PREDICTION–LOSS ALIGNMENT FOR SAMPLER-ROBUST FLOW MATCHING TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent work has popularized a practical recipe in diffusion and flow matching: predict the clean signal x, convert it to a velocity, and train through a velocity-space loss. The conversion contains a singular endpoint amplification and therefore appears prone to unstable optimization, yet recent systems obtain strong empirical results with this recipe. We investigate this tension through the integrability of the pre-optimizer stochastic-gradient second moment. Under stated initialization conditions, the moment diverges under Uniform sampling; boundary-suppressing sampling can restore integrability under an additional upper-growth condition. We then show that prediction–loss alignment eliminates this conversion-induced source of non-integrability. Under a uniform moment bound, alignment yields a finite second moment for every timestep density, including Uniform sampling. Controlled experiments across continuous and binary settings reproduce the predicted sampler-dependent instability and show that aligned objectives remain trainable across the tested samplers. These results reconcile pointwise amplification with sampler-dependent empirical success and support alignment as a principled route to more robust flow-matching training.

1 INTRODUCTION

Recent work has brought renewed attention to clean-data prediction in diffusion and flow matching. "Just image Transformers" (JiT) predicts clean images in pixel space, while Embedded Language Flows (ELF) adopts an analogous recipe in its denoising branch for continuous language embeddings (Li & He, 2025; Hu et al., 2026). JiT and the denoising branch of ELF both predict the clean target x, convert that prediction into a velocity, and apply a velocity-space loss. We refer to this pairing as x-prediction+v-loss.

The potential issue is visible from three elementary equations. Let $X \sim p_{\text{data}}$ and $E \sim \mathcal{N}(0, I)$. Along the linear path from noise to data,

$$Z_t = tX + (1-t)E, \qquad t \in (0,1), \tag{1}$$

the target velocity is $V = X - E$. If a network directly predicts the clean signal $\hat{X}_\theta(Z_t, t)$, its implied velocity is

$$\hat{V}_\theta(Z_t, t) = \frac{\hat{X}_\theta(Z_t, t) - Z_t}{1-t}. \tag{2}$$

Consequently, writing $\ell_x = \left\|\hat{X}_\theta - X\right\|^2$, the velocity loss is

$$\ell_v = \left\|\hat{V}_\theta - V\right\|^2 = \frac{1}{(1-t)^2}\ell_x. \tag{3}$$

Equation (3) places every remaining signal-prediction error under an inverse-square endpoint factor. As $t \to 1$, the corresponding sample gradient is amplified by $(1-t)^{-2}$ and its second moment by $(1-t)^{-4}$. On its face, this recipe appears poorly conditioned: rare samples near the clean endpoint may dominate stochastic updates. Yet JiT and ELF report strong empirical performance with precisely this prediction–loss pairing. This apparent contradiction motivates the central question of our work: *why does this seemingly singular recipe work, and when does it fail?*

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

We investigate this question through complementary theoretical analysis and controlled experiments. We focus the analysis on the integrability of the pre-optimizer stochastic-gradient second moment over training timesteps. The pointwise amplification alone does not determine the outcome. Its integrated effect also depends on how the prediction residual and parameter gradient behave near the endpoint, and on how frequently the timestep sampler exposes that region. We formalize this interaction through a sufficient endpoint criterion for initialized networks.

This analysis provides a mechanism through which empirical success and endpoint amplification can coexist. Under the stated upper-growth conditions, boundary-suppressing samplers make the integrated gradient moment finite; under the initialization lower bounds, Uniform sampling instead gives divergence. Our controlled studies reproduce both sides of this behavior: mismatched training exhibits orders-of-magnitude pre-optimizer gradient amplification and can collapse under Uniform sampling, while boundary-suppressing sampling can recover competitive training. Thus, sampler choice can make the mismatched recipe viable without removing its underlying pointwise amplification.

We further study *prediction–loss alignment*: train an x-predictor directly with an x-space loss and use its implied velocity only for ODE generation. Alignment removes the conversion factor from training. Under the stated uniform gradient-moment conditions, the aligned gradient second moment is integrable for every timestep density, including Uniform sampling. In this precise sense, alignment makes flow-matching training more robust to timestep-sampler choice for the mismatch-induced endpoint failure mode.

Our contributions are:

- We locate the apparent instability of x-prediction+v-loss in a sampler-dependent gradient-moment integrability problem and derive a sufficient endpoint criterion for initialized direct-head and identity-residual networks.
- We explain why boundary-suppressing samplers can conceal mismatch-induced divergence, and show that prediction–loss alignment removes the mismatch-induced endpoint factor from training.
- Controlled continuous and binary experiments exhibit the predicted endpoint amplification together with a sampler-dependent training reversal, while aligned objectives remain trainable across the tested samplers.

Together, these results reconcile the singular conversion with sampler-dependent empirical success and support alignment as a principled route to flow-matching training that is more robust to timestep-sampler choice.

2 Related Work

Parameterization and weighting. Diffusion objectives have long been understood as noise-level-weighted regression problems, with practical performance depending on preconditioning, parameterization, and loss weighting (Ho et al., 2020; Karras et al., 2022; Kingma & Gao, 2023; Hang et al., 2023). Kingma and Gao further interpret the noise schedule as an importance-sampling distribution for a weighted objective, separating that objective from its Monte Carlo estimator. Flow matching is conventionally formulated as velocity regression (Lipman et al., 2023; Liu et al., 2023). Recent clean-data parameterizations instead predict x directly and may convert that prediction into velocity space for training (Li & He, 2025; Hu et al., 2026). Our work isolates this specific x-prediction+v-loss recipe and characterizes how its terminal pre-optimizer gradient amplification interacts with timestep sampling.

Timestep sampling. Non-uniform noise-level or timestep sampling is an established component of diffusion and rectified-flow training (Karras et al., 2022; Esser et al., 2024). Logit-Normal sampling is used in the Stable Diffusion 3 rectified-flow recipe and subsequently adopted by JiT (Esser et al., 2024; Li & He, 2025). This practice is directly relevant to the empirical puzzle motivating our work: it reduces exposure to the clean endpoint, precisely where converting an x-prediction into velocity space amplifies prediction error. We characterize when this suppression makes the stochastic-gradient second moment integrable and when the underlying endpoint amplification remains exposed.

Discrete and analog-bit modeling. Discrete generative models may operate directly on discrete state spaces through discrete-time transition kernels or continuous-time transition rates (Austin et al., 2021; Lou et al., 2024; Campbell et al., 2024), or use continuous-state representations and continuous flows for categorical endpoints (Chen et al., 2023; Eijkelboom et al., 2024; Dieleman et al., 2022). We use an analog-bit representation from the latter family as a controlled binary setting for testing the same optimization mechanism under continuous Gaussian interpolation. Our contribution concerns prediction–loss conversion and endpoint exposure within this continuous-state representation.

3 Problem Formulation and Assumptions

3.1 Conditional flow-matching setup

We study conditional flow matching, which learns a time-dependent velocity field that transports a tractable source distribution to a data distribution conditioned on side information. Under a linear Gaussian interpolation, we examine how clean-signal prediction, velocity-space supervision, and timestep sampling interact; Figure 1 summarizes the setup.

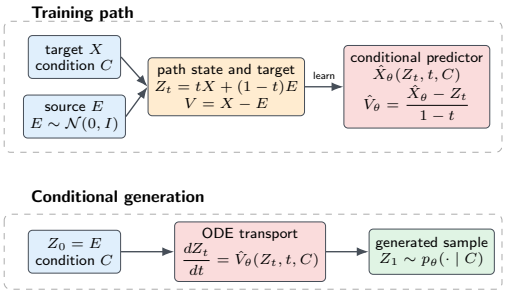

Figure 1: **Conditional flow matching.** Training forms Z_t and V; generation integrates $\hat{V}_\theta(Z_t, t, C)$ from Z_0 to Z_1.

Let (X, C) denote the target data and its optional conditioning variable, with $X \in \mathbb{R}^D$ and $\mathbb{E}\,\|X\|^2 < \infty$. Let $E \sim \mathcal{N}(0, I)$ be independent of (X, C). For a fixed $t \in (0, 1)$, the linear interpolation and its samplewise velocity target are

$$Z_t = tX + (1-t)E, \qquad V = X - E. \quad (4)$$

Training regresses the conditional velocity along this path; generation or conditional inference integrates the learned velocity field from the Gaussian source while holding C fixed. Here t denotes a deterministic path location. We reserve T for a random training timestep whose distribution is specified with the objective below.

For fixed t, all expectations and essential suprema are over the joint law of (X, C, E) and, where applicable, independent model randomness such as dropout. Predictors and conditional means may depend on C; we suppress this argument when it is not needed explicitly.

3.2 Prediction–loss conversion and stochastic optimization

Pointwise conversion. Recall Equations (1)–(3). Fix a parameter state θ, write $r_\theta(t) = \hat{X}_\theta(Z_t, t) - X$ and $\ell_x(t) = \|r_\theta(t)\|^2$, and define $a(t) := (1-t)^{-2}$. Since $a(t)$ is independent of θ,

$$\ell_v(t) = a(t)\ell_x(t), \qquad g_x(t) := \nabla_\theta \ell_x(t), \qquad g_v(t) := \nabla_\theta \ell_v(t) = a(t) g_x(t). \quad (5)$$

Here the sampled input Z_t is not differentiated through, and $g_x(t)$ and $g_v(t)$ are the per-sample parameter gradients before any optimizer-specific transformation.

