi] = N^\top \nabla^3 \log p_t(z)[T\xi, T\xi],$$

so

$$\|\nabla^2 \phi(0)\| \le \|(\partial_v F(0,0))^{-1}\| \|\partial_{uu} F(0,0)\| \le \beta_t^{-1} \|\nabla^3 \log p_t(z)\|.$$

Since $\partial_{ij}^2 \gamma(0) = N \partial_{ij}^2 \phi(0)$,

$$\mathrm{II}_{t,z}(\nabla\gamma(0)\xi, \nabla\gamma(0)\vartheta) = P^\perp(z) \nabla^2 \gamma(0)[\xi, \vartheta] = N \nabla^2 \phi(0)[\xi, \vartheta].$$

The stated bound follows from the orthonormality of N. □

Lemma D.9 (Normal velocity field expression)**.** *Let $E_t(z) \in \mathbb{R}^{d \times (d-d^*)}$ be the normal eigen-matrix from Definition 3.2. The normal-gauge expression used in the submitted argument is*

$$v_t(z) = -E_t(z) \big(E_t(z)^\top \nabla^2 \log p_t(z) E_t(z)\big)^{-1} E_t(z)^\top \partial_t \nabla \log p_t(z). \tag{17}$$

Proof. Fix $t_0 \in (\delta, T)$ and $z_0 \in \mathcal{R}_{t_0}$. In the normal-gauge calculation define $F(t,x) = E_{t_0}(z_0)^\top \nabla \log p_t(x)$. At (t_0, z_0),

$$\partial_t F = E_{t_0}(z_0)^\top \partial_t \nabla \log p_t(z_0)\big|_{t=t_0}, \qquad \nabla F = E_{t_0}(z_0)^\top \nabla^2 \log p_{t_0}(z_0).$$

Take a normal-gauge curve $z(t)$ with $z(t_0) = z_0$ and write $\partial_t z(t)|_{t=t_0} = E_{t_0}(z_0)\vartheta_0$. Differentiating the local constraint used in the argument gives

$$\begin{aligned} 0 &= \partial_t F(t_0, z_0) + \nabla F(t_0, z_0) \partial_t z(t)|_{t=t_0} \\ &= E_{t_0}(z_0)^\top \partial_t \nabla \log p_t(z_0)|_{t=t_0} + E_{t_0}(z_0)^\top \nabla^2 \log p_{t_0}(z_0) E_{t_0}(z_0) \vartheta_0. \end{aligned}$$

The normal Hessian eigenvalues are at most $-\beta_{t_0}$, so its inverse norm is at most $1/\beta_{t_0}$. Solving for ϑ_0 and multiplying by $E_{t_0}(z_0)$ gives (17). □

Lemma D.10 (Expression of ridge motion)**.** *For $z \in \mathcal{R}_t$ and the normal eigen-matrix $E_t(z)$,*

$$E_t(z)^\top \partial_t \nabla \log p_t(z) = \frac{1}{h_t} E_t(z)^\top \partial_t m(t,z).$$

Proof. Tweedie's formula gives $\nabla \log p_t(x) = (m(t,x) - x)/h_t$. Differentiating first and then projecting gives

$$E_t(x)^\top \partial_t \nabla \log p_t(x) = \frac{1}{h_t} E_t(x)^\top \partial_t m(t,x) - \frac{\partial_t h_t}{h_t^2} E_t(x)^\top (m(t,x) - x). \tag{18}$$

For $x = z \in \mathcal{R}_t$, the last term vanishes by the ridge condition. □

Lemma D.11. *For $z \in \mathcal{R}_t$ and the eigen-matrix $E_t(z)$ from Definition 3.2,*

$$\|E_t(z)^\top \partial_t m(t,z)\| \lesssim (1 + \dot{h}_t) a_t R.$$

Proof. The submitted proof differentiates the posterior-mean identity and uses the forward Fokker–Planck equation to expand $\partial_t m$. In its notation the resulting displayed calculation is

$$\begin{aligned}
\partial_t m(t,z) &= \partial_t \big(h_t \nabla \log p_t(z) + h_t z\big) \\
&= \dot{h}_t \nabla \log p_t(z) + h_t \nabla \left(\frac{\partial_t p_t(z)}{p_t(z)} \right) + \dot{h}_t z \\
&= \dot{h}_t \nabla \log p_t(z) + h_t \nabla \left(\frac{z \cdot \nabla p_t(z) + \Delta p_t(z) + d p_t(z)}{p_t(z)} \right) + \dot{h}_t z \\
&= (\dot{h}_t + d h_t) \nabla \log p_t(z) + h_t \nabla^2 \log p_t(z) z + \dot{h}_t z \\
&\quad + h_t \big(\mathrm{cont}_{23}\, \nabla^3 \log p_t(z) + 2 \nabla^2 \log p_t(z) \nabla \log p_t(z) \big).
\end{aligned}$$

At a spectral ridge point, $E_t^\top \nabla \log p_t = 0$ and $E_t^\top \nabla^2 \log p_t \nabla \log p_t = 0$. The projected expression used in the proof is

$$E_t^\top \partial_t m = h_t E_t^\top \nabla^2 \log p_t(z) z + \dot{h}_t E_t^\top z + h_t E_t^\top \,\mathrm{cont}_{23}\, \nabla^3 \log p_t(z).$$

The first two terms are estimated using

$$\|E_t^\top z\| = \|E_t^\top m(t,z)\| \le a_t R, \qquad \|E_t^\top \nabla^2 \log p_t(z) z\| \le h_t^{-1} \|E_t^\top z\| \le \frac{a_t R}{h_t}.$$

The Hessian bound follows because its negative eigenvalues lie between $-1/h_t$ and $-\beta_t$. For the remaining contraction, Lemma D.5 gives

$$\|E_t^\top \,\mathrm{cont}_{23}\, \nabla^3 \log p_t(z)\| \le \sqrt{d} \|\nabla^3 \log p_t(z)\| = O\left(\frac{\sqrt{d}\, \theta_t R^3}{h_t^3} \right).$$

The proof chooses $\theta_t = \exp(-o(h_t^{-1}))$ so this term is negligible in the stated order estimate. □

E Analysis of Tube Entry

We analyze entry into a tube neighborhood of the log-density ridge. The formal version of Theorem 3.5 uses the exact reverse OU process as a reference:

$$dX_t^{\leftarrow} = \left[\left(1 - \frac{2}{h_{T-t}} \right) X_t^{\leftarrow} + \frac{2}{h_{T-t}} m(T-t, X_t^{\leftarrow}) \right] dt + \sqrt{2}\, dB_t^{\leftarrow}, \qquad X_0^{\leftarrow} \sim p_T.$$

Then $X_t^{\leftarrow} \sim p_{T-t}$ for $0 \le t \le T - \delta$. Define

$$\begin{aligned}
t_{\mathrm{in}}^{\leftarrow} &:= \inf\{0 \le t \le T - \delta : X_t^{\leftarrow} \in \mathcal{T}_{T-t}(\rho_{T-t})\}, \\
t_{\mathrm{in}} &:= \inf\{0 \le t \le T - \delta : Y_t \in \mathcal{T}_{T-t}(\rho_{T-t})\}.
\end{aligned}$$

The learned-process entry time $\widetilde{t}_{\mathrm{in}}$ is defined in Section 3.2.

Theorem E.1. *Under Assumption 3.1 and the geometric inputs used below,*

$$\begin{aligned}
&\mathbb{P}(t_{\mathrm{in}}^{\leftarrow} \le T - \delta) \ge 1 - e_\delta, \\
&\mathbb{P}(t_{\mathrm{in}} \le T - \delta) \ge 1 - e_\delta - \varepsilon(T), \\
&\mathbb{P}(\widetilde{t}_{\mathrm{in}} \le T - \delta) \ge 1 - e_\delta - \varepsilon(T) - \sqrt{\varepsilon_A(T,\delta)/2},
\end{aligned}$$

where $e_\delta = h_\delta^\zeta$ for any fixed $\zeta > 0$,

$$\varepsilon(T) = \frac{a_T}{2}\left(\frac{\sqrt{d}}{\sqrt{h_T}} + R\right), \qquad \varepsilon_A(T,\delta) = \int_\delta^T h_t^{-2}\mathbb{E}\|m_A(t,X_t) - m(t,X_t)\|^2\,dt.$$

Thus $e_\delta \to 0$ as $\delta \to 0^+$ and $\varepsilon(T) \to 0$ exponentially as $T \to \infty$.

Proof. For the exact process, $X^{\leftarrow}_{T-\delta} \sim p_\delta$ and $X_\delta = a_\delta X_0 + \sqrt{h_\delta}z$, with $z \sim \mathcal{N}(0, I_d)$. Therefore

$$\begin{aligned}\mathbb{P}(X^{\leftarrow}_{T-\delta} \in \mathcal{T}_\delta(\rho_\delta)) &= \mathbb{P}(\operatorname{dist}(X_\delta, \mathcal{R}_\delta) \le \rho_\delta) \\ &\ge \mathbb{P}\big(\operatorname{dist}(a_\delta X_0, \mathcal{R}_\delta) + \sqrt{h_\delta}\|z\| \le \rho_\delta\big).\end{aligned}$$

For sufficiently small δ, Lemma E.2 gives $\operatorname{dist}(a_\delta X_0, \mathcal{R}_\delta) \le \rho_\delta/2$. Hence

$$\mathbb{P}(X^{\leftarrow}_{T-\delta} \in \mathcal{T}_\delta(\rho_\delta)) \ge \mathbb{P}\left(\sqrt{h_\delta}\|z\| \le \frac{\rho_\delta}{2}\right) = \mathbb{P}\left(\|z\|^2 \le \frac{\rho_\delta^2}{4h_\delta}\right).$$

Choose an admissible tube radius such that

$$\frac{\rho_\delta^2}{4h_\delta} \ge q_\delta, \qquad q_\delta := d + 2\sqrt{d\zeta\log(1/h_\delta)} + 2\zeta\log(1/h_\delta).$$

The admissibility of this choice is the geometric input from Appendix D. The Gaussian concentration bound [28] then gives

$$\mathbb{P}(X^{\leftarrow}_{T-\delta} \in \mathcal{T}_\delta(\rho_\delta)) \ge \mathbb{P}(\|z\|^2 \le q_\delta) \ge 1 - h_\delta^\zeta.$$

Next, Y_t and $X_t^{\leftarrow}$ have the same transition kernel but different initializations, $Y_0 \sim \mathcal{N}(0, I_d)$ and $X_0^{\leftarrow} \sim p_T$. By data processing, Pinsker's inequality, and the OU initialization estimate [19, Proposition C.1],

$$\begin{aligned}&\left|\mathbb{P}(Y_{T-\delta} \in \mathcal{T}_\delta(\rho_\delta)) - \mathbb{P}(X^{\leftarrow}_{T-\delta} \in \mathcal{T}_\delta(\rho_\delta))\right| \\ &\quad\le \operatorname{TV}(\operatorname{Law}(Y_{T-\delta}), p_\delta) \le \operatorname{TV}(\mathcal{N}(0, I_d), p_T) \\ &\quad\le \sqrt{\tfrac{1}{2}\operatorname{KL}(p_T \mid \mathcal{N}(0, I_d))} \le \frac{a_T}{2}\left(\frac{\sqrt{d}}{\sqrt{h_T}} + R\right) = \varepsilon(T).\end{aligned}$$

For the learned process, let $\mathbb{P}^{\leftarrow}$ and $\widetilde{\mathbb{P}}$ be the exact and learned path laws. The drift difference is $2e_A/h_t$ and the diffusion coefficient is $\sqrt{2}I_d$. Under the change-of-measure conditions,

$$\begin{aligned}\operatorname{KL}(\mathbb{P}^{\leftarrow} \mid \widetilde{\mathbb{P}}) &= \operatorname{KL}(p_T \mid \mathcal{N}(0, I_d)) + \frac{1}{2}\int_0^{T-\delta} \mathbb{E}\left\|\frac{\sqrt{2}\,e_A(T-u, X_u^{\leftarrow})}{h_{T-u}}\right\|^2 du \\ &= \operatorname{KL}(p_T \mid \mathcal{N}(0, I_d)) + \varepsilon_A(T,\delta).\end{aligned}$$

Data processing and Pinsker's inequality give

$$\begin{aligned}&\left|\mathbb{P}(\widetilde{Y}_{T-\delta} \in \mathcal{T}_\delta(\rho_\delta)) - \mathbb{P}(X^{\leftarrow}_{T-\delta} \in \mathcal{T}_\delta(\rho_\delta))\right| \\ &\quad\le \sqrt{\tfrac{1}{2}\operatorname{KL}(\mathbb{P}^{\leftarrow} \mid \widetilde{\mathbb{P}})} \le \varepsilon(T) + \sqrt{\varepsilon_A(T,\delta)/2}.\end{aligned}$$

Finally, terminal tube occupancy implies first entry by the terminal time. Applying this inclusion to each process proves the stated bounds. □

For fixed positive δ, terminal occupancy of a proper finite-radius tube need not have probability one. The theorem gives a conservative lower bound for first entry, not the exact failure probability, and does not quantify the separation between entry and termination.

Lemma E.2. *Under Assumption 3.1, for sufficiently small δ and each $i \in [n]$, there is a point $z_i \in \mathcal{R}_\delta$ satisfying*

$$\|z_i - a_\delta x_0^{(i)}\| \le 2a_\delta R(n-1)\exp\left(-\frac{a_\delta^2\Delta^2}{2h_\delta}\right).$$

Proof. By Lemma D.7, with $\gamma_j(\delta, x)$ denoting the empirical posterior weights,

$$\nabla \log p_\delta(x) = -\frac{x}{h_\delta} + \frac{1}{h_\delta}\sum_j \gamma_j(\delta, x) a_\delta x_0^{(j)},$$

$$\nabla^2 \log p_\delta(x) = -\frac{I_d}{h_\delta} + \frac{1}{h_\delta^2}\sum_j \gamma_j(\delta, x)(a_\delta x_0^{(j)} - m(\delta, x))(a_\delta x_0^{(j)} - m(\delta, x))^\top.$$

Critical points are fixed points of

$$x = \sum_j \gamma_j(\delta, x) a_\delta x_0^{(j)} = m(\delta, x).$$

The proof applies the contraction mapping theorem on $B(a_\delta x_0^{(i)}, a_\delta \Delta/4)$. The posterior-ratio estimate used in the submission is

$$\frac{\gamma_j(\delta, x)}{\gamma_i(\delta, x)} = \exp\left(-\frac{\|x - a_\delta x_0^{(j)}\|^2 - \|x - a_\delta x_0^{(i)}\|^2}{2h_\delta}\right) \le \exp\left(-\frac{a_\delta^2 \Delta^2}{2h_\delta}\right), \quad j \ne i.$$

It yields

$$\sum_{j \ne i} \gamma_j(\delta, x) \le (n-1) e^{-a_\delta^2 \Delta^2/(2h_\delta)},$$

and thus

$$\|m(\delta, x) - a_\delta x_0^{(i)}\| \le \sum_{j \ne i} \gamma_j(\delta, x) a_\delta \|x_0^{(j)} - x_0^{(i)}\| \le 2a_\delta R(n-1) e^{-a_\delta^2 \Delta^2/(2h_\delta)} < a_\delta \Delta/4$$

for sufficiently small δ. The corresponding derivative estimate in the argument is

$$\begin{aligned}\|\nabla m(\delta, x)\| &\le \frac{1}{h_\delta}\sum_{j \ne i} \gamma_j(\delta, x) a_\delta^2 \|x_0^{(j)} - x_0^{(i)}\|^2 \\ &\le \frac{a_\delta^2 R^2 (n-1)}{h_\delta} e^{-a_\delta^2 \Delta^2/(2h_\delta)} < 1.\end{aligned}$$

Thus the mapping has a unique fixed point z_i in the ball. At the fixed point the Hessian estimate is

$$\nabla^2 \log p_\delta(z_i) \preceq -\frac{I_d}{h_\delta} + \frac{a_\delta^2 R^2 (n-1)}{h_\delta^2} e^{-a_\delta^2 \Delta^2/(2h_\delta)} I_d \preceq -\frac{I_d}{2h_\delta}$$

when δ is sufficiently small. Finally,

$$\|z_i - a_\delta x_0^{(i)}\| = \|m(\delta, z_i) - a_\delta x_0^{(i)}\| \le \sum_{j \ne i} \gamma_j(\delta, z_i) a_\delta \|x_0^{(j)} - x_0^{(i)}\| \le 2a_\delta R(n-1) e^{-a_\delta^2 \Delta^2/(2h_\delta)}.$$

□

F Analysis of Post-entry Normal Alignment

We give the time-resolved normal-displacement estimate underlying Theorem 3.6. As in the main text, the analysis is local to the regular projection tube; it does not assert that first entry alone guarantees subsequent residence.

Theorem F.1. *Under Assumption 3.1, in the regular post-entry regime, define*

$$\kappa_{s,t} = 2\int_s^t (\beta_{T-u} - 1)\, du, \qquad B_{s,t}(d, R) = \int_s^t e^{-\kappa_{u,t}} (\rho_{T-u} R + d)\, du,$$

and $e_A^\perp(t, x) = P_t^\perp(\Pi_t(x)) e_A(t, x)$ *with* $e_A = m_A - m$. *For* $t \in [\widetilde{t}_{\mathrm{in}}, T - \delta]$,

$$\mathbb{E} D_{T-t}(\widetilde{Y}_t) \lesssim e^{-\kappa_{\widetilde{t}_{\mathrm{in}}, t}} \rho_{T-\widetilde{t}_{\mathrm{in}}}^2 + B_{\widetilde{t}_{\mathrm{in}}, t}(d, R) + \int_{\widetilde{t}_{\mathrm{in}}}^t e^{-\kappa_{u,t}} \frac{\mathbb{E}\|e_A^\perp(T-u, \widetilde{Y}_u)\|^2}{h_{T-u}^2 \beta_{T-u}}\, du.$$

Choosing $\beta_t = c/h_t$, $c \in [1/2, 1)$, *the terminal estimate is*

$$\mathbb{E} D_\delta(\widetilde{Y}_{T-\delta}) = O\left(d\delta^c + \delta^c \int_{\widetilde{t}_{\mathrm{in}}}^{T-\delta} h_{T-u}^{-1-c} \mathbb{E}\|e_A^\perp(T-u, \widetilde{Y}_u)\|^2\, du\right).$$

Corollary F.2. *The squared-distance function $D_t(x) = \|x - \Pi_t(x)\|^2$ satisfies*

$$D_t \in C^2(\mathcal{T}_t(\rho_t)), \qquad \nabla D_t(x) = 2n_t(x), \qquad \sup_x \|\nabla^2 D_t(x)\| \le \frac{2}{1 - \rho_t/r_t}.$$

In particular, $\sup_x \Delta D_t(x) \le 2d/(1 - \rho_t/r_t)$.

Write the exact-score and learned drifts as

$$b(T-t,x) = x - \frac{2}{h_{T-t}}x + \frac{2}{h_{T-t}}m(T-t,x),$$
$$b_A(T-t,x) = x - \frac{2}{h_{T-t}}x + \frac{2}{h_{T-t}}m_A(T-t,x).$$

Both processes start from $\mathcal{N}(0, I_d)$ and have diffusion coefficient $\sqrt{2}I_d$. The exact-score process differs from the exact reverse OU process only through initialization; the learned process also has posterior-mean approximation error.

Proof of Theorem F.1. Inside the regular tube, Itô's formula gives

$$dD_{T-t}(Y_t) = \big[\partial_t D_{T-t}(Y_t) + 2\langle n_{T-t}(Y_t), b(T-t,Y_t)\rangle + \mathrm{tr}(\nabla^2 D_{T-t}(Y_t))\big]dt + 2\sqrt{2}\,\langle n_{T-t}(Y_t), d\bar{B}_t\rangle.$$

With $\rho_t = r_t/2$, Corollary F.2 bounds the trace term by $4d$, and hence also by the $8d$ upper bound used below. For the moving ridge,

$$\partial_t D_{T-t}(Y_t) = 2\langle n_{T-t}(Y_t), \partial_s \Pi_s(Y_t)|_{s=T-t}\rangle \le 2\rho_{T-t}S_{T-t}, \quad S_s := \sup_{x\in\mathcal{T}_s(\rho_s)} \|\partial_s \Pi_s(x)\|.$$

By Remark D.2, $S_s = O(R)$ in the late-time regime.

Let $z = \Pi_{T-t}(Y_t)$ and $n = n_{T-t}(Y_t)$. The score contribution is expanded as

$$\langle n, \nabla \log p_{T-t}(Y_t)\rangle = \langle n, \nabla \log p_{T-t}(z)\rangle + \langle n, \nabla^2 \log p_{T-t}(z) n\rangle + \frac{1}{2}\langle n, \nabla^3 \log p_{T-t}(z') n^{\otimes 2}\rangle,$$

where z' lies between z and Y_t. The local ridge estimates used in the argument are

$$\langle n, \nabla \log p_{T-t}(z)\rangle = 0,$$
$$\langle n, P^\perp(z)\nabla^2 \log p_{T-t}(z) n\rangle \le -\beta_{T-t}\|n\|^2,$$
$$\big|\langle n, \nabla^3 \log p_{T-t}(z') n^{\otimes 2}\rangle\big| \le \sup_x \|\nabla^3 \log p_{T-t}(x)\|\|n\|^3 \le \frac{a_{T-t}^3 \beta_{T-t}}{2} D_{T-t}(Y_t).$$

The last estimate uses the center-dominant derivative control and a tube radius small enough for the Taylor remainder. Since $a_{T-t} \le 1$, these imply

$$\langle n, \nabla \log p_{T-t}(Y_t)\rangle \le -\frac{3}{4}\beta_{T-t} D_{T-t}(Y_t).$$

The submitted drift estimate then yields, after absorbing the geometric remainder,

$$\frac{d}{dt}\mathbb{E}D_{T-t}(Y_t) \le -(3\beta_{T-t} - 2)\mathbb{E}D_{T-t}(Y_t) + C\rho_{T-t}R + 8d.$$

Gronwall's inequality gives, for the interval on which the local estimate holds,

$$\mathbb{E}D_{T-t}(Y_t) \le e^{-\int_{t_{\mathrm{in}}}^t (3\beta_{T-u}-2)du}\mathbb{E}D_{T-t_{\mathrm{in}}}(Y_{t_{\mathrm{in}}}) + \int_{t_{\mathrm{in}}}^t e^{-\int_u^t (3\beta_{T-s}-2)ds}(C\rho_{T-u}R + 8d)\,du.$$

For the learned process, $b_A - b = 2e_A/h_{T-t}$ under the convention $e_A = m_A - m$. The additional term in the squared-distance drift is $4h_{T-t}^{-1}\mathbb{E}\langle n_{T-t}(\widetilde{Y}_t), e_A^\perp(T-t,\widetilde{Y}_t)\rangle$. Young's inequality bounds it by

$$\beta_{T-t}\mathbb{E}D_{T-t}(\widetilde{Y}_t) + \frac{4\mathbb{E}\|e_A^\perp(T-t,\widetilde{Y}_t)\|^2}{h_{T-t}^2\beta_{T-t}}.$$

Thus

$$\frac{d}{dt}\mathbb{E}D_{T-t}(\widetilde{Y}_t) \le -2(\beta_{T-t}-1)\mathbb{E}D_{T-t}(\widetilde{Y}_t) + C(\rho_{T-t}R+d) + \frac{C\mathbb{E}\|e_A^{\perp}(T-t,\widetilde{Y}_t)\|^2}{h_{T-t}^2\beta_{T-t}}.$$

Integrating this differential inequality gives the time-resolved bound in the theorem. At $t = T-\delta$, with $\beta_s = c/h_s$, the integrating-factor calculation used in the terminal estimate gives

$$\int_{\widetilde{t}_{\mathrm{in}}}^{T-\delta} e^{-2\int_u^{T-\delta}(\beta_{T-s}-1)ds}\,du = O(\delta^c).$$

The weighted error integral satisfies

$$\begin{aligned}\int_{\widetilde{t}_{\mathrm{in}}}^{T-\delta} & e^{-2\int_u^{T-\delta}(\beta_{T-s}-1)ds}\frac{\mathbb{E}\|e_A^{\perp}(T-u,\widetilde{Y}_u)\|^2}{h_{T-u}^2\beta_{T-u}}\,du \\ &= O\left(\delta^c\int_{\widetilde{t}_{\mathrm{in}}}^{T-\delta} h_{T-u}^{-1-c}\mathbb{E}\|e_A^{\perp}(T-u,\widetilde{Y}_u)\|^2\,du\right).\end{aligned}$$

Combining with the geometric and noise terms yields the stated terminal estimate. □

G Analysis of Post-entry Tangential Sliding

We analyze the tangential sliding behavior of the ideal reverse process Y_t and learned reverse process $\widetilde{Y}_t$. The time-resolved estimate below controls the squared center-relative tangential residual. The terminal estimate in Theorem 3.7 follows by integration. Both normal alignment and tangential sliding can be active on the same post-entry interval.

Theorem G.1. *Under Assumption 3.1, in a regular center-dominant chart, define* $e_A^{\parallel,i}(s,x) = (U_s^{(i)})^\top e_A(s,x)$ *with* $e_A = m_A - m$. *While the local chart assumptions hold,*

$$\frac{d}{dt}\mathbb{E}\|\widetilde{u}_t^{(i)}\|^2 \le -\left(\frac{1}{h_{T-t}}-2\right)\mathbb{E}\|\widetilde{u}_t^{(i)}\|^2 + \widetilde{\epsilon}_{T-t}^{(i)} + \frac{4}{h_{T-t}}\mathbb{E}\|e_A^{\parallel,i}(T-t,\widetilde{Y}_t)\|^2,$$

where $\widetilde{\epsilon}_{T-t}^{(i)} = O(d)$ *in the late-time regime. For* $\delta \ll 1$, *the corresponding terminal estimate is*

$$\mathbb{E}\|\widetilde{u}_{T-\delta}^{(i)}\|^2 = O\left(d\sqrt{\delta} + \sqrt{\delta}\int_{\widetilde{t}_{\mathrm{in}}}^{T-\delta} h_{T-u}^{-3/2}\mathbb{E}\|e_A^{\parallel,i}(T-u,\widetilde{Y}_u)\|^2\,du\right).$$

Recall the drifts b, b_A from Appendix F. The frame $U_{T-t}^{(i)} \in \mathbb{R}^{d\times d^*}$ contains the top d^* eigenvectors of the Hessian $\nabla^2 \log p_{T-t}(m_{T-t}^{(i)})$. Define

$$r_t^{(i)} = Y_t - m_{T-t}^{(i)}, \quad \widetilde{r}_t^{(i)} = \widetilde{Y}_t - m_{T-t}^{(i)}, \quad u_t^{(i)} = (U_{T-t}^{(i)})^\top r_t^{(i)}, \quad \widetilde{u}_t^{(i)} = (U_{T-t}^{(i)})^\top \widetilde{r}_t^{(i)}.$$

The center-based frame is a local proxy for the corresponding ridge frame. Lemma E.2 places the center near a ridge point at small noise; its use as a tangent proxy is restricted to the regular chart. Sliding concerns the coordinate relative to a nearby center, not the full occupation law along the ridge.

Proof of Theorem G.1 and Theorem 3.7. For notational clarity in the following local calculation, set $s = T-t$, $U = U_s^{(i)}$, $m_i = m_s^{(i)}$, and $u = u_t^{(i)}$. Since $\partial_t m_{T-t}^{(i)} = m_{T-t}^{(i)}$, Itô's formula yields

$$\begin{aligned}du &= (\partial_t U^\top)(Y_t - m_i)\,dt + U^\top\big(b(s,Y_t) - m_i\big)\,dt + \sqrt{2}\,U^\top d\bar{B}_t \\ &= -\left(\frac{2}{h_s}-1\right)u\,dt + (\partial_t U^\top)(Y_t-m_i)\,dt + \sqrt{2}\,U^\top d\bar{B}_t \\ &\quad + U^\top\left[b(s,Y_t) - \left(Y_t + \frac{2}{h_s}(m_i - Y_t)\right)\right]dt.\end{aligned}$$

Applying Itô's formula to $\|u\|^2$ gives

$$d\|u\|^2 = -2\left(\frac{2}{h_s} - 1\right)\|u\|^2 dt + 2\langle u, (\partial_t U^\top)(Y_t - m_i)\rangle dt \\ + 2\langle u, U^\top\left[b(s, Y_t) - \left(Y_t + \frac{2}{h_s}(m_i - Y_t)\right)\right]\rangle dt + 2d^* dt + 2\sqrt{2}\,\langle u, U^\top d\bar{B}_t\rangle.$$

Let $P = UU^\top$. By Lemma G.4, the part of the frame derivative within the tangent space is skew-symmetric. Accordingly,

$$\langle u, (\partial_t U^\top)(Y_t - m_i)\rangle = \langle u, (\partial_t U^\top)Uu\rangle + \langle u, (\partial_t U^\top)(I - UU^\top)(Y_t - m_i)\rangle \\ \le \frac{1}{h_s}\|u\|^2 + h_s\|P(\partial_t P)(I - P)\|^2\|r_t^{(i)}\|^2.$$

Lemma G.3 and the local residual bounds used in the argument give

$$\left|\langle u, U^\top\left[b(s, Y_t) - \left(Y_t + \frac{2}{h_s}(m_i - Y_t)\right)\right]\rangle\right| \le 8R^2 a_s^2 h_s^{-1}\theta_s.$$

The local proof estimates $\|r_t^{(i)}\|^2 = O(R^2)$ and

$$\|P(\partial_t P)(I - P)\| = O\big(\theta_s \operatorname{poly}(R, h_s^{-1})\big),$$

so the frame, center-approximation, and noise contributions are collected into $\epsilon_s^{(i)} = O(d)$. Taking expectations yields

$$\frac{d}{dt}\mathbb{E}\|u_t^{(i)}\|^2 \le -2\left(\frac{1}{h_{T-t}} - 1\right)\mathbb{E}\|u_t^{(i)}\|^2 + \epsilon_{T-t}^{(i)}.$$

Gronwall's inequality gives

$$\mathbb{E}\|u_t^{(i)}\|^2 \le \exp\left[-2\int_{t_{\text{in}}}^t \left(\frac{1}{h_{T-u}} - 1\right) du\right]\mathbb{E}\|u_{t_{\text{in}}}^{(i)}\|^2 \\ + \int_{t_{\text{in}}}^t \exp\left[-2\int_u^t \left(\frac{1}{h_{T-v}} - 1\right) dv\right]\epsilon_{T-u}^{(i)}\, du.$$

At $t = T - \delta$, the accumulated remainder is $O(d\delta\log(1/\delta))$ in the stated regime. This describes the ideal-process center-relative contraction within the chart.

For the learned process, use the same decomposition with $\widetilde{r}_t^{(i)}$ and $\widetilde{u}_t^{(i)}$. The only learning contribution to the coordinate drift is $+2e_A^{\|,i}/h_{T-t}$. Thus

$$\frac{d}{dt}\mathbb{E}\|\widetilde{u}_t^{(i)}\|^2 \le -2\left(\frac{1}{h_{T-t}} - 1\right)\mathbb{E}\|\widetilde{u}_t^{(i)}\|^2 + \widetilde{\epsilon}_{T-t}^{(i)} \\ + \frac{4}{h_{T-t}}\mathbb{E}\langle\widetilde{u}_t^{(i)}, (U_{T-t}^{(i)})^\top e_A(T - t, \widetilde{Y}_t)\rangle.$$

Young's inequality gives

$$\frac{4}{h_{T-t}}\langle\widetilde{u}_t^{(i)}, e_A^{\|,i}\rangle \le \frac{1}{h_{T-t}}\|\widetilde{u}_t^{(i)}\|^2 + \frac{4}{h_{T-t}}\|e_A^{\|,i}\|^2,$$

which proves the differential bound. Writing $J_{u,t} = \int_u^t (h_{T-v}^{-1} - 2)\, dv$, its integrated form is

$$\mathbb{E}\|\widetilde{u}_t^{(i)}\|^2 \le e^{-J_{\widetilde{t}_{\text{in}},t}}\mathbb{E}\|\widetilde{u}_{\widetilde{t}_{\text{in}}}^{(i)}\|^2 + \int_{\widetilde{t}_{\text{in}}}^t e^{-J_{u,t}}\widetilde{\epsilon}_{T-u}^{(i)}\, du \\ + \int_{\widetilde{t}_{\text{in}}}^t e^{-J_{u,t}}\frac{4}{h_{T-u}}\mathbb{E}\|(U_{T-u}^{(i)})^\top e_A(T - u, \widetilde{Y}_u)\|^2\, du.$$

For $t = T - \delta$, the remainder integral is $O(d\sqrt{\delta})$, and the error integral is

$$O\left(\sqrt{\delta}\int_{\widetilde{t}_{\text{in}}}^{T-\delta} h_{T-u}^{-3/2}\mathbb{E}\|(U_{T-u}^{(i)})^\top e_A(T - u, \widetilde{Y}_u)\|^2\, du\right).$$

This gives the stated terminal estimate. □

Remark G.2 (Tangential distribution of the residual)**.** *The following is a local leading-order interpretation, rather than a theorem identifying the entire tangential distribution. With $e_A = m_A - m$, the learned drift is the exact drift plus $2e_A/h_{T-t}$. The leading tangent equation has the sign convention*

$$d\widetilde{u}_t^{(i)} = -\frac{2}{h_{T-t}}\left(\widetilde{u}_t^{(i)} - (U_{T-t}^{(i)})^\top e_A(T-t, \widetilde{Y}_t)\right)dt + \sqrt{2}\,(U_{T-t}^{(i)})^\top d\widetilde{B}_t + \textit{local remainder}.$$

The corresponding contribution to the squared residual is $+4h_{T-t}^{-1}\langle \widetilde{u}_t^{(i)}, (U_{T-t}^{(i)})^\top e_A\rangle$. Decompose

$$\widetilde{Y}_t = m_{T-t}^{(i)} + U_{T-t}^{(i)}\widetilde{u}_t^{(i)} + \big(I - U_{T-t}^{(i)}(U_{T-t}^{(i)})^\top\big)(\widetilde{Y}_t - m_{T-t}^{(i)}).$$

If the normal component and the local remainder can be neglected, then

$$d\widetilde{u}_t^{(i)} \approx -\frac{2}{h_{T-t}}\Big[\widetilde{u}_t^{(i)} - (U_{T-t}^{(i)})^\top e_A\big(T-t, m_{T-t}^{(i)} + U_{T-t}^{(i)}\widetilde{u}_t^{(i)}\big)\Big]dt + \sqrt{2}\,d\widetilde{W}_t, \tag{19}$$

where $\widetilde{W}$ is the projected d^*-dimensional Brownian motion. This approximation is restricted to a regular positive-dimensional center-dominant chart; it should not be continued through a singularity or an eigengap closure.