At every fixed $t < 1$, multiplying the loss by the positive scalar $a(t)$ leaves its unrestricted population minimizer unchanged. Thus, in the idealized infinite-capacity limit, the two formulations target the same conditional predictor; Appendix A.1 verifies that its implied velocity recovers the fixed-time population-optimal flow-matching target. The distinction appears in finite training: one network shares parameters across all timesteps, while $a(t)$ gives errors near $t = 1$ greater weight and multiplies their parameter gradients. The conversion can therefore change how finite model capacity is allocated across timesteps and alter the stochastic-gradient sequence received by the optimizer.

Timestep sampling. At each training step, we sample a data–condition pair (X, C), noise E, and a timestep T from a density q on $(0, 1)$, then form $Z_T = TX + (1-T)E$ and evaluate the selected loss. Equivalently, for a small interval around t,

$$T \sim q, \qquad \Pr(T \in [t, t + \mathrm{d}t]) \approx q(t)\,\mathrm{d}t. \quad (6)$$

Thus $q(t)$ controls how frequently training visits each timestep region.

Optimizer-facing second moment. The random parameter gradients presented to the optimizer and the conditional aligned-gradient moment are

$$\hat{g}_x = g_x(T), \qquad \hat{g}_v = a(T)g_x(T), \qquad G_\theta(t) := \mathbb{E}\left\|g_x(t)\right\|^2. \tag{7}$$

Conditional on sampling near t, the aligned and mismatched gradient second moments are $G_\theta(t)$ and $a(t)^2 G_\theta(t)$, respectively. Their local contributions to the timestep-integrated second moment are therefore

$$\mathrm{d}M_x(t) = q(t)G_\theta(t)\,\mathrm{d}t, \qquad \mathrm{d}M_v(t) = q(t)a(t)^2 G_\theta(t)\,\mathrm{d}t. \tag{8}$$

Integrating these local contributions gives the exact total moments

$$M_x(q) = \mathbb{E}\left\|\hat{g}_x\right\|^2 = \int_0^1 q(t)G_\theta(t)dt, \qquad M_v(q) = \mathbb{E}\left\|\hat{g}_v\right\|^2 = \int_0^1 q(t)a(t)^2 G_\theta(t)dt. \tag{9}$$

The total second moment characterizes the aggregate magnitude of raw stochastic gradients over timestep sampling. Its divergence identifies a failure of finite-second-moment regularity, with rare endpoint samples contributing increasingly large gradients. For Uniform sampling, $q(t) = 1$ and mismatch inserts the factor $a(t)^2$ directly. A boundary-suppressing sampler can instead make the integral finite by reducing endpoint exposure, without removing the pointwise conversion factor.

3.3 Initialization regimes

Our primary question is how prediction–loss mismatch affects a flow-matching network trained from scratch. We therefore begin at the initialized parameter state and characterize the pre-optimizer gradient before training changes the prediction residual. This requires making the initialized prediction head explicit. Let θ_0 denote the initialized parameters,

$$r_0(t) := \hat{X}_{\theta_0}(Z_t, t) - X, \qquad J_0(t) := \nabla_\theta \hat{X}_{\theta_0}(Z_t, t). \tag{10}$$

Assumption 3.1 (Initialization regimes)**.** At the fixed initialized state θ_0, one of the following regimes holds throughout $t \in (t_0, 1)$ for some $t_0 \in (0, 1)$ and $c_J > 0$:

Regime A (direct head). The predictor has no end-to-end identity skip and $\mathbb{E}\left\|J_0(t)^\top r_0(t)\right\|^2 \geq c_J$.

Regime B (identity residual). The residual branch is zero-initialized, so $\hat{X}_{\theta_0}(Z_t, t) = Z_t$, and $\mathbb{E}\left\|J_0(t)^\top r_0(t)/(1-t)\right\|^2 \geq c_J$.

The two regimes imply the following lower bounds on the initialized aligned-gradient moment:

$$G_{\theta_0}(t) \geq \begin{cases} 4c_J, & \text{Regime A,} \\ 4c_J(1-t)^2, & \text{Regime B.} \end{cases} \tag{11}$$

Proof sketch. Since $G_{\theta_0}(t) = 4\mathbb{E}\left\|J_0(t)^\top r_0(t)\right\|^2$, Regime A follows immediately. In Regime B,

$$r_0(t) = Z_t - X = tX + (1-t)E - X = (1-t)(E - X), \tag{12}$$

$$G_{\theta_0}(t) = 4(1-t)^2\mathbb{E}\left\|J_0(t)^\top (E - X)\right\|^2 \geq 4c_J(1-t)^2. \tag{13}$$

This proves Equation (11). The conditions concern only the pre-optimizer gradient at initialization.

A checkable output-head condition. An additive output bias provides a sufficient condition without requiring every network direction to be nondegenerate. Write its contribution as Sb, where b is a trainable bias vector and S is its fixed replication map across output coordinates. The bias block alone gives

$$G_{\theta_0}(t) \geq \mathbb{E}\left\|\nabla_b \ell_x(t)\right\|^2 = 4\mathbb{E}\left\|S^\top r_0(t)\right\|^2. \tag{14}$$

For a zero-initialized direct output, this is the time-independent lower bound $4\mathbb{E}\left\|S^\top X\right\|^2$, positive whenever the projected data are nonzero with positive probability. In the identity-residual case, independent Gaussian noise makes $\mathbb{E}\left\|S^\top (E - X)\right\|^2 > 0$ for every nonzero S. The JiT implementation uses a zero-initialized patch output layer with a shared bias, so this check concerns sums of target patches. Appendix A.2 derives both cases and states the corresponding endpoint condition for the nonzero-initialized U-Net head.

4 TERMINAL AMPLIFICATION

4.1 SAMPLER-DEPENDENT DIVERGENCE AT INITIALIZATION

Substituting Equation (11) into the exact moment identity (9) gives the endpoint criterion below.

Theorem 4.1 (Sampler-dependent initialization-time divergence)**.** *Suppose Assumption 3.1 holds and training timesteps have density q. Then, for every $b \in (t_0, 1)$,*

$$M_v(q) = \int_0^1 q(t)(1-t)^{-4} G_{\theta_0}(t)dt \geq 4c_J \int_{t_0}^b q(t)(1-t)^{-p}dt, \tag{15}$$

where $p = 4$ in Regime A and $p = 2$ in Regime B. More explicitly, suppose that for some $t_1 \in [t_0, 1)$, $c_q > 0$, and $\beta \geq 0$,

$$q(t) \geq c_q(1-t)^\beta \quad \text{for a.e. } t \in [t_1, 1). \tag{16}$$

If $\beta \leq p - 1$, then $M_v(q) = \infty$. Hence Uniform sampling, for which $q_{\mathrm{U}}(t) = 1$ and $\beta = 0$, diverges in both regimes.

The proof is in Appendix A.3. The thresholds are $\beta \leq 3$ in Regime A and $\beta \leq 1$ in Regime B, obtained from lower bounds on the conditional mismatch moment of orders $(1-t)^{-4}$ and $(1-t)^{-2}$, respectively. These are sufficient divergence conditions; larger β alone does not establish integrability. At a later parameter state, Equation (9) remains exact, but integrability depends on the learned $G_\theta(t)$.

4.2 ALIGNMENT AND ENDPOINT-SUPPRESSING SAMPLING

Prediction–loss alignment. For an x-prediction head, alignment removes the conversion weight from training: $\ell_x(t) = (1-t)^2 \ell_v(t)$ and

$$M_x(q) = \int_0^1 q(t) G_\theta(t)dt. \tag{17}$$

Proposition 4.2 (Aligned objective)**.** *If $C_G(\theta) := \operatorname{ess\,sup}_{t\in(0,1)} G_\theta(t) < \infty$, then for every timestep density q,*

$$M_x(q) \leq C_G(\theta) < \infty. \tag{18}$$

The bound follows immediately from $\int_0^1 q(t)dt = 1$. Appendix A.4 gives sufficient Jacobian–residual moment conditions and analogous bounds for aligned velocity MSE and BCE-with-logits. The implied velocity in Equation (2) is still used at inference.

Logit-Normal endpoint suppression. Let $U \sim \mathcal{N}(m, s^2)$ with $m \in \mathbb{R}$ and $s > 0$, and set $T = \sigma(U) = (1 + e^{-U})^{-1}$. The induced density on $(0, 1)$ is

$$q_{\mathrm{LN}}(t) = \frac{1}{s\sqrt{2\pi}\, t(1-t)} \exp\left[-\frac{(\operatorname{logit}(t) - m)^2}{2s^2}\right]. \tag{19}$$

Proposition 4.3 (Logit-Normal integrability under polynomial endpoint growth)**.** *Fix θ. Suppose that for some $t_0 \in (0, 1)$, $C < \infty$, and $p \geq 0$,*

$$G_\theta(t) \leq C(1-t)^{-p}, \qquad t \in [t_0, 1), \tag{20}$$

and $\int_0^{t_0} q_{\mathrm{LN}}(t)(1-t)^{-4} G_\theta(t)dt < \infty$. Then $M_v(q_{\mathrm{LN}}) < \infty$.

Proof sketch. The logistic change of variables $t = \sigma(u)$ gives

$$\begin{aligned} &\int_{t_0}^1 q_{\mathrm{LN}}(t)(1-t)^{-4} G_\theta(t)dt \\ &\quad \leq C\, \mathbb{E}\left[(1-\sigma(U))^{-(p+4)} \mathbf{1}\{U \geq \operatorname{logit}(t_0)\}\right]. \end{aligned} \tag{21}$$

Because $(1-\sigma(U))^{-1} = 1 + e^U$ and the Gaussian moment-generating function satisfies $\mathbb{E}[e^{rU}] = e^{rm + r^2 s^2/2} < \infty$ for every finite r, the right-hand side is finite. Adding the assumed nonterminal integral proves the proposition. In particular, $C_G(\theta) < \infty$ implies

$$M_v(q_{\mathrm{LN}}) \leq C_G(\theta)\, \mathbb{E}[(1-\sigma(U))^{-4}] < \infty. \tag{22}$$

The full proof is in Appendix A.5. Alignment removes the mismatch factor pointwise; Logit-Normal sampling retains it but can make its integrated contribution finite by suppressing the endpoint.

5 EXPERIMENTS

Guided by our theoretical analysis, we organize the experiments around two questions. **(i) Endpoint amplification and training quality:** Does prediction–loss mismatch produce the endpoint gradient amplification predicted by the theory, and how does this amplification manifest in model fitting and generation quality? Section 5.1 examines these questions through direct gradient measurements and controlled training interventions. **(ii) Alignment for sampler-robust training:** Does prediction–loss alignment support effective training under both Uniform and Logit-Normal sampling, and can Logit-Normal sampling mitigate the training failure observed under mismatch? Section 5.2 evaluates both effects through objective–sampler comparisons. Full experimental protocols and supplementary MIMO results are provided in Appendices B and B.7.

5.1 ENDPOINT EXPOSURE AND GRADIENT AMPLIFICATION

We first tested the predicted growth of gradient second moments as the timestep approached the endpoint at initialization. We then evaluated how restricting the corresponding endpoint weighting affected generation quality.

Direct gradient measurements at initialization. To test the two initialization regimes, we used a two-hidden-layer gated MLP with a zero-initialized output layer as a direct signal predictor (Regime A). Adding an identity skip from the input gave a residual predictor (Regime B), so the two networks initially produced 0 and Z_t, respectively. Both used paired branch parameters and identical data–noise pairs; implementation details are provided in Appendix B.1.

For each frozen network, we independently backpropagated the aligned and mismatched losses to measure individual-example gradient second moments. Table 1 shows that mismatched moments grew sharply as $\epsilon = 1 - t$ decreased in both regimes, whereas aligned moments remained approximately constant in A and decreased in B. In Regime B, the approximately quadratic decay of the aligned moment was insufficient to offset the mismatch multiplier. These measurements supported the predicted endpoint amplification in both initialization regimes and its removal by alignment over the tested endpoint distances.

Table 1: Directly measured gradient second moments at initialization, with mismatch as the tested objective and alignment as the control.

Initialization / loss	$\epsilon = 10^{-2}$	$\epsilon = 10^{-4}$	$\epsilon = 10^{-6}$
A: mismatch	$(2.92 \pm 0.03) \times 10^{7}$	$(2.93 \pm 0.03) \times 10^{15}$	$(2.93 \pm 0.03) \times 10^{23}$
A: aligned	0.292 ± 0.003	0.293 ± 0.003	0.293 ± 0.003
B: mismatch	$(5.93 \pm 0.05) \times 10^{3}$	$(5.95 \pm 0.05) \times 10^{7}$	$(5.95 \pm 0.05) \times 10^{11}$
B: aligned	$(5.93 \pm 0.05) \times 10^{-5}$	$(5.95 \pm 0.05) \times 10^{-9}$	$(5.95 \pm 0.05) \times 10^{-13}$

Endpoint weighting and generation quality. Following the initialization-time endpoint measurements, we examined whether restricting the endpoint weighting responsible for mismatch amplification improved generation quality. We trained a conditional U-Net on Binary MNIST under Uniform sampling, capping the inverse-square mismatch weight at C and normalizing it to unit mean. Smaller caps imposed stronger restrictions on endpoint weighting, while the normalization held its mean scale fixed (Appendix B.5).

Table 2: Binary-MNIST endpoint-cap sweep. Weight moments are analytical; FID-50K is mean $\pm$ s.d. over three seeds.

Cap C	$\mathbb{E}\widetilde{w}_C^2$	FID-50K
10^2	3.69	16.02 ± 2.13
10^4	33.67	27.18 ± 4.25
10^6	333.67	614.37 ± 10.10
10^8	3333.67	1625.48 ± 70.82

Table 2 shows that increasing C from 10^2 to 10^8 raised the weight second moment approximately 10^3-fold and worsened FID from 16.02 to 1625.48. Stronger restrictions therefore yielded substantially better generation, even though all tested weights were bounded. This supported restricting endpoint weighting as an effective intervention in this setting: the strength of the retained weighting mattered for generation quality beyond its numerical

finiteness. The cap jointly changed the strength of endpoint weighting and the time profile of supervision.

5.2 PREDICTION–LOSS ALIGNMENT FOR SAMPLER-ROBUST TRAINING

To test the predicted distinction between suppressing endpoint exposure and removing conversion-induced amplification, we compared timestep sampling and prediction–loss alignment in three settings. A Gaussian MLP examined the sampler-dependent gradient behavior of the mismatched objective. Conditional U-Net and JiT-B/4 experiments then tested whether this sensitivity appeared in image generation, and whether alignment supported effective training under both Uniform and Logit-Normal sampling. Together, these comparisons evaluated the practical claim of sampler-robust training across binary and continuous data.

Sampler control. To examine whether sampling moderated the gradients produced by the singular conversion, we held the Gaussian MLP and raw mismatched objective fixed and paired initialization and data–noise streams across samplers. Median gradient norms were of order 10^3 under Uniform and 10^{-1} under Logit-Normal, with much larger Uniform excursions (Figure 2). This contrast supported the predicted role of endpoint exposure: changing how often terminal timesteps were sampled substantially moderated the observed gradients while retaining the mismatched conversion. Appendix B.2 gives the protocol.

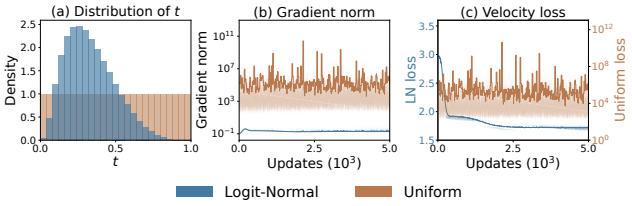

Figure 2: Gaussian sampler control. (a) Sampler densities. (b,c) Raw trajectories (faint) and 25-update means (solid) across three paired seeds; loss axes are sampler-specific.

Effect of timestep sampling. To test whether sampler sensitivity also appeared in generation quality, we evaluated a conditional U-Net (Ronneberger et al., 2015) on Binary MNIST (LeCun et al., 1998) and JiT-B/4 (Li & He, 2025) on Tiny-ImageNet. Each within-backbone comparison changed the sampler while retaining the prediction–loss pairing and evaluation protocol. Under mismatched x-prediction+v-MSE, replacing Uniform with Logit-Normal sampling reduced FID-50K (Heusel et al., 2017) from 1269.53 ± 222.57 to 11.02 ± 1.80 for U-Net and from 390.50 to 14.56 for JiT.

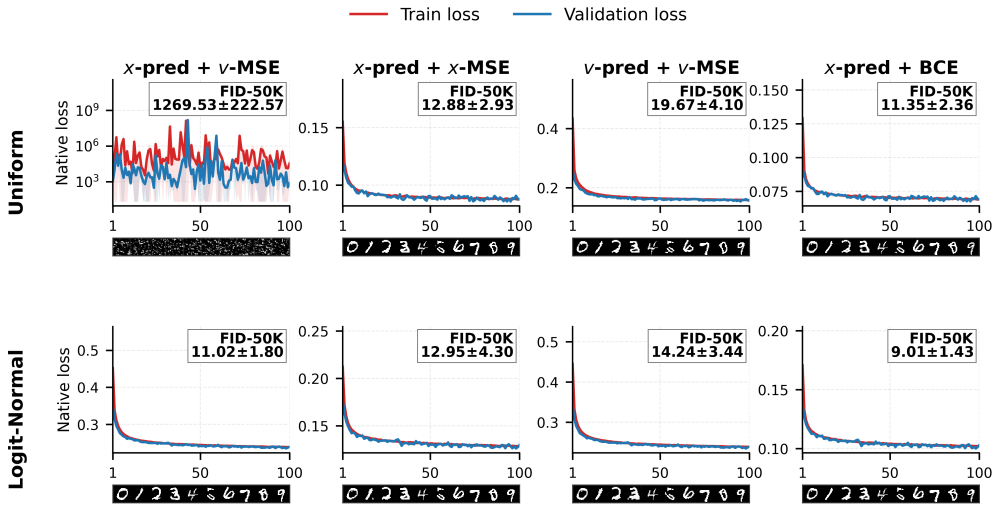

Figure 3: Binary-MNIST objective–sampler matrix: FID-50K and curves summarize three training seeds (mean ± s.d.), with samples below. Rows vary samplers; columns vary prediction–loss pairings. Native losses are comparable only within each objective.