Lemma G.3. *Under Assumption 3.1, if $x \in B_s^{(i)}(\theta_s)$, then*

$$\left\| b(s,x) - \left(x + \frac{2}{h_s}(m_s^{(i)} - x)\right)\right\| \le 4Ra_s h_s^{-1}\theta_s.$$

Proof. Using posterior weights,

$$m(s,x) - m_s^{(i)} = \sum_{j\neq i}\gamma_j(s,x)a_s(x_0^{(j)} - x_0^{(i)}).$$

Thus $\|m(s,x) - m_s^{(i)}\| \le 2Ra_s\theta_s$. Substituting in the drift expression gives

$$\left\| b(s,x) - \left(x + \frac{2}{h_s}(m_s^{(i)} - x)\right)\right\| = \frac{2}{h_s}\|m(s,x) - m_s^{(i)}\| \le 4Ra_s h_s^{-1}\theta_s.$$

□

Lemma G.4. *The local orthonormal frame $U_t^{(i)}$ satisfies $\langle u, (\partial_t U_t^{(i)})^\top U_t^{(i)} u\rangle = 0$ for all $u \in \mathbb{R}^{d^*}$.*

Proof. Differentiate $(U_t^{(i)})^\top U_t^{(i)} = I_{d^*}$ to obtain

$$(\partial_t U_t^{(i)})^\top U_t^{(i)} = -\big((\partial_t U_t^{(i)})^\top U_t^{(i)}\big)^\top.$$

The matrix is skew-symmetric, so its quadratic form vanishes. □

H From Inference to Training

This section converts the pathwise error terms in Theorems 3.6–3.7 into quantities evaluated along the exact reverse OU process $X_t^{\leftarrow}$. To separate the initialization error, introduce $\widehat{Y}_t$, which has the same transition kernel as $\widetilde{Y}_t$ but starts from p_T:

$$\begin{aligned}
dX_t^{\leftarrow} &= b(T-t, X_t^{\leftarrow})dt + \sqrt{2}\,dB_t^{\leftarrow}, & X_0^{\leftarrow} &\sim p_T,\\
d\widehat{Y}_t &= b_A(T-t, \widehat{Y}_t)dt + \sqrt{2}\,d\widetilde{B}_t, & \widehat{Y}_0 &\sim p_T,\\
d\widetilde{Y}_t &= b_A(T-t, \widetilde{Y}_t)dt + \sqrt{2}\,d\widetilde{B}_t, & \widetilde{Y}_0 &\sim \mathcal{N}(0, I_d).
\end{aligned}$$

Let $\mathbb{P}$ and $\mathbb{Q}$ denote the path laws of $X^{\leftarrow}$ and $\widehat{Y}$. Their initial laws agree, so Girsanov's theorem does not introduce an initial-density factor. The remaining difference between $\widetilde{Y}$ and $\widehat{Y}$ is controlled by the proximity of p_T to the standard Gaussian.

Assumption H.1. *Assume:*

(0) A sufficient Novikov condition holds:

$$\mathbb{E}_{\mathbb{P}} \exp\left(2 \int_0^{T-\delta} \frac{\|e_A(T-s, X_s^{\leftarrow})\|^2}{h_{T-s}^2} \, ds \right) < \infty.$$

(1) The path-law density ratios are bounded in χ^2: $\sup_{t\in[0,T-\delta]} \chi^2(\mathbb{Q}_t \mid \mathbb{P}_t) \le C_\chi^2 = O(1)$, *where $\mathbb{P}_t, \mathbb{Q}_t$ are the laws restricted to time t in the change-of-measure argument.*

(2) For $\dagger \in \{\perp, \|\}$,

$$\left(\mathbb{E}\|e_A^\dagger(T-t, X_t^{\leftarrow})\|^4\right)^{1/2} \le C_m \mathbb{E}\|e_A^\dagger(T-t, X_t^{\leftarrow})\|^2, \qquad C_m = O(1).$$

(3) The weighted mean error is uniformly bounded:

$$\sup_{t\in[0,T-\delta]} \sup_{x\in\mathcal{T}_{T-t}(\rho_{T-t})} \frac{w(T-t)\|e_A(T-t,x)\|^2}{h_{T-t}^2} \le \frac{C_u}{T-\delta},$$

with $C_u = O(1)$ and the decreasing weight $t \mapsto w(t)$ specified in the submitted assumption.

Theorem H.2. *Under Assumption H.1, the directional bounds are estimated by projected mean-matching losses:*

$$\textit{normal-error bound} \lesssim C_\delta^\perp \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E}\|e_A^\perp(t, X_t)\|^2 dt + C_\delta^\perp(\sqrt{d}+R)e^{-T} + d\delta^c, \tag{20}$$

$$\textit{tangent-error bound} \lesssim C_\delta^\| \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E}\|e_A^\|(t, X_t)\|^2 dt + C_\delta^\|(\sqrt{d}+R)e^{-T} + d\sqrt{\delta}, \tag{21}$$

where

$$C_\delta^\perp = \delta^c \left(1 \vee \frac{\delta^{1-c}}{w(\delta)}\right), \qquad C_\delta^\| = \sqrt{\delta}\left(1 \vee \frac{\sqrt{\delta}}{w(\delta)}\right), \qquad c = \lim_{t\to\delta^+} h_t\beta_t.$$

Proof. Define

$$L_t = \sqrt{2}\int_0^t \frac{e_A(T-s, X_s^{\leftarrow})}{h_{T-s}} \cdot dB_s^{\leftarrow}, \qquad \mathcal{E}(L)_t = \exp\left(L_t - \tfrac{1}{2}[L]_t\right).$$

Assumption H.1(0) makes this exponential a martingale. Under $\mathcal{E}(L)_{T-\delta}\mathbb{P}$,

$$B_t^{\leftarrow} - \sqrt{2}\int_0^t \frac{e_A(T-s, X_s^{\leftarrow})}{h_{T-s}} \, ds$$

is Brownian motion, and $\mathbb{Q} = \mathcal{E}(L)_{T-\delta}\mathbb{P}$.

For $\beta_s = c/h_s$, the normal accumulated-error term in Theorem 3.6 is bounded by

$$\begin{aligned}
&\int_{\widetilde{t}_{\text{in}}}^{T-\delta} e^{-2\int_u^{T-\delta}(\beta_{T-s}-1)ds} \frac{\mathbb{E}\|e_A^\perp(T-u, \widetilde{Y}_u)\|^2}{h_{T-u}^2\beta_{T-u}} \, du \\
&\quad\lesssim \delta^c \int_{\widetilde{t}_{\text{in}}}^{T-\delta} \frac{h_{T-t}^{1-c}}{w(T-t)} \frac{w(T-t)\mathbb{E}\|e_A^\perp(T-t, \widetilde{Y}_t)\|^2}{h_{T-t}^2} \, dt \\
&\quad\lesssim C_\delta^\perp \int_{\widetilde{t}_{\text{in}}}^{T-\delta} \frac{w(T-t)\mathbb{E}\|e_A^\perp(T-t, \widetilde{Y}_t)\|^2}{h_{T-t}^2} \, dt.
\end{aligned}$$

Enlarging the nonnegative time integral to $[0, T-\delta]$, split it into an initialization difference and a change-of-measure term:

$$\begin{aligned}
&\int_0^{T-\delta} \frac{w(T-t)}{h_{T-t}^2} \mathbb{E}\|e_A^\perp(T-t, \widetilde{Y}_t)\|^2 \, dt \\
&= \int_0^{T-\delta} \frac{w(T-t)}{h_{T-t}^2} \left[\mathbb{E}\|e_A^\perp(T-t, \widetilde{Y}_t)\|^2 - \mathbb{E}\|e_A^\perp(T-t, \widehat{Y}_t)\|^2\right] dt \\
&\quad + \int_0^{T-\delta} \frac{w(T-t)}{h_{T-t}^2} \mathbb{E}\left[\mathcal{E}(L)_t \|e_A^\perp(T-t, X_t^{\leftarrow})\|^2\right] dt.
\end{aligned}$$

By Cauchy–Schwarz,

$$\begin{aligned}\mathbb{E}\left[\mathcal{E}(L)_t\|e_A^{\perp}(T-t,X_t^{\leftarrow})\|^2\right] &= \mathbb{E}\|e_A^{\perp}(T-t,X_t^{\leftarrow})\|^2 \\ &\quad + \mathbb{E}\left[(\mathcal{E}(L)_t-1)\|e_A^{\perp}(T-t,X_t^{\leftarrow})\|^2\right] \\ &\le \mathbb{E}\|e_A^{\perp}(T-t,X_t^{\leftarrow})\|^2 \\ &\quad + \sqrt{\chi^2(\mathbb{Q}_t \mid \mathbb{P}_t)}\left(\mathbb{E}\|e_A^{\perp}(T-t,X_t^{\leftarrow})\|^4\right)^{1/2} \\ &\le (1+C_\chi C_m)\mathbb{E}\|e_A^{\perp}(T-t,X_t^{\leftarrow})\|^2.\end{aligned}$$

The bounded-observable initialization comparison used in the argument gives

$$\begin{aligned}&\left|\int_0^{T-\delta}\frac{w(T-t)}{h_{T-t}^2}\left[\mathbb{E}\|e_A^{\perp}(T-t,\widetilde{Y}_t)\|^2-\mathbb{E}\|e_A^{\perp}(T-t,\widehat{Y}_t)\|^2\right]dt\right| \\ &\qquad \le C_u\,\mathrm{TV}(\mathcal{N}(0,I_d),p_T) \lesssim (\sqrt{d}+R)e^{-T}.\end{aligned}$$

Combining yields the normal loss contribution plus the initialization remainder in (20).

Similarly, the tangential accumulated-error term is bounded by

$$\begin{aligned}&\int_{\widetilde{t}_{\mathrm{in}}}^{T-\delta} e^{-\int_u^{T-\delta}(h_{T-s}^{-1}-2)ds}\frac{4}{h_{T-u}}\mathbb{E}\|(U_{T-u}^{(i)})^{\top}e_A(T-u,\widetilde{Y}_u)\|^2\,du \\ &\qquad \lesssim C_\delta^{\|}\int_{\widetilde{t}_{\mathrm{in}}}^{T-\delta}\frac{w(T-u)}{h_{T-u}^2}\mathbb{E}\|(U_{T-u}^{(i)})^{\top}e_A(T-u,\widetilde{Y}_u)\|^2\,du.\end{aligned}$$

On the regular late-time center-dominant chart, the local comparison used in the submission is

$$U_{T-t}^{(i)}(U_{T-t}^{(i)})^{\top} \approx P_{T-t}^{\|}(m_{T-t}^{(i)}) \approx P_{T-t}^{\|}(\widetilde{Y}_t).$$

Using this tangent proxy and the same change-of-measure argument bounds the tangential term by

$$C_\delta^{\|}\int_0^{T-\delta}\frac{w(T-t)}{h_{T-t}^2}\mathbb{E}\|e_A^{\|}(T-t,X_t^{\leftarrow})\|^2\,dt + C_\delta^{\|}(\sqrt{d}+R)e^{-T}.$$

Finally, $X_t^{\leftarrow} \sim p_{T-t}$. Changing the integration variable to forward time and adding the geometric/noise contributions from Theorems 3.6–3.7 gives (20)–(21). □

I Analysis of the Training Process

Recall from (7) that

$$A_{k+1} = A_k(I_p - 2\eta\widetilde{U}) + 2\eta\widetilde{V},$$

where, writing $U_t = \mathbb{E}[\sigma_t(X_t)\sigma_t(X_t)^\top]$ and $V_t = \mathbb{E}[a_t X_0 \sigma_t(X_t)^\top]$,

$$\widetilde{U} = \int_\delta^T \frac{w(t)}{h_t^2}\frac{U_t}{p}\,dt, \qquad \widetilde{V} = \int_\delta^T \frac{w(t)}{h_t^2}\frac{V_t}{\sqrt{p}}\,dt.$$

The RFNN kernel matrix $\widetilde{U}$ has rank $r \le p$ and eigendecomposition $\widetilde{U} = \sum_{j=1}^r \lambda_j u_j u_j^\top$, with orthonormal $u_j \in \mathbb{R}^p$ and $\lambda_1 \ge \cdots \ge \lambda_r > 0$. Recursively applying the iteration gives

$$A_k = A_0(I_p - 2\eta\widetilde{U})^k + 2\eta\widetilde{V}\sum_{j=0}^{k-1}(I_p - 2\eta\widetilde{U})^j.$$

For $0 < \eta < 1/\lambda_1$, all positive-eigenvalue modes contract, and

$$A_k \longrightarrow A_\infty = A_0(I_p - \widetilde{U}\widetilde{U}^+) + \widetilde{V}\widetilde{U}^+,$$

where $\widetilde{U}^+$ is the Moore–Penrose pseudoinverse. The null-space component is unchanged by GD and makes no contribution to the feature output under the training measure.

Proposition I.1. *Let $\{A_k\}$ follow (7), with $0 < \eta < 1/\lambda_1$ and initialization A_0. Then*

$$\mathcal{L}_{\mathrm{MM}}(A_k) = \mathrm{Err}_{\mathrm{arc}} + \mathrm{Err}_{\mathrm{train}}(k), \qquad \mathcal{L}_{\mathrm{MM}}(A_\infty) = \mathrm{Err}_{\mathrm{arc}},$$

where

$$\mathrm{Err}_{\mathrm{arc}} = \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}\left\| m(t, X_t) - \frac{\widetilde{V}\widetilde{U}^+}{\sqrt{p}}\sigma_t(X_t)\right\|^2 dt,$$

$$\mathrm{Err}_{\mathrm{train}}(k) = \sum_{j=1}^r \lambda_j(1 - 2\eta\lambda_j)^{2k}\|a_j\|^2, \qquad a_j = (A_0 - \widetilde{V}\widetilde{U}^+)u_j \in \mathbb{R}^d.$$

Proof. The denoising mean-matching loss is quadratic in the trained matrix:

$$\begin{aligned}
\mathcal{L}_{\mathrm{DMM}}(A) &= \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}\left\| -a_t X_0 + \frac{A}{\sqrt{p}}\sigma_t(X_t)\right\|^2 dt \\
&= \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}\|a_t X_0\|^2 dt + \int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}\left\|\frac{A}{\sqrt{p}}\sigma_t(X_t)\right\|^2 dt \\
&\quad - 2\int_\delta^T \frac{w(t)}{h_t^2}\mathbb{E}\left\langle a_t X_0, \frac{A}{\sqrt{p}}\sigma_t(X_t)\right\rangle dt \\
&= C + \mathrm{tr}(A\widetilde{U}A^\top) - 2\,\mathrm{tr}(A\widetilde{V}^\top),
\end{aligned} \tag{22}$$

with $C = \int_\delta^T w(t)h_t^{-2}\mathbb{E}\|a_t X_0\|^2 dt$, independent of A. Consequently, $\nabla_A \mathcal{L}_{\mathrm{DMM}}(A) = 2(A\widetilde{U} - \widetilde{V})$, and every equilibrium $A^\star$ satisfies $A^\star\widetilde{U} = \widetilde{V}$. All equilibria have the same loss. Writing $\Delta A_k = A_k - A^\star$ and expanding (22),

$$\begin{aligned}
\mathcal{L}_{\mathrm{DMM}}(A_k) &= C + \mathrm{tr}(A^\star\widetilde{U}(A^\star)^\top) - 2\,\mathrm{tr}(A^\star\widetilde{V}^\top) + \mathrm{tr}(\Delta A_k\widetilde{U}\Delta A_k^\top) \\
&\quad + 2\left\{\mathrm{tr}(\Delta A_k\widetilde{U}(A^\star)^\top) - \mathrm{tr}(\Delta A_k\widetilde{V}^\top)\right\} \\
&= \mathcal{L}_{\mathrm{DMM}}(A^\star) + \mathrm{tr}(\Delta A_k\widetilde{U}\Delta A_k^\top) \\
&= \mathcal{L}_{\mathrm{DMM}}(A^\star) + \|\Delta A_k\widetilde{U}^{1/2}\|_F^2.
\end{aligned}$$

The cross term vanishes by the equilibrium identity. Moreover,

$$\begin{aligned}
\Delta A_{k+1} &= \Delta A_k(I_p - 2\eta\widetilde{U}) + 2\eta(\widetilde{V} - A^\star\widetilde{U}) \\
&= \Delta A_k(I_p - 2\eta\widetilde{U}).
\end{aligned}$$

Therefore

$$\mathcal{L}_{\mathrm{DMM}}(A_k) = \mathcal{L}_{\mathrm{DMM}}(A^\star) + \|\Delta A_0 (I_p - 2\eta \widetilde{U})^k \widetilde{U}^{1/2}\|_F^2,$$

and diagonalizing $\widetilde{U}$ yields

$$\mathcal{L}_{\mathrm{DMM}}(A_k) = \mathcal{L}_{\mathrm{DMM}}(A^\star) + \sum_{j=1}^{r} \lambda_j (1 - 2\eta\lambda_j)^{2k} \|a_j\|^2. \tag{23}$$

To pass from denoising loss to posterior-mean error, use the conditional-mean orthogonality established in Appendix B. In particular,

$$\begin{aligned}
\mathcal{L}_{\mathrm{DMM}}(A) &= \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E} \left\| m(t, X_t) - \frac{A}{\sqrt{p}} \sigma_t(X_t) \right\|^2 dt \\
&\quad + \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E} \|a_t X_0 - m(t, X_t)\|^2 dt \\
&= \mathcal{L}_{\mathrm{MM}}(A) + C_{\mathrm{den}},
\end{aligned}$$

where C_{den} is independent of A. The relevant cross term vanishes because

$$\mathbb{E}[a_t X_0 - m(t, X_t) \mid X_t] = 0.$$

It follows that

$$\begin{aligned}
\mathcal{L}_{\mathrm{MM}}(A_k) &= \mathcal{L}_{\mathrm{DMM}}(A_k) - C_{\mathrm{den}} \\
&= \sum_{j=1}^{r} \lambda_j (1 - 2\eta\lambda_j)^{2k} \|a_j\|^2 + \mathcal{L}_{\mathrm{DMM}}(A^\star) - C_{\mathrm{den}}.
\end{aligned} \tag{24}$$

Using $A^\star \widetilde{U} = \widetilde{V}$ and conditioning on X_t in the remaining term gives

$$\begin{aligned}
\mathcal{L}_{\mathrm{DMM}}(A^\star) - C_{\mathrm{den}} &= \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E} \left\langle m(t, X_t), m(t, X_t) - \frac{A^\star}{\sqrt{p}} \sigma_t(X_t) \right\rangle dt \\
&= \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E} \left\| m(t, X_t) - \frac{A^\star}{\sqrt{p}} \sigma_t(X_t) \right\|^2 dt.
\end{aligned}$$

Thus

$$\begin{aligned}
\mathcal{L}_{\mathrm{MM}}(A_k) &= \sum_{j=1}^{r} \lambda_j (1 - 2\eta\lambda_j)^{2k} \|a_j\|^2 \\
&\quad + \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E} \left\| m(t, X_t) - \frac{A^\star}{\sqrt{p}} \sigma_t(X_t) \right\|^2 dt.
\end{aligned} \tag{25}$$

The second term is the architecture-induced residual from the finite feature class. Although the equilibrium can include the initialization-dependent null component, this component has zero feature output under the training measure. Hence the residual is independent of initialization and can be represented using $A^\star = \widetilde{V}\widetilde{U}^+$. The first term is the finite-iteration optimization contribution and vanishes as $k \to \infty$ under the stated step-size condition. This proves the proposition. □

Proof of Theorem 4.2. We apply the same quadratic expansion to each directional loss, without assuming that the equilibrium for the total loss separately minimizes each projected loss. For $\dagger \in \{\perp, \|\}$, let $P_t^\dagger = P_t^\dagger(X_t)$ inside expectations and define

$$\mathcal{L}_{\mathrm{DMM}}^\dagger(A) = \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E} \left\| P_t^\dagger \left(\frac{A}{\sqrt{p}} \sigma_t(X_t) - a_t X_0 \right) \right\|^2 dt.$$

Since $P_t^\dagger$ is an orthogonal projector,

$$\begin{aligned}
\mathcal{L}_{\mathrm{DMM}}^\dagger(A) &= C^\dagger + \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E} \left\| P_t^\dagger \frac{A}{\sqrt{p}} \sigma_t(X_t) \right\|^2 dt \\
&\quad - 2 \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E} \left\langle P_t^\dagger a_t X_0, \frac{A}{\sqrt{p}} \sigma_t(X_t) \right\rangle dt,
\end{aligned}$$

where $C^\dagger = \int_\delta^T w(t) h_t^{-2} \mathbb{E}\|P_t^\dagger a_t X_0\|^2 dt$. With $A_k = A^\star + \Delta A_k$,

$$\mathcal{L}^\dagger_{\mathrm{DMM}}(A_k) = \mathcal{L}^\dagger_{\mathrm{DMM}}(A^\star) + \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E}\left\| P_t^\dagger \frac{\Delta A_k}{\sqrt{p}} \sigma_t(X_t) \right\|^2 dt + 2\int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E}\left\langle P_t^\dagger \left(\frac{A^\star}{\sqrt{p}} \sigma_t(X_t) - a_t X_0 \right), \frac{\Delta A_k}{\sqrt{p}} \sigma_t(X_t) \right\rangle dt.$$

Conditional-mean orthogonality remains valid after multiplication by $P_t^\dagger(X_t)$. Subtracting the projected conditional variance therefore yields

$$\begin{aligned}\mathcal{L}^\dagger_{\mathrm{MM}}(A_k) = &\int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E}\left\| P_t^\dagger \left(m(t, X_t) - \frac{A^\star}{\sqrt{p}} \sigma_t(X_t) \right) \right\|^2 dt \\ &+ \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E}\left\| P_t^\dagger \frac{\Delta A_k}{\sqrt{p}} \sigma_t(X_t) \right\|^2 dt \\ &+ 2\int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E}\left\langle P_t^\dagger \left(\frac{A^\star}{\sqrt{p}} \sigma_t(X_t) - m(t, X_t) \right), \frac{\Delta A_k}{\sqrt{p}} \sigma_t(X_t) \right\rangle dt.\end{aligned}$$

The first integral defines $\mathrm{Err}^\dagger_{\mathrm{arc}}$. The sum of the second and third integrals defines $\mathrm{Err}^\dagger_{\mathrm{train}}(k)$. Thus

$$\mathcal{L}^\dagger_{\mathrm{MM}}(A_k) = \mathrm{Err}^\dagger_{\mathrm{arc}} + \mathrm{Err}^\dagger_{\mathrm{train}}(k).$$

The cross term need not vanish separately in each direction, so $\mathrm{Err}^\dagger_{\mathrm{train}}(k)$ is a signed finite-iteration correction, unlike the nonnegative total optimization error in Proposition I.1.

For an explicit spectral expression, put $\sigma_{t,u_j} = u_j^\top \sigma_t(X_t)/\sqrt{p}$. On the training measure,

$$\frac{\Delta A_k}{\sqrt{p}} \sigma_t(X_t) = \sum_{j=1}^r (1 - 2\eta\lambda_j)^k \sigma_{t,u_j} a_j.$$

The quadratic term becomes

$$\begin{aligned}&\int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E}\left\| P_t^\dagger \frac{\Delta A_k}{\sqrt{p}} \sigma_t(X_t) \right\|^2 dt \\ &= \sum_{j,l=1}^r (1 - 2\eta\lambda_j)^k (1 - 2\eta\lambda_l)^k a_j^\top \left(\int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E}[\sigma_{t,u_j} \sigma_{t,u_l} P_t^\dagger] dt \right) a_l.\end{aligned}$$

Similarly, the linear term becomes

$$\begin{aligned}&2\int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E}\left\langle P_t^\dagger \left(\frac{A^\star}{\sqrt{p}} \sigma_t(X_t) - m(t, X_t) \right), \frac{\Delta A_k}{\sqrt{p}} \sigma_t(X_t) \right\rangle dt \\ &= 2\sum_{j=1}^r (1 - 2\eta\lambda_j)^k a_j^\top \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E}\left[\sigma_{t,u_j} P_t^\dagger \left(\frac{A^\star}{\sqrt{p}} \sigma_t(X_t) - m(t, X_t) \right) \right] dt.\end{aligned}$$

Consequently,

$$\mathrm{Err}^\dagger_{\mathrm{train}}(k) = \sum_{j,l=1}^r (1 - 2\eta\lambda_j)^k (1 - 2\eta\lambda_l)^k a_j^\top P^\dagger_{jl} a_l + 2\sum_{j=1}^r (1 - 2\eta\lambda_j)^k a_j^\top b_j^\dagger,$$

where

$$\begin{aligned}P^\dagger_{jl} &= \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E}[\sigma_{t,u_j} \sigma_{t,u_l} P_t^\dagger(X_t)] dt, \\ b_j^\dagger &= \int_\delta^T \frac{w(t)}{h_t^2} \mathbb{E}\left[\sigma_{t,u_j} P_t^\dagger(X_t) \left(\frac{A^\star}{\sqrt{p}} \sigma_t(X_t) - m(t, X_t) \right) \right] dt.\end{aligned} \tag{26}$$

These expressions apply to both normal and tangential directions. Since $P_t^\perp + P_t^\parallel = I_d$ and $A^\star \widetilde{U} = \widetilde{V}$,

$$P^\perp_{jl} + P^\parallel_{jl} = \lambda_j \mathbf{1}_{\{j=l\}} I_d, \qquad b_j^\perp + b_j^\parallel = 0.$$

Therefore $\mathrm{Err}^\perp_{\mathrm{train}}(k) + \mathrm{Err}^\parallel_{\mathrm{train}}(k) = \mathrm{Err}_{\mathrm{train}}(k)$. Combining the directional identities with Theorem 4.1 proves the geometry bounds in Theorem 4.2. The factors $1 - 2\eta\lambda_j$ also explain the initialization- and spectrum-dependent behavior in the two-point example. □

J Experimental Details

This appendix lists the model architectures and hyperparameters not included in the main text.

J.1 Two Points in 2D Plane

Model architectures. The RFNN follows the architecture in Section 2, with width $p = 2000$ and temporal embedding parameter $K_t = 128$.

Training and loss computation. We train by gradient descent for 10^6 epochs with learning rate 0.005, using the full-batch SGD optimizer in PyTorch and the loss in (5). The time integral is evaluated numerically using the trapezoidal rule with 2000 discretization points.

SDE and sampling details. The forward drift is $f(x,t) = -x$ and the diffusion coefficient is $g(t) = \sqrt{2}$. Generation uses Euler–Maruyama with $N = 1000$ steps on a geometric schedule from $T = 10$ to $\delta = t_{\min} = 10^{-3}$:

$$t_i = T(t_{\min}/T)^{i/N}, \qquad i = 0, \dots, N.$$

Computational resources. Experiments use a single NVIDIA RTX 4090 GPU. The machine has 24.0 GB GPU memory, a 14-core AMD EPYC 7453 CPU, and 60.1 GB system memory. Nearly all training and evaluation computations run on the GPU; the relatively long RFNN training time comes primarily from the number of epochs rather than heavy per-epoch computation. Completing the RFNN experiments takes approximately 1.5 hours.

J.2 More Points in 2D Plane

The RFNN architecture, training method, and sampler are the same as in Appendix J.1. The four-point example in Figure 2 is trained for 500,000 epochs, and the three-point example in Figure 3 for 300,000 epochs, both with learning rate 0.005. These experiments use a single NVIDIA RTX 4090 GPU and take approximately one hour.

J.3 MNIST

Model architectures. All components use multilayer perceptrons. The VAE encoder maps flattened images in $\mathbb{R}^{784}$ through hidden layers of sizes 400 and 200, with ReLU activations, to the latent mean and log-variance parameters. The decoder mirrors this structure, with latent dimension $\to 200 \to 400 \to 784$, and a final Sigmoid activation restricting pixels to $[0, 1]$. The experiments use a 32-dimensional latent space. The latent score network maps time through a 128-dimensional sinusoidal embedding, concatenates it with the latent input, and applies fully connected layers with widths input $\to 256 \to 256 \to$ latent dimension and ReLU activations.

Training configuration. Both models use Adam with batch size 1024. The VAE is trained for 200 epochs with learning rate 10^{-3} using the evidence lower bound loss. The latent diffusion model is trained for 300 epochs with learning rate 10^{-4} and the denoising objective in (5), with constant weight $w(t) = 1$.

SDE and sampling details. We use the OU forward coefficients $f(x,t) = -x$ and $g(t) = \sqrt{2}$, and the Euler–Maruyama sampler with $N = 1000$ steps. The noise-time grid is geometric from $T = 10$ to $t_{\min} = \delta = 0.005$, namely $t_i = T(t_{\min}/T)^{i/N}$.

Computational resources. The VAE and latent score model are trained on a single NVIDIA RTX 4090 GPU, taking less than ten minutes together. Sampling and geometric evaluation are more expensive: evaluating all four ridge dimensions $d^* = 3, 5, 8, 12$ takes approximately forty minutes.

J.4 Projection onto Log-density Ridge Sets

Given a generated state $x_0 \in \mathbb{R}^d$ at a fixed diffusion time, we estimate its nearest point on the log-density ridge $\mathcal{R}_{d^*}(p;\beta)$ by a constrained nonlinear optimization problem:

$$\begin{aligned} \min_{y\in\mathbb{R}^d} \quad & \tfrac{1}{2}\|y-x_0\|^2 \\ \text{subject to} \quad & F(y) := E(y)^\top \nabla \log p(y) = 0, \\ & \lambda_{d^*+1}(y) \le -\beta. \end{aligned} \tag{27}$$

Here $E(y)$ contains the bottom $d-d^*$ Hessian eigenvectors, and $F(y) \in \mathbb{R}^{d-d^*}$ is the normal score residual. For the equality constraint we introduce a multiplier $\nu \in \mathbb{R}^{d-d^*}$ and the Lagrangian

$$L(y,\nu) = \tfrac{1}{2}\|y-x_0\|^2 + \nu^\top F(y). \tag{28}$$

At an interior feasible solution of the curvature constraint, the equality-constrained stationarity equations are

$$\begin{cases} (y-x_0) + J(y)^\top \nu = 0, \\ F(y) = 0, \end{cases} \qquad J(y) := \nabla_y F(y) \in \mathbb{R}^{(d-d^*)\times d}. \tag{29}$$

The numerical procedure below seeks a feasible stationary candidate; feasibility alone does not certify the globally nearest point, and an active curvature constraint requires its own KKT treatment.

In high dimensions, finite-difference approximations of J can be unstable. We instead use the product rule:

$$J(y) = E(y)^\top \nabla^2 \log p(y) + (\nabla_y E(y))^\top \nabla \log p(y). \tag{30}$$

Write $H(y) = \nabla^2 \log p(y)$, let $T(y)$ contain the top d^* eigenvectors, and let $\Lambda_\parallel(y)$ and $\Lambda_\perp(y)$ be the corresponding diagonal eigenvalue matrices. The first term in (30) is $\Lambda_\perp E^\top$. Differentiating $HE = E\Lambda_\perp$ with respect to coordinate $y^{(\ell)}$ gives

$$(\partial_\ell H)E + H(\partial_\ell E) = (\partial_\ell E)\Lambda_\perp + E(\partial_\ell \Lambda_\perp).$$

Premultiplication by $T^\top$ and the orthogonality $T^\top E = 0$ yield the Sylvester equation

$$\Lambda_\parallel B_\ell - B_\ell \Lambda_\perp = -T^\top(\partial_\ell H)E, \qquad B_\ell = T^\top \partial_\ell E.$$

The strict eigengap $\lambda_{d^*} > \lambda_{d^*+1}$ separates the two spectra. In a local frame gauge with vanishing within-normal rotation at the query point, the cross-subspace derivative is $\partial_\ell E = T B_\ell$. Equivalently, vectorization gives the Kronecker linear system

$$\mathrm{vec}(B_\ell) = -\big(I_{d-d^*} \otimes \Lambda_\parallel - \Lambda_\perp^\top \otimes I_{d^*}\big)^{-1} \mathrm{vec}\big(T^\top(\partial_\ell H)E\big).$$

The ℓth column of the second term in (30) is $(\partial_\ell E)^\top \nabla \log p(y)$. This accounts for variation of the Hessian eigenspace rather than freezing its orientation.