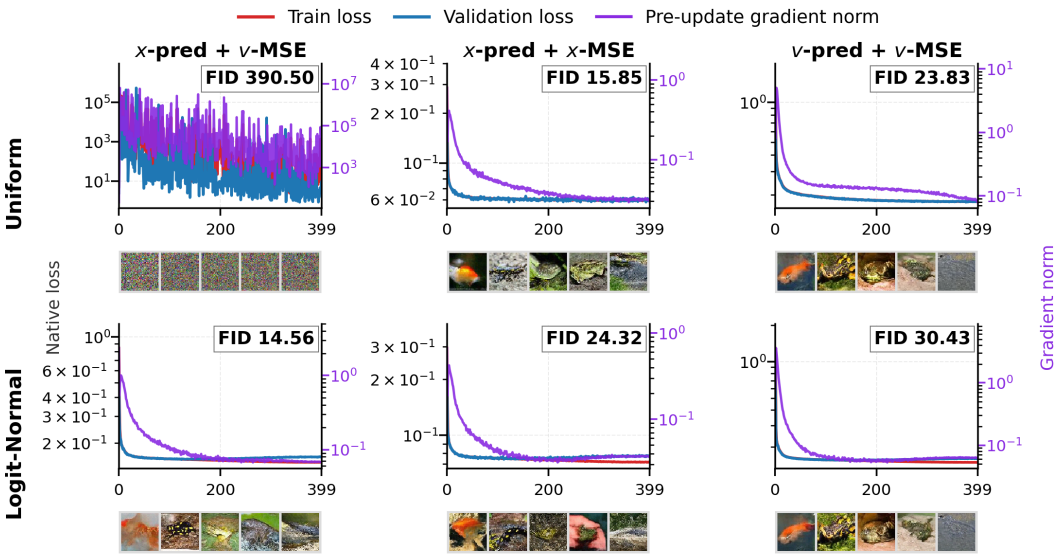

Figure 4: JiT-B/4 objective–sampler matrix. Each cell reports native train/validation losses, mean pre-update full-parameter gradient norm, FID-50K, and the first row of its epoch-400 sample grid. Each cell is one run. Native loss values are comparable within, not across, objectives.

Figures 3 and 4 show generation collapse under Uniform and recognizable samples under Logit-Normal, together with the corresponding loss trajectories. The improvement in both backbones supported boundary-suppressing sampling as an effective remedy for the observed mismatch-related training difficulties. It also illustrated how a recipe with strong endpoint amplification could nevertheless achieve good generation quality under a suitable sampler. The JiT comparison retained the same small denominator floor in both sampler cells. Full protocols are given in Appendices B.3 and B.6.

Effect of prediction–loss alignment. To test alignment as a remedy that did not rely on boundary-suppressing sampling, we replaced velocity MSE with signal MSE while retaining the direct x-prediction backbone under each sampler. Aligned U-Net obtained FID 12.88 ± 2.93 under Uniform and 12.95 ± 4.30 under Logit-Normal sampling. Aligned JiT obtained 15.85 and 24.32, respectively. Both aligned configurations therefore produced useful samples under Uniform as well as Logit-Normal sampling, in contrast to the strong sampler dependence of mismatch. The complete objective–sampler matrices also included direct v-MSE and, for U-Net, aligned BCE; both alternatives remained trainable under the two samplers. These within-backbone comparisons supported alignment's reduced sensitivity to sampler replacement across the evaluated continuous and binary generation tasks.

Generation quality and sampler robustness. JiT achieved its best observed FID with mismatch plus Logit-Normal sampling. Alignment's advantage was effective generation under both tested samplers, whereas mismatch depended strongly on sampler choice. Taken together, these results supported the practical benefit of removing the conversion-induced amplification: effective training was maintained across the two tested timestep distributions. The lower FID of mismatch plus Logit-Normal further showed that best-case generation quality and robustness to sampler replacement were distinct aspects of the training recipe.

Supplementary training diagnostics. The paired JiT early-training comparison retained the initialization and data–noise–timestep stream while changing the objective. Raw mismatch produced a median pre-update gradient norm of order 10^2, compared with 10^{-1} for alignment, and retained higher signal MSE (Appendix B.6). Frozen trained-checkpoint measurements additionally showed endpoint gradient amplification on the evaluated timestep grid (Appendix B.6). These measurements complemented the initialization analysis by showing that endpoint amplification was also observable during early fitting and at the evaluated trained states, connecting the local gradient diagnostics with the broader training comparisons.

6 Discussion and Limitations

Why a singular recipe can still work. The conversion amplifies gradients at a given timestep; the sampler determines how often training encounters it. Under the growth conditions in Section 4.2, endpoint suppression makes the aggregate second moment finite despite the singular pointwise multiplier. The controlled comparisons illustrated how these effects can coexist: changing the sampler restored useful generation without removing the conversion.

Generation quality and sampler robustness. The JiT results suggest evaluating both the best observed FID and sensitivity to sampler replacement. Alignment removes the conversion multiplier, yielding the uniform-in-sampler moment bound in Proposition 4.2. It retains the prediction architecture and ODE velocity formula, but can change the learned predictor and final sample quality. Its practical value therefore need not coincide with the lowest FID under a particular sampler.

Beyond finite second moments. The endpoint-cap sweep shows why integrability is a starting point rather than a complete account of training quality. All tested caps bounded the conversion weight; at a fixed state with bounded $G_\theta(t)$, this also bounds the gradient second moment. Nevertheless, larger caps produced substantially worse generation. Thus, the strength of the retained endpoint weighting matters in addition to whether it is bounded. Because capping changes both the objective's time profile and the stochastic-gradient sequence, these results motivate examining their joint effects on fitting and generation quality.

Implications for sampler design and reporting. Timestep sampling should be interpreted together with the prediction–loss parameterization. For fixed θ and $T \sim q$ sampled independently of the data and noise, Equation (5) gives

$$\mathbb{E}[\ell_v(T)] = \int_0^1 \frac{q(t)}{(1-t)^2}\,\mathbb{E}[\ell_x(t)]\,dt, \qquad \mathbb{E}[\ell_x(T)] = \int_0^1 q(t)\,\mathbb{E}[\ell_x(t)]\,dt. \tag{23}$$

The outer expectations include timestep sampling; the fixed-t expectations follow Section 3.1. Mismatch therefore carries the signal-space time weight $q(t)a(t)$, whereas alignment retains $q(t)$ alone. Sampling remains $T \sim q$; only the objective weighting changes. We view aligned signal space as a useful common reference for future sampler studies: removing the conversion weight makes the sampler's allocation of signal-space supervision easier to interpret.

In practice, evaluations should report the sampler, endpoint truncation or denominator handling, and pre-clipping gradient diagnostics alongside generation metrics. This distinguishes a favourable sampler–objective combination from robustness to endpoint exposure. With conversion-induced amplification removed, learning-rate selection and optimizer tuning remain part of the training recipe.

Limitations. The divergence theorem is a sufficient result at initialization under the stated output-gradient conditions, rather than a convergence theorem for an optimizer. The frozen diagnostic covers two trained checkpoints and a finite set of inputs and timesteps; it cannot certify trained-state asymptotics. The full JiT matrix has one run per cell, whereas Binary MNIST used three training seeds and a dataset-specific FID feature extractor. Finally, the formal results concern the stated linear Gaussian interpolation. Other paths require analyzing their own prediction-to-velocity conversion.

7 Conclusion

We investigated how x-prediction+v-loss can work well despite its singular endpoint amplification. Under the stated initialization conditions, its stochastic-gradient second moment diverges under Uniform sampling; under the stated growth conditions, boundary-suppressing sampling restores integrability without removing pointwise amplification. Alignment removes the conversion factor and, under a uniform moment bound, yields a finite second moment for every timestep density. Continuous and binary experiments corroborated mismatch's sampler sensitivity and aligned objectives' trainability across the tested samplers. These findings reconcile amplification with empirical success and support alignment as a principled route to sampler-robust training.

AI USE STATEMENT

Generative AI tools were used during revision for language editing, code assistance, mathematical adversarial checking, and organization of experimental diagnostics. The authors independently re-derived the mathematical statements, reviewed all generated text, tested code changes, inspected experimental outputs, and checked cited sources. Generative AI tools were not treated as scientific sources or final arbiters of correctness. The authors take responsibility for all claims, proofs, code, and artifacts in this submission.