When J is ill-conditioned, we use a Levenberg–Marquardt regularization with damping $\sigma > 0$. Linearizing $F(y_k + s) \approx F(y_k) + J(y_k)s$ leads to

$$\begin{aligned} J(y_k)J(y_k)^\top \nu &= (1+\sigma)F(y_k) - J(y_k)(y_k - x_0), \\ s &= -\frac{(y_k - x_0) + J(y_k)^\top \nu}{1+\sigma}. \end{aligned} \tag{31}$$

The dual system is additionally regularized by a small diagonal perturbation in the implementation. To penalize constraint violations, the trial step is evaluated with

$$M(y) = \tfrac{1}{2}\|y-x_0\|^2 + \tfrac{c}{2}\|F(y)\|^2 + \gamma \max\{0, \lambda_{d^*+1}(y) + \beta\}^2. \tag{32}$$

A step is accepted if it decreases M; otherwise the damping is increased. The iteration stops when $\|F(y)\| \le \epsilon_{\mathrm{tol}}$ and $\lambda_{d^*+1}(y) \le -\beta$, or the maximum iteration count is reached. These conditions test ridge feasibility, not global nearest-point optimality.

For MNIST, we use $d^* = 3, 5, 8, 12$, curvature margin $\beta = 10^{-3}$, at most 30 iterations, tolerance $\epsilon_{\mathrm{tol}} = 10^{-6}$, and diagonal regularization $\epsilon_{\mathrm{reg}} = 10^{-6}$. Both penalty weights are $c = \gamma = 100$. During the 1000-step sampling trajectory, we evaluate the geometry every 100 steps using 200 samples. Because reliability depends on spectral separation, we also track eigengaps and solver failures, as reported in Appendix J.5.

Algorithm 1 Projection onto log-density ridge sets via regularized LM

Require: State $x_0 \in \mathbb{R}^d$, score $\nabla \log p$, intrinsic dimension d^*.

Set $\beta = 10^{-3}$, $\epsilon_{\mathrm{tol}} = 10^{-6}$, $N_{\max} = 30$, $\sigma = 10^{-3}$, $\epsilon_{\mathrm{reg}} = 10^{-6}$, $c = \gamma = 100$.
Initialize $y \leftarrow x_0$.
for $k = 0, \dots, N_{\max} - 1$ **do**
    Compute $g \leftarrow \nabla \log p(y)$ and $H \leftarrow \nabla^2 \log p(y)$.
    Diagonalize H to obtain $T, E, \Lambda_{\parallel}, \Lambda_{\perp}$.
    Evaluate $F \leftarrow E^\top g$.
    if $\|F\| \le \epsilon_{\mathrm{tol}}$ and $\lambda_{d^*+1} \le -\beta$ **then**
        return y ▷ Ridge-feasible candidate
    end if
    for $\ell = 1, \dots, d$ **do**
        Solve $\Lambda_{\parallel} B_\ell - B_\ell \Lambda_{\perp} = -T^\top (\partial_\ell H) E$.
        Set $D_\ell \leftarrow T B_\ell$ in the local frame gauge.
    end for
    Assemble $J \leftarrow \Lambda_{\perp} E^\top + [D_1^\top g, \dots, D_d^\top g]$.
    Solve $(JJ^\top + \epsilon_{\mathrm{reg}} I)\nu = (1+\sigma)F - J(y - x_0)$.
    Set $s \leftarrow -\big((y - x_0) + J^\top \nu\big)/(1+\sigma)$.
    Set $y_{\mathrm{trial}} \leftarrow y + s$.
    Evaluate the merit function M in (32).
    if $M(y_{\mathrm{trial}}) < M(y)$ **then**
        $y \leftarrow y_{\mathrm{trial}}$, $\sigma \leftarrow \max\{\sigma/2, \epsilon_{\mathrm{reg}}\}$.
    else
        $\sigma \leftarrow 10\sigma$.
    end if
end for
return y ▷ Approximation after the iteration limit

J.5 Stability

Tables 1–4 track the numerical geometric quantities during reverse sampling. *Step* and t are the sampling index and noise time. *Mean Dist.* is the mean computed distance to a ridge candidate. *Curv. OK* counts samples satisfying $\lambda_{d^*+1} \leq -\beta$. *Mean Resid.* is the initial normal score residual $\|F(x_t)\|$ before projection. *Mean Gap* is the mean tangent/normal Hessian eigengap. *Failures* counts solver runs that do not converge within 30 iterations. The reported *Reliable* count means convergence with a strictly positive eigengap; it is not a certificate of projection uniqueness or nearest-point optimality.

In the earliest stages (Steps 0 and 100), many samples meet this binary reliability test even though the eigengaps are extremely small, typically below 10^{-6}. The geometry is then close to isotropic Gaussian noise, with poorly separated spectral directions and weak numerical conditioning. Distances computed in this regime have limited geometric meaning. We therefore omit Steps 0 and 100, corresponding to $t = 10.0$ and approximately 4.67, from Figures 6(a)–(b).

Why the eigengap is small at large noise scales. Since $X_t \mid X_0 \sim \mathcal{N}(a_t X_0, h_t I)$,

$$\nabla^2 \log p_t(x) = -\frac{1}{h_t} I + \frac{a_t^2}{h_t^2} \operatorname{Cov}(X_0 \mid X_t = x).$$

The isotropic term shifts all eigenvalues equally. If $\mu_j(x)$ are the eigenvalues of the conditional covariance, then

$$\lambda_{d^*}(\nabla^2 \log p_t(x)) - \lambda_{d^*+1}(\nabla^2 \log p_t(x)) = \frac{a_t^2}{h_t^2}\big(\mu_{d^*}(x) - \mu_{d^*+1}(x)\big).$$

At large t, h_t is close to one and a_t is small, so the spectral separation becomes small. This explains the near-degeneracy observed in the initial sampling steps.

Table 1: Dynamic ridge-distance monitoring during reverse sampling, $d^* = 3$. Values reproduce the submitted experiment; "Reliable" denotes the numerical test described in the text.

Step	t	Mean Dist.	Reliable	Curv. OK	Mean Resid.	Mean Gap	Failures
0	10.00000	5.422006	200	200	5.422012	0.000000	0
100	4.67624	5.361732	200	200	5.361917	0.000004	0
200	2.18672	5.272104	200	200	5.333307	0.000738	0
300	1.02257	5.149778	200	200	5.800307	0.013401	0
400	0.47818	4.451155	200	200	6.642476	0.080860	0
500	0.22361	3.426006	197	198	8.438073	0.409678	1
600	0.10456	2.501570	160	164	11.973254	1.657129	4
700	0.04890	1.838713	177	178	18.188848	4.139380	1
800	0.02287	1.442437	193	194	30.535279	5.491860	1
900	0.01069	1.197336	199	199	53.974764	2.059901	0
1000	0.00500	1.233171	200	200	116.260759	0.420446	0

Table 2: Dynamic ridge-distance monitoring during reverse sampling, $d^* = 5$. Values reproduce the submitted experiment; "Reliable" denotes the numerical test described in the text.

Step	t	Mean Dist.	Reliable	Curv. OK	Mean Resid.	Mean Gap	Failures
0	10.00000	5.130307	200	200	5.130310	0.000000	0
100	4.67624	2.989573	200	200	2.989411	0.000007	0
200	2.18672	2.882148	200	200	2.919268	0.001082	0
300	1.02257	2.724726	200	200	3.124859	0.013572	0
400	0.47818	2.207066	200	200	3.423936	0.058334	0
500	0.22361	1.710056	200	200	4.212194	0.196339	0
600	0.10456	1.407467	180	180	5.857416	0.852296	0
700	0.04890	1.058827	192	192	9.560239	2.028122	0
800	0.02287	0.799892	200	200	16.567018	0.794553	0
900	0.01069	0.741769	200	200	32.952072	0.029825	0
1000	0.00500	0.664260	200	200	65.937224	0.001428	0

Table 3: Dynamic ridge-distance monitoring during reverse sampling, $d^* = 8$. Values reproduce the submitted experiment; "Reliable" denotes the numerical test described in the text.

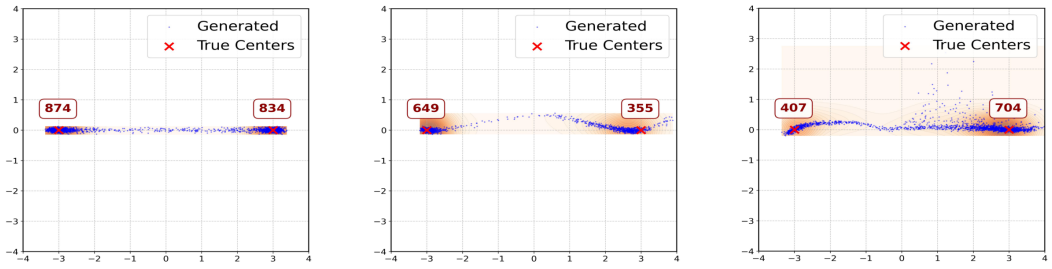
Step	t	Mean Dist.	Reliable	Curv. OK	Mean Resid.	Mean Gap	Failures
0	10.00000	4.880301	200	200	4.880306	0.000000	0
100	4.67624	4.852153	200	200	4.852658	0.000004	0
200	2.18672	4.776520	200	200	4.910469	0.000521	0
300	1.02257	4.649101	200	200	5.567016	0.009011	0
400	0.47818	4.142639	200	200	6.632765	0.047487	0
500	0.22361	3.048265	200	200	8.350910	0.157566	0
600	0.10456	2.196695	200	200	11.629959	0.269339	0
700	0.04890	1.566005	200	200	16.753251	0.163079	0
800	0.02287	1.209245	200	200	26.358911	0.020655	0
900	0.01069	1.014856	200	200	44.848112	0.000287	0
1000	0.00500	1.091011	200	200	97.710713	0.000013	0

Table 4: Dynamic ridge-distance monitoring during reverse sampling, $d^* = 12$. Values reproduce the submitted experiment; "Reliable" denotes the numerical test described in the text.

Step	t	Mean Dist.	Reliable	Curv. OK	Mean Resid.	Mean Gap	Failures
0	10.00000	4.463758	200	200	4.463762	0.000000	0
100	4.67624	4.484123	200	200	4.484518	0.000000	0
200	2.18672	4.381252	200	200	4.437409	0.000049	0
300	1.02257	4.121614	200	200	4.737415	0.000577	0
400	0.47818	3.480388	200	200	5.719486	0.002088	0
500	0.22361	2.561228	200	200	7.348668	0.004829	0
600	0.10456	1.824004	200	200	10.150477	0.006965	0
700	0.04890	1.314270	200	200	14.600166	0.003107	0
800	0.02287	0.982827	200	200	22.123408	0.000174	0
900	0.01069	0.820614	200	200	36.948018	0.000017	0
1000	0.00500	0.892560	200	200	85.796590	0.000000	0

K More Experiment Results

K.1 Biases under Different Initializations



(a) Zero (b) Ones (c) Slow-Spec

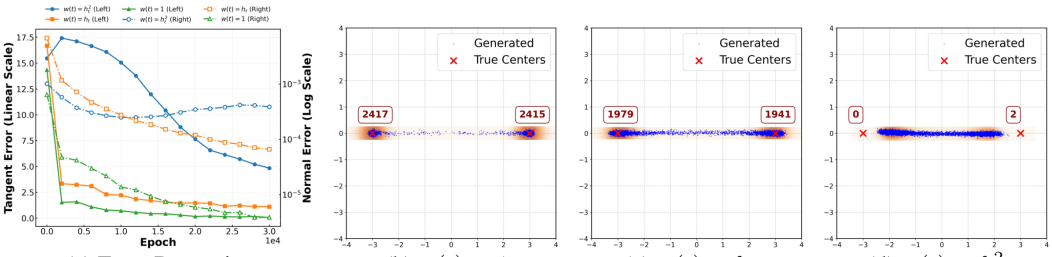
Figure 7: **Initialization effects at epoch 40k.** Generated samples under zero, all-ones, and slow-spectrum initializations. The background shading represents KDE.

We test the initialization effects predicted by the RFNN analysis in Section 4.2. Figure 7 compares finite-training-time generation under three initialization schemes: zero, all-ones, and slow-spectrum. Zero initialization yields samples that remain essentially on the horizontal ridge, whereas the other two produce a visible arch before stronger alignment is reached. This agrees with the role of the normal optimization contribution in Section 4.2: the initialization changes how quickly normal error is reduced. Emphasizing slow spectral modes can delay this reduction and produce the transient arch-shaped geometry.

K.2 Numerical Verification of Geometric Biases for Two-Point on MLP

We also test the two-point example using a standard MLP instead of an RFNN. The purpose is to check that the geometric picture in Section 3 is not specific to the RFNN parametrization. The dataset is $\mathcal{D} = \{(-3, 0), (3, 0)\} \subset \mathbb{R}^2$, for which the ridge is the horizontal axis and the tangent and normal directions are explicit.

Model architectures. The two-layer MLP parametrizes $m_A(x, t)$. The time variable is mapped to a 32-dimensional sinusoidal embedding, concatenated with $x \in \mathbb{R}^2$, passed through a hidden layer of 128 units with ReLU activation, and projected to $\mathbb{R}^2$.



(a) Error Dynamics (b) $w(t) = 1$ (c) $w(t) = h_t$ (d) $w(t) = h_t^2$

Figure 8: **Error dynamics and generated samples for the MLP.** (a) Tangential errors (solid, left linear axis) and normal errors (dash-dot, right logarithmic axis) during training. (b)–(d) Generated samples for the three weights $w(t) = 1, h_t, h_t^2$. Boxed counts use radius 0.5 around the target modes; the background shows KDE.

Training and loss computation. Training uses 3×10^4 epochs, learning rate 10^{-4}, the full-batch SGD optimizer in PyTorch, and the loss in (5). The time integral is approximated by the trapezoidal rule with 2000 discretization points.

SDE and sampling details. The forward coefficients are $f(x,t) = -x$ and $g(t) = \sqrt{2}$. Euler–Maruyama generation uses $N = 1000$ time steps on the geometric grid $t_i = T(t_{\min}/T)^{i/N}$ from $T = 10$ to $t_{\min} = \delta = 10^{-3}$.

Computational resources. All MLP experiments use a single NVIDIA RTX 4090 GPU and take approximately fifteen minutes.

Figure 8 exhibits the same directional pattern as the RFNN experiment: normal error stays small and generated samples concentrate near $y = 0$, whereas tangential error settles at a larger floor and samples can spread between the two training points. The weight schedule primarily changes tangential behavior. The constant weight gives the strongest concentration near the training points; $w(t) = h_t$ and h_t^2 lead to progressively more pronounced intermediate generation. These observations support the qualitative directional interpretation in Remark 3.8 without requiring the RFNN-specific training calculation.

K.3 RFNN on Different Sets

Additional two-dimensional examples illustrate data-induced ridge geometries beyond a single line segment. These experiments are qualitative illustrations rather than quantitative verification of every assumption in the theorems. The captions describe the relative prominence of directional effects, not disjoint intervals in which one effect must finish before another begins.

Two-point case: $(-3,0),(3,0)$. Figures 9–11 compare the weights $w(t) = 1, h_t, h_t^2$. The trajectories approach the ridge and continue evolving both normally and tangentially; the later tangential behavior depends substantially on the weight.