REPRODUCIBILITY STATEMENT

Appendix B specifies datasets, architectures, objectives, samplers, seeds, and evaluation procedures. The supplementary material will include the Binary-MNIST harness, JiT diagnostics, and machine-readable per-run summaries.

REFERENCES

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in Neural Information Processing Systems*, 34:17981–17993, 2021.

Andrew Campbell, Jason Yim, Regina Barzilay, Tom Rainforth, and Tommi Jaakkola. Generative flows on discrete state-spaces: Enabling multimodal flows with applications to protein co-design. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 5453–5512. PMLR, 2024. URL https://proceedings.mlr.press/v235/campbell24a.html.

Ting Chen, Ruixiang Zhang, and Geoffrey Hinton. Analog bits: Generating discrete data using diffusion models with self-conditioning. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=3itjR9QxFw.

Sander Dieleman, Laurent Sartran, Arman Roshannai, Nikolay Savinov, Yaroslav Ganin, Pierre H Richemond, Arnaud Doucet, Robin Strudel, Chris Dyer, Conor Durkan, Curtis Hawthorne, Rémi Leblond, Will Grathwohl, and Jonas Adler. Continuous diffusion for categorical data. *arXiv preprint arXiv:2211.15089*, 2022.

Floor Eijkelboom, Grigory Bartosh, Christian A Naesseth, Max Welling, and Jan-Willem van de Meent. Variational flow matching for graph generation. *Advances in Neural Information Processing Systems*, 37:11735–11764, 2024. doi: 10.52202/079017-0374.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 12606–12633. PMLR, 2024. URL https://proceedings.mlr.press/v235/esser24a.html.

Tiankai Hang, Shuyang Gu, Chen Li, Jianmin Bao, Dong Chen, Han Hu, Xin Geng, and Baining Guo. Efficient diffusion training via min-SNR weighting strategy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7441–7451, October 2023.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. URL https://arxiv.org/abs/2207.12598.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851, 2020.

Jiadong Hong, Lei Liu, Xinyu Bian, Wenjie Wang, and Zhaoyang Zhang. Soft graph transformer for MIMO detection. In *ICASSP 2026-2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 21421–21425. IEEE, 2026. doi: 10.1109/ICASSP55912.2026.11463763.

Keya Hu, Linlu Qiu, Yiyang Lu, Hanhong Zhao, Tianhong Li, Yoon Kim, Jacob Andreas, and Kaiming He. ELF: Embedded language flows. *arXiv preprint arXiv:2605.10938*, 2026.

Nan Jiang, Jiadong Hong, Lei Liu, Xinyu Bian, Wenjie Wang, and Zhaoyang Zhang. Soft graph diffusion transformer for MIMO detection. *arXiv preprint arXiv:2605.00449*, 2026. doi: 10.48550/arXiv.2605.00449. URL https://arxiv.org/abs/2605.00449.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 26565–26577, 2022. doi: 10.52202/068431-1926.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015. URL https://arxiv.org/abs/1412.6980.

Diederik P. Kingma and Ruiqi Gao. Understanding diffusion objectives as the ELBO with simple data augmentation. In *Advances in Neural Information Processing Systems*, volume 36, pp. 65484–65516, 2023. doi: 10.52202/075280-2858.

Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. CS231N course project report, Stanford University, 2015. URL https://cs231n.stanford.edu/reports/2015/pdfs/yle_project.pdf.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. doi: 10.1109/5.726791.

Tianhong Li and Kaiming He. Back to basics: Let denoising generative models denoise. *arXiv preprint arXiv:2511.13720*, 2025.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations*, 2023.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 32819–32848. PMLR, 2024. URL https://proceedings.mlr.press/v235/lou24a.html.

Anton Obukhov, Maximilian Seitzer, Po-Wei Wu, Semen Zhydenko, Jonathan Kyl, and Elvis Yu-Jing Lin. High-fidelity performance metrics for generative models in PyTorch, 2020. URL https://github.com/toshas/torch-fidelity.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4195–4205, 2023.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241. Springer, 2015.

A Proofs and Additional Analysis

A.1 Fixed-time population target equivalence

Proposition A.1 (Signal regression recovers the fixed-time population flow target). *Fix $t \in (0,1)$ and let $m_t(z,c)$ be any version of $\mathbb{E}[X \mid Z_t = z, C = c]$, with c omitted in the unconditional setting. The unrestricted population minimizer of signal-space MSE is $\hat{X}^*(z,c,t) = m_t(z,c)$, uniquely up to null sets of (Z_t, C) (or Z_t unconditionally). Moreover, a version of the marginal flow-matching velocity is*

$$v^*(z,c,t) := \mathbb{E}[X - E \mid Z_t = z, C = c] = \frac{m_t(z,c) - z}{1-t}, \tag{24}$$

for $P_{(Z_t,C)}$-almost every (z,c). Since $X - E \in L^2$, this conditional expectation is also the unrestricted velocity-MSE minimizer.

Proof. For fixed t, the conditional-expectation orthogonality identity gives, for every measurable predictor f with finite risk,

$$\mathbb{E}\left\|f(Z_t,C,t) - X\right\|^2 = \mathbb{E}\left\|f(Z_t,C,t) - m_t(Z_t,C)\right\|^2 + \mathbb{E}\left\|m_t(Z_t,C) - X\right\|^2, \tag{25}$$

where $m_t(z,c)$ is any version of $\mathbb{E}[X \mid Z_t = z, C = c]$. Hence signal-space MSE is minimized by $f = m_t$, uniquely up to $P_{(Z_t,C)}$-null sets. From $Z_t = tX + (1-t)E$ we also have, conditional on $(Z_t, C) = (z, c)$,

$$X - E = X - \frac{z - tX}{1-t} = \frac{X - z}{1-t}. \tag{26}$$

Taking conditional expectations yields

$$\mathbb{E}[X - E \mid Z_t = z, C = c] = \frac{m_t(z,c) - z}{1-t}, \tag{27}$$

for $P_{(Z_t,C)}$-almost every (z,c), which is exactly the implied velocity of the population-optimal signal predictor. Since $X - E \in L^2$, applying the same orthogonality identity with target $X - E$ shows that this conditional expectation is the unrestricted velocity-MSE minimizer. In the unconditional setting, all occurrences of C and c are omitted. □

A.2 Sufficient conditions from the output bias

The gradient lower bounds in Assumption 3.1 can be checked through a subset of trainable parameters. Suppose the prediction contains an additive bias Sb, where $S \in \mathbb{R}^{D \times d_b}$ is a fixed nonzero matrix and the remaining prediction does not depend on b. For the summed signal loss,

$$\nabla_b \ell_x(t) = 2S^\top r_0(t), \qquad G_{\theta_0}(t) \geq 4\mathbb{E}\left\|S^\top r_0(t)\right\|^2. \tag{28}$$

Sharing a bias across pixels or patches changes S; it does not invalidate this parameter-block lower bound.

Zero-initialized direct output. If $\hat{X}_{\theta_0}(z,t) = 0$ for all inputs, then $r_0(t) = -X$ and

$$G_{\theta_0}(t) \geq 4\mathbb{E}\left\|S^\top X\right\|^2. \tag{29}$$

Thus Regime A holds whenever $\Pr(S^\top X \neq 0) > 0$. For independent per-coordinate biases, $S = I_D$ and the condition is simply $\mathbb{E}\left\|X\right\|^2 > 0$. The JiT implementation used in our diagnostics instead sets both the weights and bias of `final_layer.linear` to zero. This layer predicts a patch vector, and its trainable bias is shared across patches. Writing an image as P target patches $X_1, \ldots, X_P$,

$$S^\top X = \sum_{j=1}^{P} X_j, \qquad G_{\theta_0}(t) \geq 4\mathbb{E}\left\|\sum_{j=1}^{P} X_j\right\|^2. \tag{30}$$

The sufficient nondegeneracy condition is therefore a positive second moment of the summed target patches. Zero output does not imply zero parameter gradient: the output bias remains trainable at initialization.

Zero-initialized identity residual. If $\hat{X}_{\theta_0}(Z_t, t) = Z_t$, then $r_0(t) = (1-t)(E-X)$. Independence and the zero mean of E imply

$$\mathbb{E}\left\|S^\top(E-X)\right\|^2 = \|S\|_{\mathrm{F}}^2 + \mathbb{E}\left\|S^\top X\right\|^2 > 0. \tag{31}$$

Consequently,

$$G_{\theta_0}(t) \geq 4(1-t)^2\big(\|S\|_{\mathrm{F}}^2 + \mathbb{E}\left\|S^\top X\right\|^2\big), \tag{32}$$

which supplies Regime B. This argument concerns a trainable residual output bias; the identity skip alone does not supply a parameter gradient.

Nonzero-initialized direct output. More generally, suppose $r_0(t) \to r_0(1)$ in L^2 as $t \uparrow 1$ and let $d_0 = \mathbb{E}\left\|S^\top r_0(1)\right\|^2 > 0$. Continuity of the L^2 norm gives $\mathbb{E}\left\|S^\top r_0(t)\right\|^2 \geq d_0/2$ on a sufficiently small endpoint neighborhood, yielding Regime A with $c_J = d_0/2$.

The Binary-MNIST U-Net uses a direct convolutional output `out_conv` with one trainable scalar bias shared across its D pixels. It is not zero-initialized. Conditional on all other initial parameters, write its endpoint residual as $r_0(1) = h(X, C) - X + b\mathbf{1}$, with dropout randomness suppressed in the notation for h. The squared projected residual is

$$\mathbb{E}\big[\big(\mathbf{1}^\top(h(X,C) - X) + Db\big)^2\big]. \tag{33}$$

As a function of b, this is a nonnegative quadratic with positive leading coefficient D^2, so it vanishes at at most one value of b. A continuous initialization of b independent of the remaining parameters therefore gives $d_0 > 0$ almost surely. The implemented GroupNorm, convolutions, and SiLU activations are continuous for fixed dropout masks; the final GroupNorm followed by SiLU and a finite convolution bounds the initialized output for fixed parameters. With finite data second moments, dominated convergence gives the required L^2 endpoint continuity, also after averaging over dropout masks. This establishes Regime A under the stated initialization model without global Jacobian nondegeneracy.

A.3 Proof of Theorem 4.1

Proof. By Equations (11) and (9), for every $b \in (t_0, 1)$,

$$M_v(q) \geq \begin{cases} 4c_J \int_{t_0}^b q(t)(1-t)^{-4}dt, & \text{Regime A}, \\ 4c_J \int_{t_0}^b q(t)(1-t)^{-2}dt, & \text{Regime B}. \end{cases} \tag{34}$$

Now assume Equation (16). For every $b \in (t_1, 1)$, the nonnegativity of the integrand gives

$$M_v(q) \geq 4c_J c_q \int_{t_1}^b (1-t)^{\beta-p}dt. \tag{35}$$

If $\beta \leq p-1$, the truncated integral tends to infinity as $b \uparrow 1$. Monotone convergence therefore yields $M_v(q) = \infty$. Uniform sampling has $c_q = 1$ and $\beta = 0$, so the condition holds for both $p = 4$ and $p = 2$. □

At a trained state, Equation (9), rather than the initialization orders in Equation (11), remains the exact integrability criterion.

A.4 Proof of Proposition 4.2

Proof. Since q is a probability density, Equation (9) gives

$$M_x(q) = \int_0^1 q(t)G_\theta(t)dt \leq C_G(\theta)\int_0^1 q(t)dt = C_G(\theta). \tag{36}$$

□

A transparent sufficient condition for $C_G(\theta) < \infty$ is the following bound, with the essential supremum taken under Lebesgue time measure and the underlying data–noise law:

$$K_\theta := \operatorname*{ess\,sup}_{t\in(0,1),\,\omega\in\Omega} \|J_\theta(t;\omega)\|_{\mathrm{op}} < \infty, \qquad C_R(\theta) := \sup_{t\in(0,1)} \mathbb{E}\,\|r_\theta(t)\|^2 < \infty, \tag{37}$$

Indeed, for Lebesgue-almost every $t \in (0,1)$,

$$G_\theta(t) \le 4\mathbb{E}\left[\|J_\theta(t)\|_{\mathrm{op}}^2\,\|r_\theta(t)\|^2\right] \le 4K_\theta^2 C_R(\theta). \tag{38}$$

The residual-moment condition does not follow from the Jacobian bound and is stated separately. Neither condition requires a pointwise bounded output head. A weaker joint-moment alternative is

$$C_{Jr}(\theta) := \sup_{t\in(0,1)} \mathbb{E}\left[\|J_\theta(t)\|_{\mathrm{op}}^2\,\|r_\theta(t)\|^2\right] < \infty, \tag{39}$$

which directly yields $C_G(\theta) \le 4C_{Jr}(\theta)$. Separate uniform fourth-moment bounds on $\|J_\theta(t)\|_{\mathrm{op}}$ and $\|r_\theta(t)\|$ imply this condition by Cauchy–Schwarz.

Other aligned heads. For direct velocity prediction with residual $r_v = \hat{V}_\theta - (X - E)$ and Jacobian $J_\theta^v = \nabla_\theta \hat{V}_\theta$, the identical argument gives

$$\operatorname*{ess\,sup}_{t\in(0,1)} \mathbb{E}\left\|2(J_\theta^v)^\top r_v\right\|^2 \le 4(K_\theta^v)^2 C_R^v(\theta) \tag{40}$$

under the corresponding fixed-state Jacobian and residual-moment conditions. For summed BCE-with-logits, let A_θ be the logits and $Y \in \{0,1\}^D$. Its gradient is

$$g_{\mathrm{BCE}} = (J_\theta^A)^\top(\sigma(A_\theta) - Y). \tag{41}$$

Because $\|\sigma(A_\theta) - Y\|^2 \le D$, a Jacobian bound $\left\|J_\theta^A\right\|_{\mathrm{op}} \le K_\theta^A$ holding almost surely for Lebesgue-almost every t, with the same constant K_θ^A, gives $\operatorname{ess\,sup}_{t\in(0,1)} \mathbb{E}\,\|g_{\mathrm{BCE}}\|^2 \le (K_\theta^A)^2 D$. Fixed class weights only multiply this bound by the squared maximum weight. These statements, like Proposition 4.2, are conditional fixed-state bounds rather than training-trajectory guarantees.

A.5 Proof of Proposition 4.3

Proof. Let $U \sim \mathcal{N}(m, s^2)$ with $s > 0$ and $t = \sigma(U)$. For any nonnegative measurable F,

$$\int_0^1 q_{\mathrm{LN}}(t)F(t)dt = \mathbb{E}_{U\sim\mathcal{N}(m,s^2)}[F(\sigma(U))]. \tag{42}$$

Suppose the base signal-gradient second moment has finite polynomial endpoint growth: for some $C < \infty$ and $p \ge 0$,

$$G_\theta(t) \le C(1-t)^{-p} \tag{43}$$

on a terminal interval. Under velocity-space mismatch, the integrand is bounded by $C(1-t)^{-(p+4)}$. Since

$$(1 - \sigma(U))^{-1} = 1 + e^U, \tag{44}$$

the transformed integrand grows at most as a constant multiple of $e^{(p+4)U}$ for sufficiently large U, whose expectation is finite under a Gaussian distribution. Thus the terminal contribution $\int_{t_0}^1 q_{\mathrm{LN}}(t)a(t)^2 G_\theta(t)dt$ is finite for every finite polynomial endpoint order. This argument does not control $(0, t_0]$: the full second moment is finite only when that nonterminal contribution is also integrable. This conclusion requires an upper-growth condition; an initialization-time lower bound such as Assumption 3.1 is not sufficient for it.

If $C_G(\theta) = \operatorname{ess\,sup}_{t\in(0,1)} G_\theta(t) < \infty$, then the full integral satisfies

$$\int_0^1 q_{\mathrm{LN}}(t)a(t)^2 G_\theta(t)dt \le C_G(\theta)\mathbb{E}\left[(1-\sigma(U))^{-4}\right] < \infty. \tag{45}$$

The last expectation is finite because $(1 - \sigma(U))^{-1} = 1 + e^U$ and a Gaussian has finite exponential moments of every finite order. □

B EXPERIMENTAL DETAILS

B.1 PAIRED INITIALIZATION-REGIME DIAGNOSTIC

Table 3 summarizes the protocol for the direct measurements in Table 1.

Table 3: Paired MLP initialization diagnostic: backbone and measurement settings.

Item	Setting
Backbone and initialization	
Branch	Gated MLP; two hidden layers, width 256; SiLU activations and sigmoid gates
Timestep embedding	128-dimensional sinusoidal embedding
Hidden and gate layers	Xavier-uniform weights; zero biases
Final linear layer	Zero weights and bias; all parameters remained trainable
Predictors	A: $f_\theta(Z_t, t)$; B: $Z_t + f_\theta(Z_t, t)$
Data and gradient measurement	
Data / noise	128 independent pairs $X, E \sim \mathcal{N}(0, I_{16})$, reused at all timesteps and seeds
Initialization seeds	42, 43, 44; paired parameter tensors across A and B
Endpoint grid	26 distinct distances $\epsilon = 1 - t$ from 10^{-1} to 10^{-6}
Loss / statistic	Mean-coordinate MSE; average of individual-example squared full-parameter gradient norms
Precision / updates	FP64; separate backward passes for each loss; no optimizer or parameter updates
Safeguards	No denominator floor, gradient clipping, or mixed precision

For each seed, the two predictors used exactly the same parameter tensors and differed only by the addition of the parameter-free input Z_t to the branch output. Their parameter Jacobians are consequently identical. The direct output was initially zero; the identity-residual output was initially Z_t. With the final layer zero-initialized, only its weight and bias gradients are active at this state. This is a controlled test of the two constructed initialization regimes, separate from the trained Gaussian sampler experiment below.