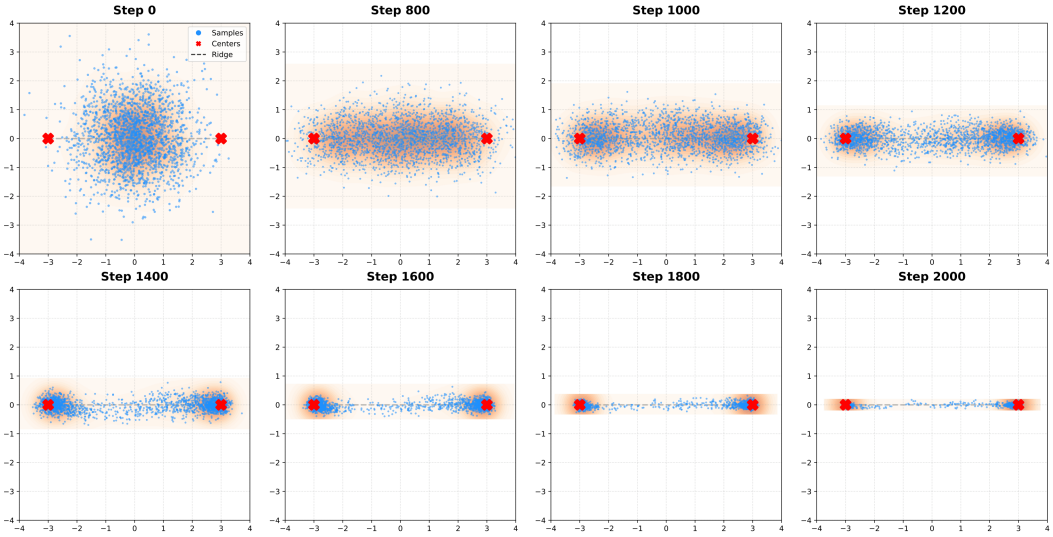


Figure 9: Two-point RFNN sampling with $w(t) = 1$. Snapshots show 2000 particles, data points, the ridge, and KDE contours. Samples approach the ridge neighborhood around Step 800, followed by increasing normal concentration and later visible tangential redistribution. These directional effects can coexist.

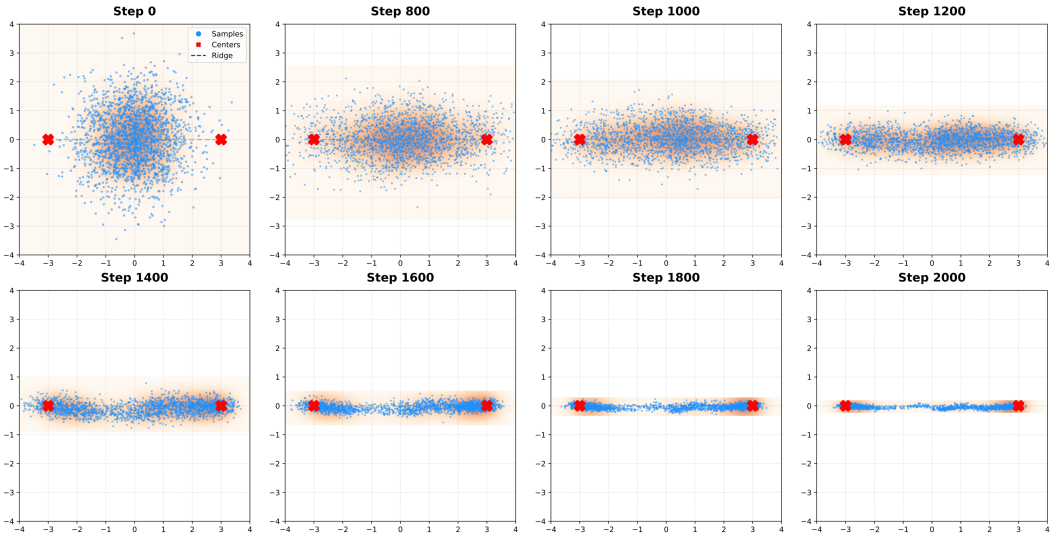


Figure 10: Two-point RFNN sampling with $w(t) = h_t$. The displayed 2000 particles approach the ridge neighborhood around Step 800. The remaining snapshots show normal alignment together with tangential motion, with the latter particularly visible near the end of sampling.

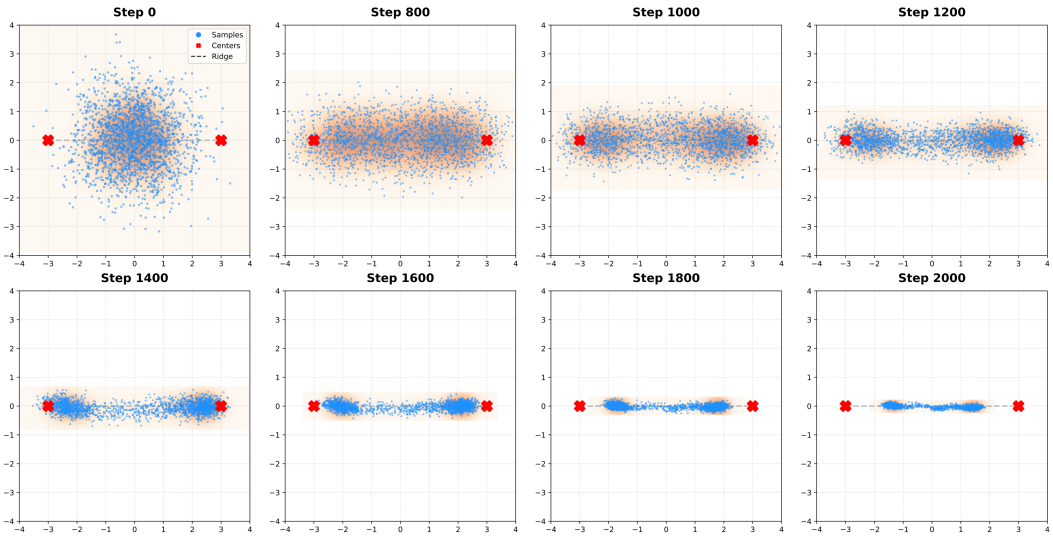


Figure 11: Two-point RFNN sampling with $w(t) = h_t^2$. The displayed particles approach the ridge neighborhood around Step 1200 and retain appreciable along-ridge spread at the final time. Background contours represent KDE.

Three-point case: $(0,0),(5,0),(0,5)$**.** This example illustrates that the relevant geometry is not restricted to straight interpolation between two modes. Figures 12–14 show the weights $w(t) = 1, h_t, h_t^2$. The late-time sample geometry follows the ridge induced by the three-point configuration, with its bent branches, rather than all straight segments joining the training points.

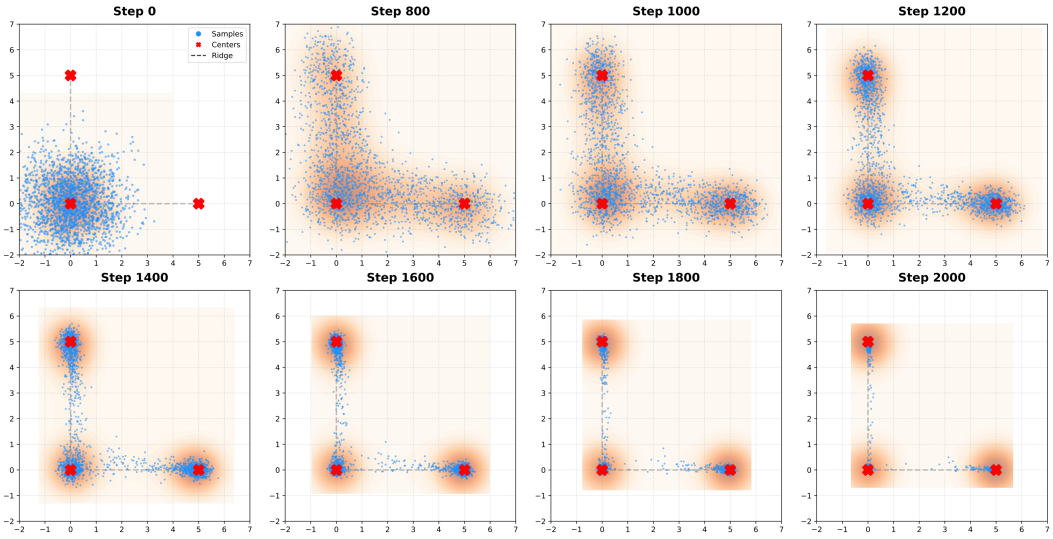


Figure 12: Three-point RFNN sampling with $w(t) = 1$, showing 2000 particles and KDE contours. Samples reach the ridge neighborhood around Step 800; subsequent snapshots show continued normal alignment and tangential evolution relative to the data-induced ridge.

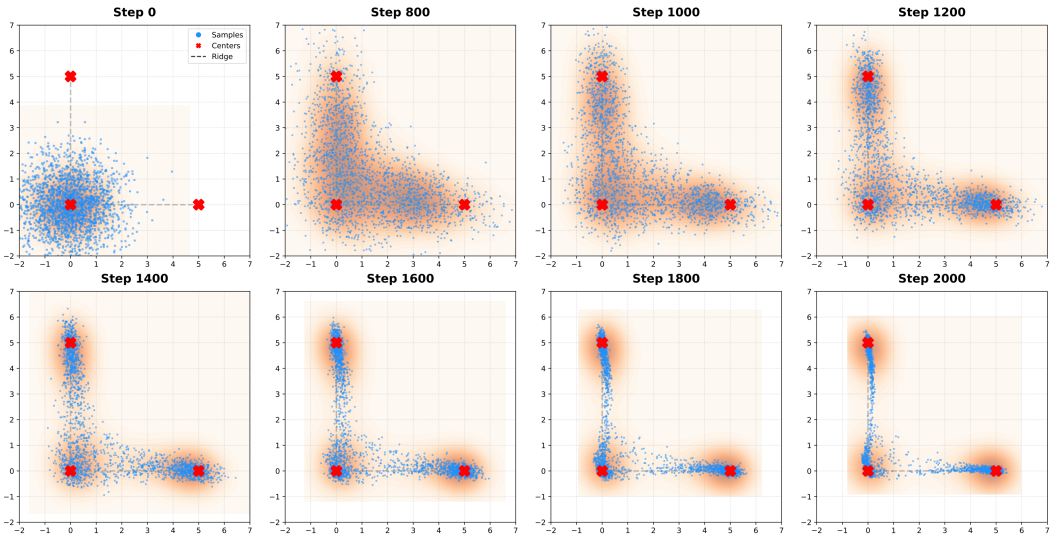


Figure 13: Three-point RFNN sampling with $w(t) = h_t$. Samples approach the ridge neighborhood around Step 1000 and become increasingly concentrated around its branches while continuing to move along them.

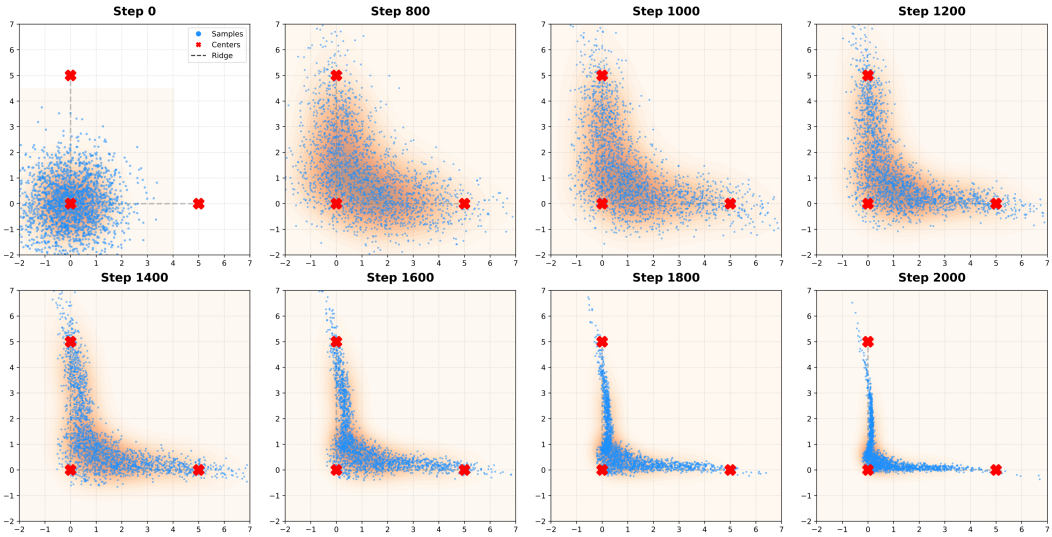


Figure 14: Three-point RFNN sampling with $w(t) = h_t^2$. The snapshots show entry near Step 1200 and the coexistence of normal alignment with residual tangential spread at later times.

Four-point cases. These examples probe more complicated ridge structures. Figure 15 uses a symmetric four-point configuration, Figure 16 an asymmetric configuration, and Figure 17 three points forming a triangle with a fourth point in its interior. The ridge can become distorted or develop more intricate interior branches, but it remains a useful reference for the displayed generation geometry.

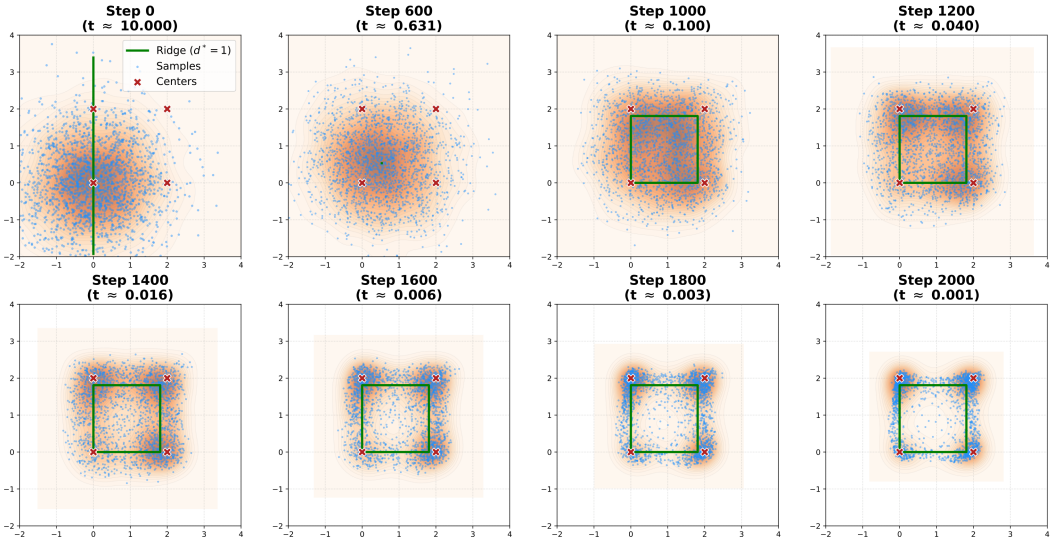


Figure 15: Symmetric four-point RFNN sampling with $w(t) = h_t$. The figure shows 2000 particles, training points, the computed ridge, and KDE contours at the displayed noise levels. Concentration near the ridge becomes apparent around Step 1200 and persists during later directional evolution.

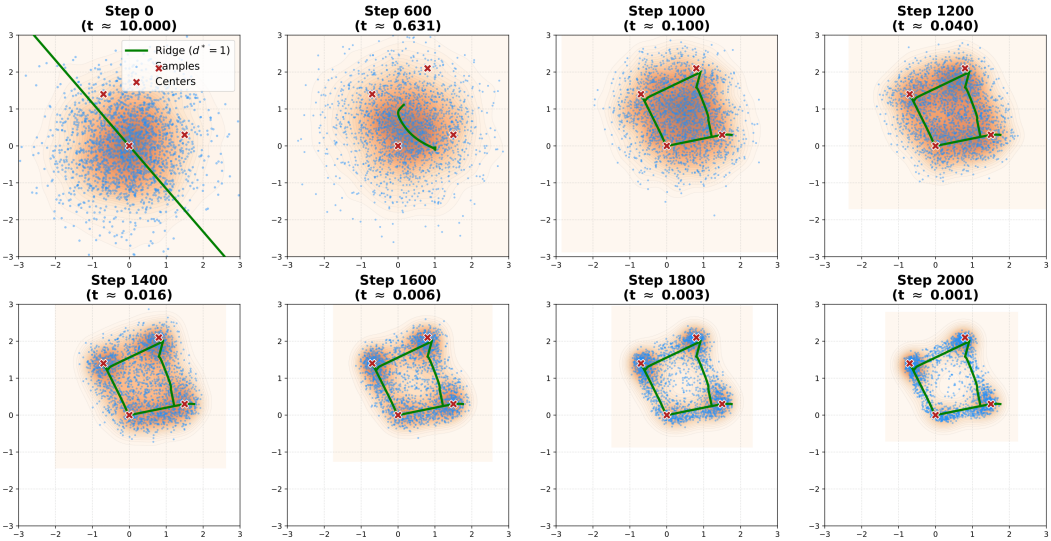


Figure 16: Asymmetric four-point RFNN sampling with $w(t) = h_t$. Generated samples organize around the moving ridge despite the distorted geometry. Normal alignment and tangential motion may both continue after entry.