We evaluated 26 distinct endpoint distances from 10^{-1} to 10^{-6} in FP64. At every distance and for every example, we performed separate forward and backward passes for the two losses:

$$\ell_{x,j}^{\mathrm{mean}}(t) = \frac{1}{16}\left\|\hat{X}_{\theta_0}(Z_{t,j}, t) - X_j\right\|^2, \tag{46}$$

$$\ell_{v,j}^{\mathrm{mean}}(t) = \frac{1}{16}\left\|\frac{\hat{X}_{\theta_0}(Z_{t,j}, t) - Z_{t,j}}{1-t} - (X_j - E_j)\right\|^2. \tag{47}$$

The recorded second moments are

$$\widehat{G}_s(t) = \frac{1}{128}\sum_{j=1}^{128}\left\|\nabla_\theta \ell_{s,j}^{\mathrm{mean}}(t)\right\|^2, \qquad s \in \{x, v\}. \tag{48}$$

These are averages of individual-example squared gradient norms, not squared norms of batch-averaged gradients. Mean-coordinate MSE scales each moment by 16^{-2} relative to the summed-loss convention in the theory. The velocity-loss observations were obtained from their own backward passes, not by multiplying the aligned measurements by $(1-t)^{-4}$. No denominator floor, gradient clipping, or mixed precision was used. Parameter hashes remained unchanged throughout each diagnostic and matched between the paired predictors; data, noise, and timesteps also matched exactly.

As an implementation check, we compared the independently computed full gradient vectors with $g_v = (1-t)^{-2} g_x$ on the first example at every timestep and initialization, and checked the squared-norm ratio on all examples. Table 1 selects three distances from this grid and reports the mean and sample standard deviation of the three initialization-specific moments. This standard deviation describes initialization variability on the fixed data, not uncertainty over independent data draws.

Figure 5 displays all 128 individual squared-gradient observations separately for each seed at $\epsilon = 1 - t \approx 10^{-1}, 10^{-3}, 10^{-6}$, without smoothing or sample-level jitter. For B, each recorded value was divided by the actual ϵ^2; the normalization is prescribed by the initialization construction, not fitted to the observations. Boxes and whiskers summarize variation across the fixed examples within one initialization, not uncertainty across initialization seeds. The three seed groups and different timesteps reused the same data and must not be pooled as independent data draws. These finite-grid measurements check empirical nondegeneracy; they do not by themselves establish a uniform population lower bound or asymptotic divergence.

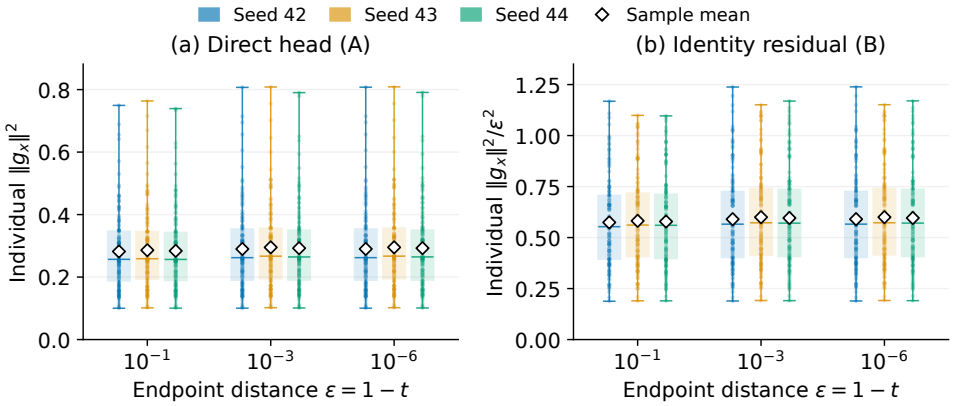

Figure 5: Individual-example distributions for the initialization control. Each stack contains 128 measurements for one seed and endpoint distance. Left: squared aligned-gradient norms for A. Right: squared aligned-gradient norms divided by ϵ^2 for B. Boxes show the interquartile range, whiskers the minimum and maximum, and diamonds the sample mean. Unlike Table 1, the right panel is endpoint-normalized to display variation within each group.

B.2 Controlled Gaussian protocol

In Figure 2, we compared Uniform and Logit-Normal sampling over three paired initializations (seeds 42, 43, 44). Each pair used identical initial parameter tensors and shared data–noise draws at every update; only the timestep sampler changed. Table 4 records the settings. This experiment measured the loss and gradient trajectories obtained when optimizing a fixed pointwise loss under different timestep samplers.

Table 4: Gaussian sampler-control training configuration.

Item	**Setting**
Backbone and data	
Backbone	Gated MLP; two hidden layers of width 256, SiLU activations; 128-dimensional sinusoidal timestep embedding
Data / interpolation	$X, E \sim \mathcal{N}(0, I_{16})$ independently; $Z_t = tX + (1-t)E$
Prediction / objective	Direct x-prediction, zero-initialized final layer, no identity skip; raw velocity MSE averaged over examples and coordinates
Training and logging	
Optimizer / learning rate	Adam, 10^{-4}; betas $(0.9, 0.999)$, $\varepsilon_{\text{Adam}} = 10^{-8}$
Budget / batch size	5,000 updates / 1,000
Timestep sampler	Uniform $(0, 1)$ or $T = \sigma(U)$, $U \sim \mathcal{N}(-0.8, 0.8^2)$
Numerics / trace	FP64; no denominator floor or gradient clipping; pre-update full-parameter minibatch-gradient norm

For each pair, a common uniform quantile u gave $T_{\text{U}} = u$ and $T_{\text{LN}} = \sigma(-0.8 + 0.8\Phi^{-1}(u))$, where Φ is the standard normal CDF. We computed the loss directly as the mean squared value of $(\hat{X}_\theta - Z_T)/(1-T) - (X - E)$, without a denominator safeguard. The zero-initialized direct head

reused the parameter snapshots from the initialization diagnostic; the sampler comparison changed neither the head nor the pointwise loss within each pair.

All six runs completed 5,000 updates with finite losses and gradients. The three Uniform runs had median gradient norms of order 10^3 and maxima ranging from 10^9 to 10^{12}, whereas Logit-Normal medians were of order 10^{-1} and maxima of order 10^0. Figure 2(a) shows the exact bin-averaged sampler densities, computed from their CDFs rather than empirical timestep counts. Panels (b,c) retain all per-update traces as faint lines and overlay the arithmetic mean across seeds followed by a trailing 25-update moving average. The loss panel uses a linear left axis for Logit-Normal and a logarithmic right axis for Uniform, as indicated by the sampler-colored labels. The native losses reflect different timestep-averaged objectives, and are not a matched-distribution prediction-quality comparison. The logged minibatch-gradient norm is distinct from the individual-example second moment estimated in Table 1.

The experiment archive contains the model source, initial states, configuration and source hashes, input-stream checks, final parameters, and every update's native loss, signal MSE, gradient norm, and minimum endpoint distance. These records make the pairing and absence of denominator clipping auditable.

B.3 BINARY-MNIST PROTOCOL

Table 5 groups the backbone, training, and evaluation settings used for Figure 3. All configurations shared the dataset, split, generation code, and FID evaluator.

Table 5: Binary-MNIST objective–sampler matrix: shared experimental configuration.

Item	Setting
Backbone and data	
Backbone	Class-conditional U-Net (Ronneberger et al., 2015)
Conditioning	Z_t, timestep through a two-layer SiLU MLP, and learned digit-label embedding
Data	Deterministically thresholded MNIST (LeCun et al., 1998), mapped to $\{-1, +1\}$; fixed train/validation split
Training: flow matching	
Interpolation	$Z_t = tX + (1-t)E$, $E \sim \mathcal{N}(0, I)$
Optimizer / learning rate	Adam (Kingma & Ba, 2015), 10^{-4}
Epochs / batch size	100 / 128
Training seeds	42, 43, 44
Timestep sampling	Uniform $(0, 1)$ or $\mathrm{logit}(T) \sim \mathcal{N}(-0.8, 0.8^2)$
Prediction–loss pairings	x-to-v MSE, x-MSE, v-MSE, and BCE-with-logits
Sampling and evaluation	
ODE sampler	Euler, 50 steps; no hard thresholding during integration
BCE signal mapping	$\hat{X}_\theta = 2\sigma(A_\theta) - 1$ before velocity computation
FID samples	50,000 conditional samples, balanced across digits
FID features	128-dimensional penultimate features of a converged MNIST classifier
Classifier training	Adam, learning rate 10^{-3}, batch size 128, 25 epochs; seed 314159
Classifier selection	Highest validation accuracy; earliest epoch in a tie (epoch 20)
Generator selection	Final epoch-100 checkpoint for every configuration
Shared evaluation	Same classifier, real-feature statistics, generation seed, and integration code
Reported variability	Mean and sample standard deviation across training seeds

The four configurations in Figure 3 used:

$$v\text{-MSE:} \quad \left\|\hat{V}_\theta - (X - E)\right\|^2, \tag{49}$$

$$x\text{-to-}v\text{ MSE:} \quad (1-t)^{-2}\left\|\hat{X}_\theta - X\right\|^2, \tag{50}$$

$$x\text{-MSE:} \quad \left\|\hat{X}_\theta - X\right\|^2, \tag{51}$$

$$\text{BCE}: \quad \text{BCEWithLogits}(A_\theta, (X+1)/2). \tag{52}$$

For BCE inference, logits were mapped back to the analog signal as $\hat{X}_\theta = 2\sigma(A_\theta) - 1$ before computing the implied velocity. All x-prediction models used the same Euler-50 ODE rule without hard thresholding during integration.