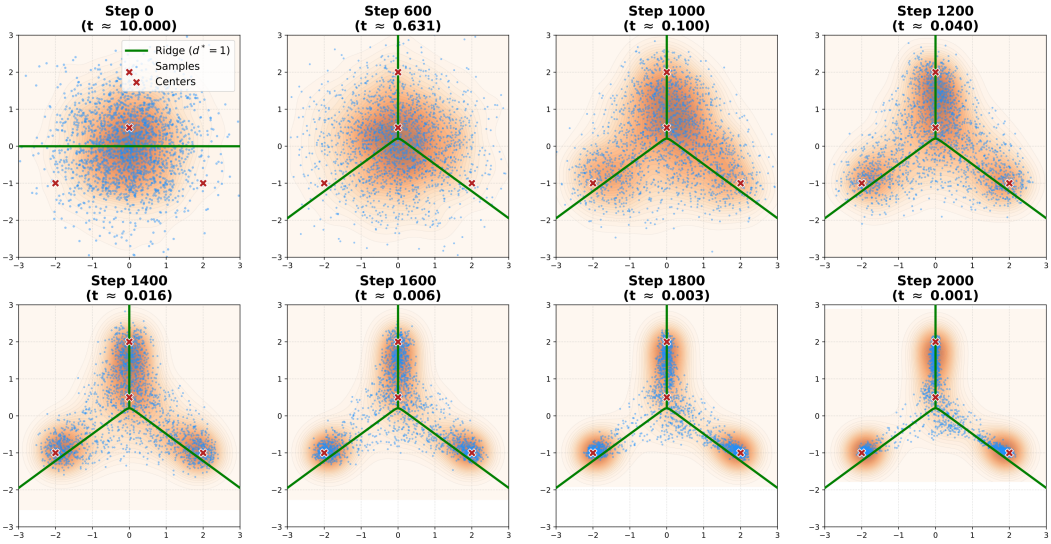


Figure 17: RFNN sampling with three outer points and one interior point, using $w(t) = h_t$. The 2000-particle snapshots illustrate the more intricate interior ridge geometry and the corresponding organization of generated samples.

K.4 MNIST Trajectories

Figures 18–19 complement the quantitative MNIST results with decoded images along ten independent trajectories. The large-scale digit structure becomes visible before the final sampling steps, while the final 200 steps make comparatively small visual changes. This is consistent with limited late-time refinement in these examples. It does not by itself establish that tangential velocity is zero, or that a plateau in a tangential residual certifies a stationary projected distribution.

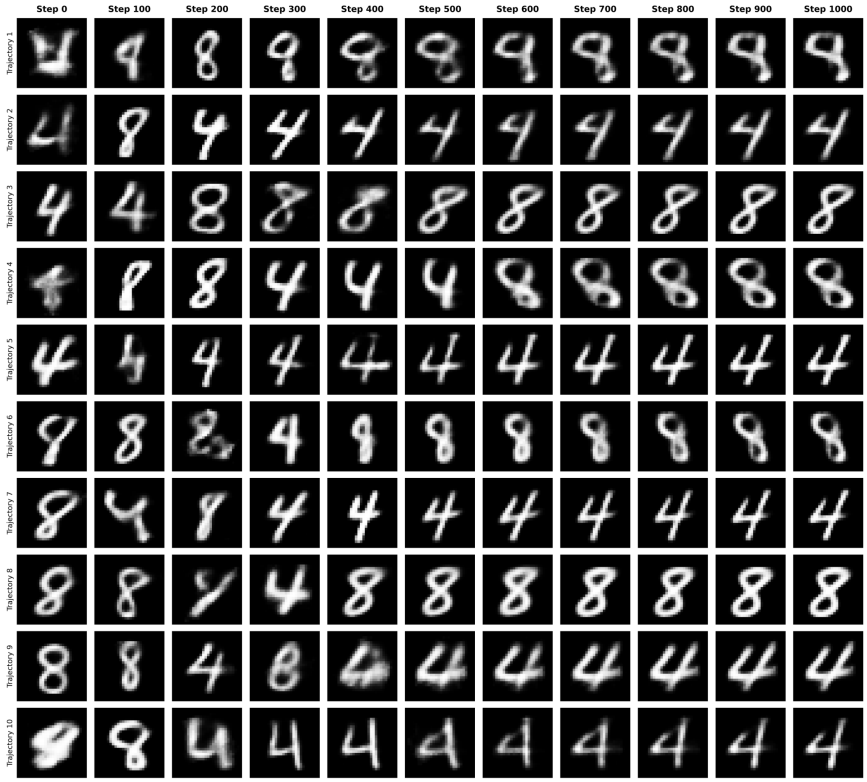


Figure 18: Full sampling evolution for ten independent MNIST latent trajectories. Sampling starts from standard Gaussian latent noise at Step 0. Recognizable structure emerges around Step 200, and the decoded digits 4 or 8 are clearly formed by approximately Step 800.

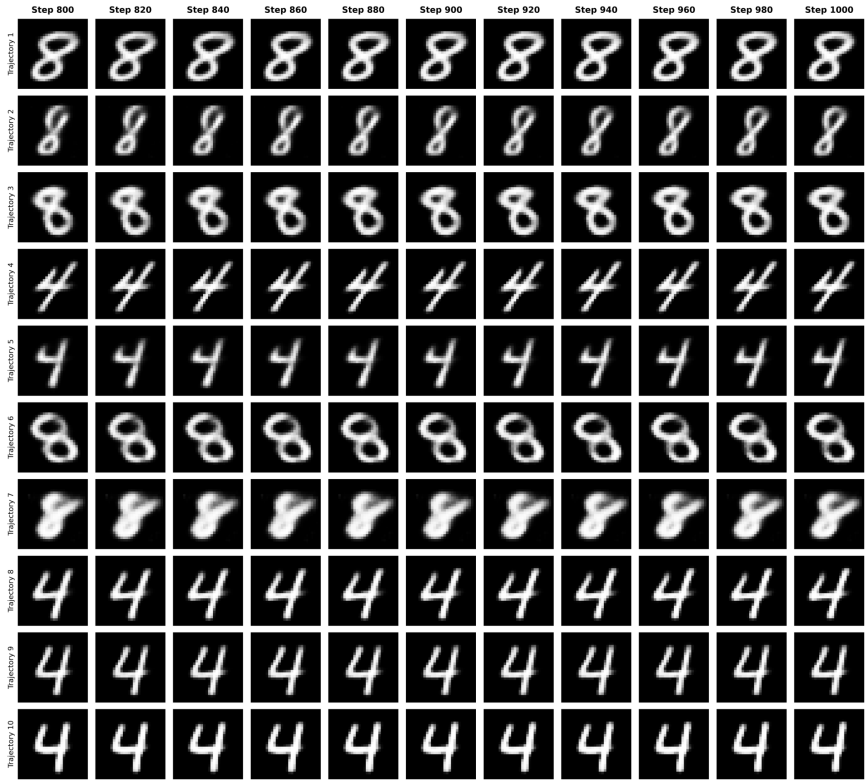


Figure 19: Decoded images during Steps 800–1000. Changes in these trajectories are visually small and mainly refine existing digit structure. Visual stability is distinct from a claim of exactly vanishing motion along the ridge.

K.5 RFNN Experiments on Two-dimensional Lattices

The experiments in this subsection aim to show more clearly how the geometric distribution of the data affects the generated samples and the associated ridge geometry. To this end, we conducted experiments on the 21 grid configurations in the following experiments. To better demonstrate the applicability of our theory, we not only visualize the generated results but also provide quantitative results, both of which lend strong support to our theory.

Data and model. Let $e_1 = (1, 0)$ and $e_2 = (\cos\theta, \sin\theta)$, with unit edge length. The lower-left layouts consist of $x_{ij} = ie_1 + je_2$, with $i, j \in \{0, 1, 2\}$ for nine points or $i, j \in \{0, 1\}$ for four points. The centered nine-point layout subtracts $e_1 + e_2$ from each point. For each layout, $\theta \in \{30^\circ, 45^\circ, 60^\circ, 90^\circ, 120^\circ, 135^\circ, 150^\circ\}$, giving 21 datasets.

The models use $p = 4000$ fixed Gaussian random features with ReLU activation and a Fourier time embedding with $K = 128$ frequencies. The forward coefficients are $a_t = e^{-t}$ and $h_t = 1 - e^{-2t}$. The supplied final samples use the learned reverse OU drift $y + 2\widehat{s}(t, y)$, where $\widehat{s} = (m_A - x)/h_t$, and Euler–Maruyama with 2000 geometric steps from $T = 10$ to $\delta = 10^{-3}$. Each final-sample panel contains 10,000 samples. The experiments applies a final denoising step $y \mapsto m_A(\delta, y)/a_\delta$ to make the visualization more clear.

For consistency, the residual notation below is $e_A = m_A - m$, with $m(t, x) = \mathbb{E}[a_t X_0 \mid X_t = x]$. The experiment record labels the two training states by $\tau = 5000$ and $\tau = \infty$.

Generated configurations. Figures 20–21 show the existing endpoint samples for all configurations. The observed patterns depend on angle, layout, and training state, and include concentration near data points as well as structured intermediate generation.

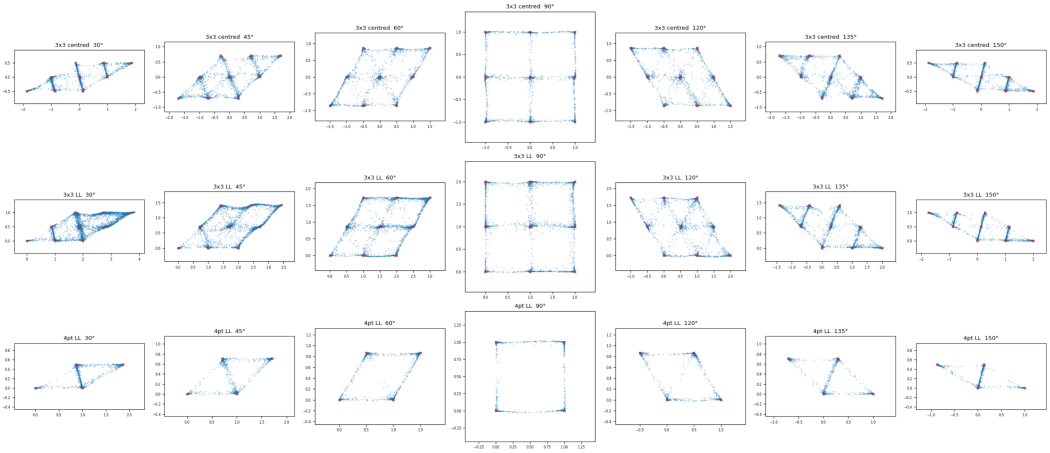


Figure 20: **Lattice generation at the supplied training state $\tau = 5000$.** Rows: centered 3×3, lower-left 3×3, and lower-left four-point layouts. Columns: $\theta = 30°, 45°, 60°, 90°, 120°, 135°, 150°$. Blue points are the 10,000 final, denoised samples per panel; red crosses are training samples.

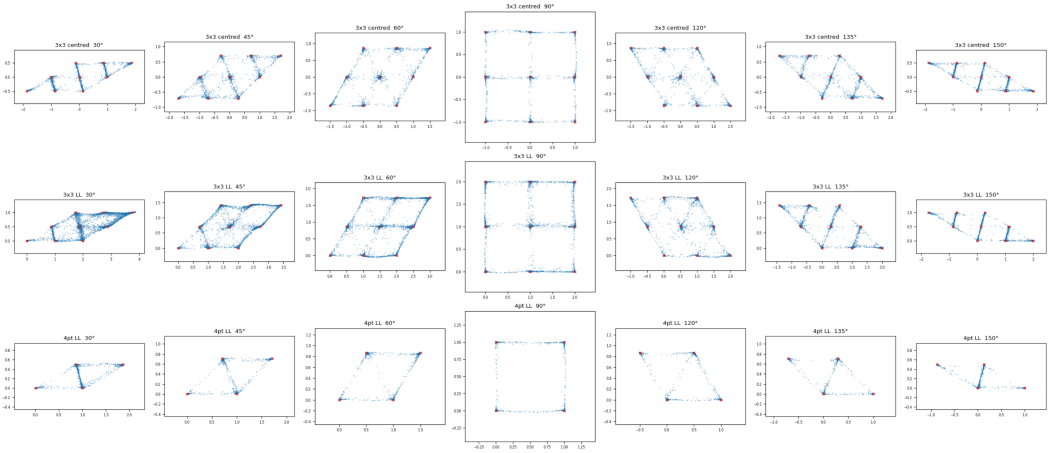


Figure 21: **Lattice generation at the supplied training state labeled $\tau = \infty$.** Layouts, angles, and sample counts match Figure 20.

Reference ridge geometry. At each positive diffusion time, the reference distribution is the equal-weight Gaussian mixture with centers $a_t x_{ij}$ and covariance $h_t I$. Figures 22–23 reproduce its supplied one-dimensional ridge visualizations at $t = 0.1, 0.04, 0.016, 0.006$, displayed in data coordinates. Gray connecting segments are visual guides and are not assertions that the entire segments belong to the ridge. The extracted structures vary with angle and noise level; comparison with the generated samples supports a qualitative geometric correspondence.

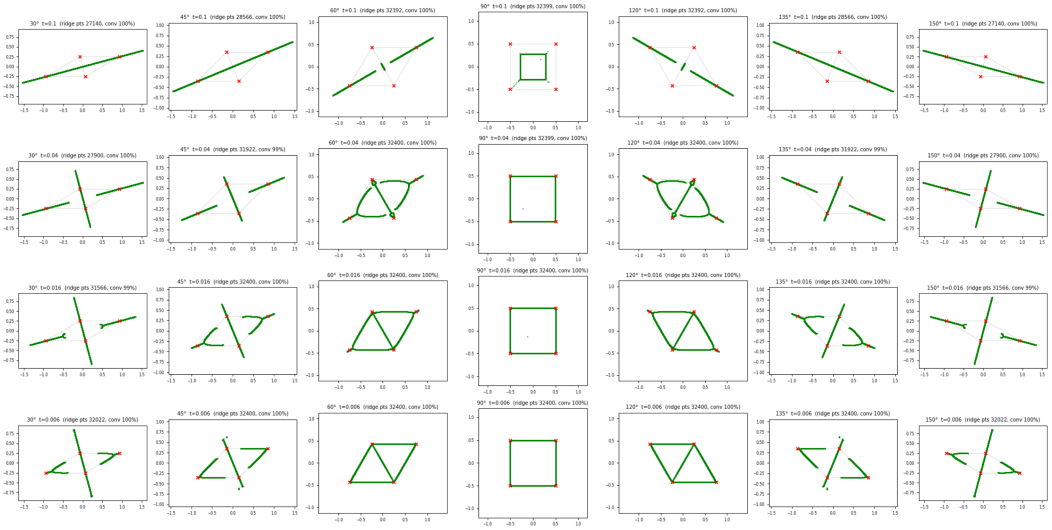


Figure 22: **Supplied one-dimensional ridge visualizations for the four-point layouts.** Rows: $t = 0.1, 0.04, 0.016, 0.006$; columns: the seven lattice angles in Figure 20. Green curves are the numerically extracted reference ridges, red crosses are training samples, and gray segments connect neighboring points.

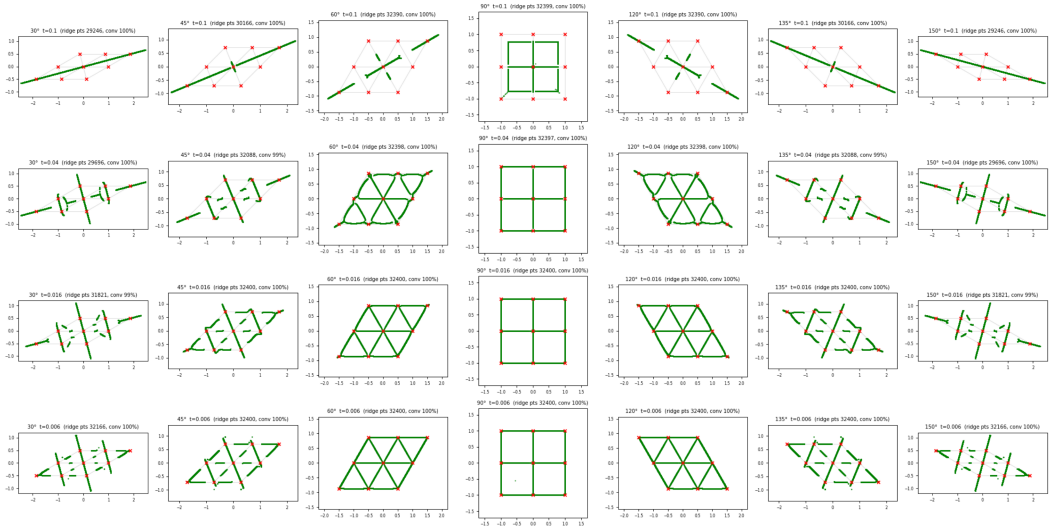


Figure 23: **Supplied one-dimensional ridge visualizations for the nine-point layouts.** Rows and columns match Figure 22. Note that changing the positions does not change the overall relative geometric relationships; consequently, the two different 3×3 grid configurations share the same ridge geometry.

Segment-based directional diagnostics. For each evaluated point x, the experiment selects a nearest straight segment joining neighboring lattice points. If u is its unit tangent, define

$$Q^{\parallel}(x) = uu^{\top}, \qquad Q^{\perp}(x) = I - uu^{\top}.$$

This is because, when t is small, the ridge lies very close to the line segment connecting two neighboring points (as Figure 22, 23 shows). We can therefore use the direction of this segment as the tangent direction and its perpendicular direction as the normal direction, without having to compute the Hessian matrix explicitly.

To examine how the error depends on direction, we plot the training and test errors along the tangential and normal directions over time for the 3×3, 30° lattice with its lower-left corner at the origin ($\tau = \infty$), as shown in Figure 24. Both errors are computed from the learned m_A and m through the residual $e_A = m_A - m$, projected onto the tangential and normal directions by $P^{\parallel}$ and $P^{\perp}$, respectively. The training error is averaged under the forward distribution p_t over the tube neighborhood of the ridge, and is given by $h_t^{-2}\mathbb{E}|Pe_A|^2$ for $P \in \{P^{\parallel}, P^{\perp}\}$. The test error is instead given by $\mathbb{E}|Pe_A|^2$, averaged along $40{,}000$ generated trajectories.

In the displayed late-time regime the tangential component is substantially larger than the normal component, and the imbalance is more pronounced under the generated states. This is consistent with the directional explanation in Section 4. The interpolation-like behavior we observe is in fact caused by the error along the tangential direction being much larger than that along the normal direction.

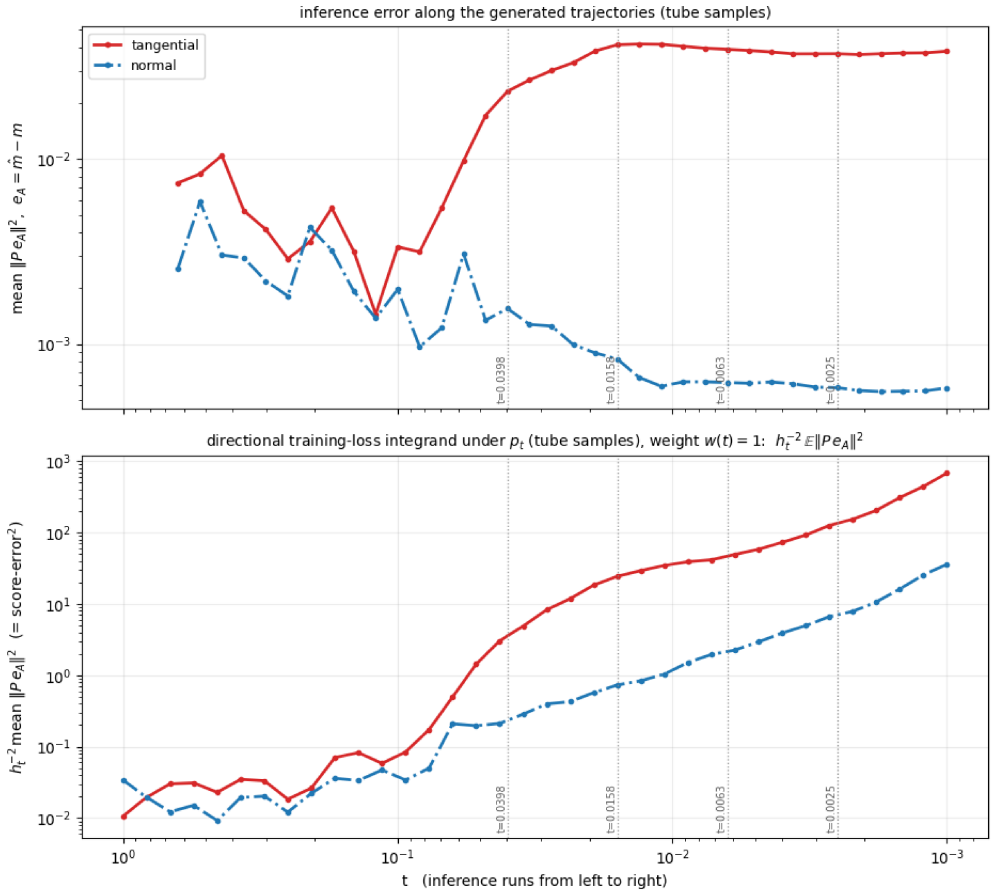


Figure 24: **Directional residuals for the lower-left 3×3, 30° lattice.** The supplied training state is labeled $\tau = \infty$. Top: segment-projected posterior-mean mean squared errors along 40,000 generated trajectories. Bottom: the corresponding h_t^{-2}-weighted diagnostic under the forward law, labeled $w(t) = 1$ in the source.

Comparison across configurations. Figure 25 compares the ratio of tangential to normal error across all 21 configurations. The forward-distribution diagnostic is plotted across noise levels (corresponding to training error), and the generated diagnostic is reported at $t = 0.0398, 0.0158, 0.0063, 0.0025$ (corresponding to test error). The late-time ratios illustrate the directional imbalance in these experiments.

As can be seen, the ratio of tangential to normal error exceeds 10 at small t for both the training error and the test error, indicating a significant imbalance: the tangential error is at least one order of

magnitude larger than the normal error. This is consistent with our experimental observations, where the generated samples concentrate near the ridge rather than around any particular true data center.

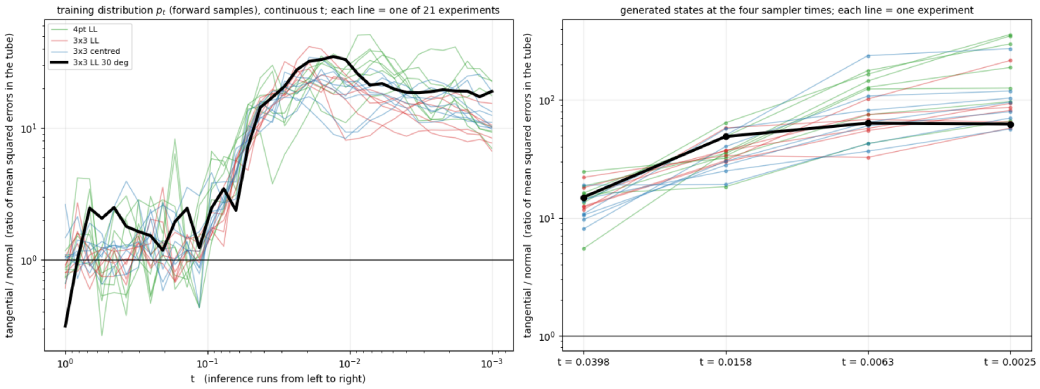


Figure 25: **Tangential-to-normal mean squared residual ratios for the 21 lattice configurations.** Each thin curve is a configuration; the thick curve is the lower-left 3×3, 30° case. Left: forward-distribution diagnostics across diffusion time. Right: generated-state diagnostics at the four reported times. The horizontal reference line is ratio one.

Dependence on translation. Figure 26 compares the lower-left and centered 3×3 lattices at 30°, for the readouts labeled $\tau = \infty$. These datasets have the same relative geometry but different positions with respect to the origin. The lower-left configuration displays more dispersed intermediate samples toward the upper-right part of the lattice. Translation of the reference data geometry therefore does not, in these fitted models, imply identical learned generation after translation back.

To better understand this phenomenon, we visualize the normal and tangential errors along the longest diagonal of the lattice. Specifically, we introduce a one-dimensional coordinate x_L by projecting sample locations onto this diagonal, measured from the lower-left corner to the upper-right corner of each lattice. Both configurations share the same coordinate system, allowing us to directly compare their errors at corresponding lattice locations.

The results are shown in Figure 27. The first two rows of the figure show the normal and tangential errors as functions of x_L at different sampling times, and the bottom row shows the distribution of generated samples projected onto the same coordinate.

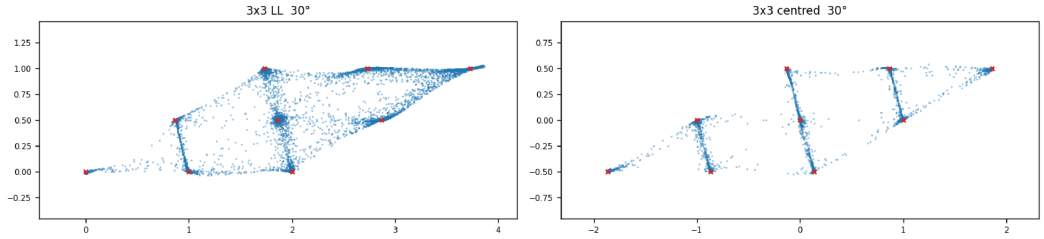


Figure 26: **Translation comparison for the 3×3, 30° lattice.** Left: lower-left layout. Right: centered layout. Blue points are final generated samples and red crosses are training samples. The panels use the supplied readouts labeled $\tau = \infty$. The relative data geometry is unchanged, but the generated configurations differ.

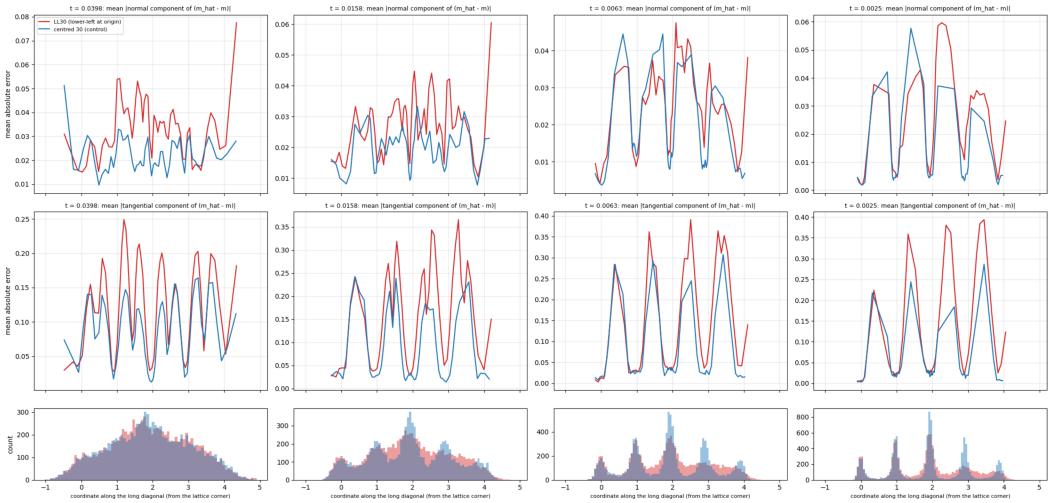


Figure 27: **Location-resolved diagnostics along the longest lattice diagonal.** The horizontal coordinate is measured from the lower-left corner, allowing comparison of corresponding locations. Rows show normal mean squared residual, tangential mean squared residual, and the projected sample distribution. Columns are the sampling times shown in the supplied plot. The projectors are segment based. Differences in error and generated occupancy are reported descriptively, not as a controlled causal attribution.

Interpretation. We observe that the LL configuration tends to produce more samples toward the right side of the lattice, especially at later sampling times. More importantly, while the errors of the two configurations are relatively similar on the left side, their differences become much more pronounced on the right side, where the LL configuration exhibits substantially larger errors in both the normal and tangential directions. This spatial asymmetry is consistent with the observed generation results, suggesting that the increased errors in regions farther from the origin are associated with the greater dispersion of generated samples between lattice points.

NeurIPS Paper Checklist—Working Draft

The questions below follow the submitted checklist. Bracketed author checks are deliberately unresolved rather than replaced by unsupported declarations. Complete them using the final source, code, and asset records, before submission.

1. **Claims.** Do the main claims in the abstract and introduction accurately reflect the paper's contributions and scope?
   Answer: [Yes].
   Justification: The abstract and Section 1 describe the paper's data-dependent geometric characterization of diffusion-model generation. Sections 3 and 4 develop the reach–align–slide analysis and its connection to directional training errors, including the RFNN example. The claims concern generation relative to geometry induced by the finite training set, rather than recovery of an unknown population distribution or guarantees of semantic sample quality.

2. **Limitations.** Does the paper discuss limitations?
   Answer: [Yes].
   Justification: Section 6 discusses the limitations of the work, including the continuous-time inference analysis and the scope of the explicit RFNN training analysis. The geometric assumptions and the local conditions for tangential analysis are stated in Section 3 and the corresponding appendices. Appendix J.5 discusses numerical limitations of ridge projection when the Hessian eigengap is small.

3. **Theory assumptions and proofs.** For each theoretical result, are all assumptions and a complete, correct proof provided?
   Answer: [Yes].
   Justification: The main theoretical results are stated in Sections 3 and 4, with their assumptions stated or referenced in the corresponding results. Appendix B derives the denoising mean-matching objective. Appendices C and D develop the geometric properties used in the analysis; Appendices E–G provide the reach, normal-alignment, and tangential-sliding arguments; Appendix H connects inference errors to training losses; and Appendix I provides the RFNN and gradient-descent analysis.

4. **Experimental reproducibility.** Is the information needed to reproduce the main results disclosed?
   Answer: [Yes].
   Justification: Section 5 describes the main experimental questions and settings. Appendices J and K provide the model architectures, training objectives, optimization settings, sampling procedures, and geometric evaluation methods used in the experiments. Appendix J.4 includes the ridge-projection algorithm, and Appendix J.5 reports its numerical stability diagnostics.

5. **Open access to data and code.** Is code and data access provided with sufficient instructions?
   Answer: [Yes].
   Justification: The experimental code is included with the supplementary materials. The synthetic data configurations are specified in the paper, and the image experiments use MNIST. The experimental procedures and settings are described in Appendices J and K.

6. **Experimental settings.** Are all training and test details specified?
   Answer: [Yes].
   Justification: Section 5 summarizes the experimental settings, and Appendices J and K provide the detailed architectures, optimizers, learning rates, batch sizes where applicable, training durations, loss-weight schedules, sampling time grids, and geometric evaluation procedures. The experiments distinguish directional training losses from geometric quantities evaluated along generated trajectories.

7. **Statistical significance.** Are error bars or other significance information reported appropriately?
   Answer: [No].
   Justification: The experiments illustrate the geometric mechanisms studied in the theory through generated-sample visualizations, directional errors, and inference trajectories. They do not report confidence intervals or statistical significance tests across independent training

runs. The reported comparisons are therefore descriptive rather than claims of statistically significant performance improvements.

8. **Compute resources.** Are compute type, memory, and execution time disclosed?
   Answer: [Yes].
   Justification: Appendices J and K report the computational resources and approximate execution times for the submitted experiments. Appendix J.1 specifies the NVIDIA RTX 4090 GPU with 24.0 GB of memory, the 14-core AMD EPYC 7453 CPU, and 60.1 GB of system memory. The experimental descriptions report runtimes for RFNN training, the additional synthetic experiments, MNIST training and geometric evaluation, and the two-point MLP experiments.

9. **Code of ethics.** Does the research conform to the NeurIPS Code of Ethics?
   Answer: [Yes].
   Justification: The research conforms to the NeurIPS Code of Ethics.

10. **Broader impacts.** Does the paper discuss positive and negative societal impacts?
    Answer: [N/A].
    Justification: This is foundational theoretical work on diffusion-model generalization, rather than an application or deployment study. The paper includes an Impact Statement and does not identify a specific societal consequence requiring separate discussion.

11. **Safeguards.** Are safeguards described for releases with high misuse risk?
    Answer: [N/A].
    Justification: The work does not introduce a release of high-risk pretrained models or scraped datasets. The experiments study controlled synthetic configurations and MNIST, and no release requiring the safeguards described in this question is part of the paper.

12. **Licenses for existing assets.** Are existing assets credited and their terms respected?
    Answer: [No].
    Justification: The paper identifies MNIST as the dataset used in the image experiments and PyTorch as software used in the implementation. However, it does not explicitly document the applicable licenses and terms of use for all existing assets. The submission therefore does not fully satisfy the asset-attribution and licensing-disclosure requirements of this question.

13. **New assets.** Are newly introduced assets documented?
    Answer: [No].
    Justification: Experimental code is included with the supplementary materials, and its experimental use is described in Appendices J and K. However, the paper does not specify the asset-level documentation accompanying that code, including its license and release information.

14. **Crowdsourcing and human subjects.** Are instructions, compensation, and related details provided where applicable?
    Answer: [N/A].
    Justification: The research does not involve crowdsourcing experiments or a new study with human participants.

15. **IRB approvals.** Are approvals or equivalent reviews described for human-subjects research?
    Answer: [N/A].
    Justification: The research does not involve a study with human participants requiring Institutional Review Board approval or an equivalent review.

16. **LLM usage.** Is an LLM described when it is an important, original, or non-standard component of the core methods?
    Answer: [N/A].
    Justification: The core methods developed in this research do not involve LLMs as important, original, or non-standard components.