Classifier training and checkpoint selection. MNIST's 60,000 training images were thresholded at 0.5, mapped to $\{-1, +1\}$, and split into 54,000 training and 6,000 validation images with split seed 42. The `SimpleClassifier` contained two 3×3 convolutional layers (32 and 64 channels), each followed by ReLU and 2×2 max pooling, a 128-unit fully connected ReLU layer, and a ten-class output layer; dropout with probability 0.5 preceded the output layer during training. It was trained with cross-entropy and Adam for 25 epochs using learning rate 10^{-3}, batch size 128, and seed 314159. We selected the checkpoint with the highest held-out validation accuracy, retaining the earlier checkpoint in a tie, rather than selecting by generated-sample FID. The recorded maximum occurred at epoch 20 (99.03% validation accuracy). This checkpoint was frozen in evaluation mode, disabling dropout, for all feature extraction.

FID computation and generation protocol. FID-50K (Heusel et al., 2017) used the 128-dimensional activations after the `fc1` ReLU, not classification logits or probabilities. Real-reference features were extracted from 50,000 images in the training loader and cached once. For each generator, we produced 50,000 conditional samples (5,000 per digit) using generation seed 1000 and Euler-50 integration from Gaussian noise. The generator and classifier were both in evaluation mode. The implemented x-to-velocity denominator was $1 - t + 10^{-8}$ at $t = k/50$, $k = 0, \ldots, 49$; this numerical offset applied only to generation, not the raw training loss. Generated signals were passed directly to the classifier without thresholding or clipping. Only the displayed sample grids were mapped to $[0, 1]$ and clipped for visualization. FID was computed from feature means and sample covariances in double precision. Every configuration shared the frozen classifier, cached real features, generation seed, and integration code.

Generator checkpoints and reported variability. Classifier selection was separate from generator selection: all entries in Figure 3 used the final epoch-100 generator checkpoint, not the checkpoint with the best validation loss or FID. Means and sample standard deviations summarize three independently trained generators (seeds 42–44), each evaluated with the same generation seed.

Evaluator diagnostics and per-seed results. Figure 6 reports the recorded classifier losses and validation accuracy throughout training. Validation accuracy increased from 97.60% at epoch 1 to its maximum of 99.03% at epoch 20, while the lowest validation cross-entropy occurred at epoch 7. The checkpoint was selected by accuracy, not loss. These curves document evaluator training and checkpoint selection. FID-ranking sensitivity to the choice of classifier checkpoint was not evaluated in this protocol.

Table 6 gives each generator's FID under the same frozen evaluator. Uniform-sampled mismatch had FID above 10^3 in all three runs, whereas Logit-Normal-sampled mismatch and the aligned configurations remained at the order of 10^1. The detailed values expose training-seed variability rather than uncertainty from repeated sampling of one generator.

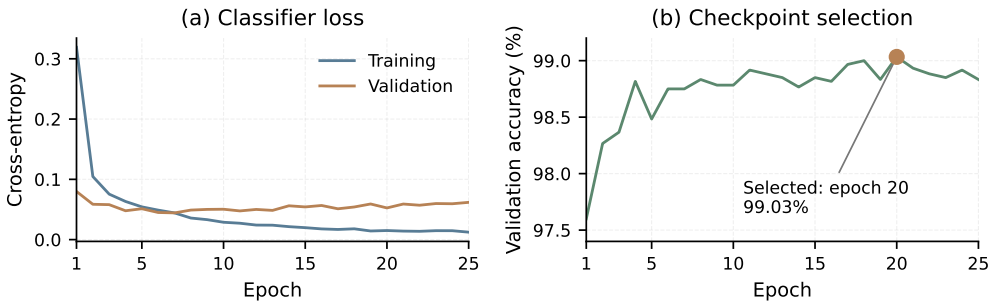

Figure 6: BMNIST FID evaluator diagnostics from the classifier used for the reported experiments. Left: training and validation cross-entropy. Right: validation accuracy, with the selected epoch-20 checkpoint marked. Curves show the recorded epoch values without smoothing.

Table 6: BMNIST FID-50K for each training seed. All entries used the final epoch-100 generator, the selected epoch-20 classifier, and generation seed 1000. The last column reports the mean and sample standard deviation.

Prediction–loss	Sampler	Seed 42	Seed 43	Seed 44	Mean ± s.d.
x-pred + v-MSE	Uniform	1170.35	1524.46	1113.80	1269.53 ± 222.57
x-pred + v-MSE	Logit-Normal	10.54	13.02	9.52	11.02 ± 1.80
x-pred + x-MSE	Uniform	15.39	9.67	13.60	12.88 ± 2.93
x-pred + x-MSE	Logit-Normal	8.45	13.38	17.02	12.95 ± 4.30
v-pred + v-MSE	Uniform	24.24	16.32	18.46	19.67 ± 4.10
v-pred + v-MSE	Logit-Normal	16.81	15.58	10.34	14.24 ± 3.44
x-pred + BCE	Uniform	12.21	8.67	13.16	11.35 ± 2.36
x-pred + BCE	Logit-Normal	7.61	10.47	8.93	9.01 ± 1.43

B.4 Binary-MNIST implementation checks

All Binary-MNIST configurations used the same threshold-binarized dataset, train/validation split, generation code, and FID evaluator. For BCE, logits were mapped to the analog signal as $2\sigma(A_\theta) - 1$ before computing the ODE velocity. Validation mode was confined to evaluation, best and final checkpoints were stored separately, and every reported seed corresponds to an independently trained model. These checks ensure that the comparison isolates the stated prediction–loss pairing and timestep sampler.

B.5 Normalized endpoint-cap sweep

The sweep used the conditional U-Net backbone and Binary-MNIST evaluation protocol in Appendix B.3. Each run used 100 epochs, batch size 128, learning rate 10^{-4}, and the fixed split with split seed 42. We crossed $C \in \{10^2, 10^4, 10^6, 10^8\}$ with training seeds 42, 43, and 44 under Uniform sampling on the full timestep interval. The loss is per-example signal MSE multiplied by

$$\widetilde{w}_C(t) = \frac{\min\{(1-t)^{-2}, C\}}{2\sqrt{C} - 1}. \tag{53}$$

Evaluation used the final epoch-100 checkpoint, 50,000 generated samples, Euler-50 integration, and generation seed 1000. Table 2 reports variability across the three training seeds.

For $C \geq 1$, define $W_C(t) = \min\{(1-t)^{-2}, C\}$. Under Uniform t,

$$\mathbb{E}W_C = 2\sqrt{C} - 1, \tag{54}$$

$$\mathbb{E}W_C^2 = \frac{4C^{3/2} - 1}{3}. \tag{55}$$

The normalized weight $\widetilde{w}_C = W_C/(2\sqrt{C}-1)$ has unit mean and

$$\mathbb{E}\widetilde{w}_C^2 = \frac{4C^{3/2}-1}{3(2\sqrt{C}-1)^2} \sim \frac{\sqrt{C}}{3}. \tag{56}$$

The corresponding FID-50K results are summarized in Table 2 of Section 5.1.

The normalization fixes the mean analytical scale but changes the time profile of supervision. The sweep therefore evaluates the joint training effect of changing endpoint concentration and the time profile of supervision.

B.6 JiT-B/4 CONTINUOUS-IMAGE PROTOCOL

For context, JiT explicitly adopts Logit-Normal timestep sampling following the Stable Diffusion 3 rectified-flow work (Esser et al., 2024; Li & He, 2025), and ELF uses the same sampler family (Hu et al., 2026). JiT documents a 0.05 floor on the denominator in the x-to-v conversion (Li & He, 2025). ELF's paper does not state this clipping value, but its official implementation sets `t_eps=0.05` and replaces $1-t$ by $\max(1-t, \texttt{t_eps})$ in the conversion.[1] We record these implementation details to distinguish the published safeguarded recipes from our near-unclipped and raw diagnostics.

Table 7 summarizes the full objective–sampler matrix. In both mismatched training cells, the target and prediction were converted using the same $\max(1-t, 10^{-6})$ denominator. This historical near-unclipped implementation caps the loss weight at 10^{12} and is much less protective than the official JiT floor 0.05; the matrix is therefore a controlled JiT-backbone evaluation rather than an exact reproduction of the published recipe. The generation safeguard is recorded separately from the training floor.

Table 7: JiT-B/4 objective–sampler matrix: training and generation settings.

Item	Setting
Backbone and data	
Model / dataset	JiT-B/4 (Li & He, 2025); Tiny-ImageNet (Le & Yang, 2015), 64×64
Pairings / repeats	x-to-v MSE, x-MSE, v-MSE; one run per objective–sampler cell
Training: flow matching	
Epochs / effective batch	400 / 256
Optimizer	AdamW (Loshchilov & Hutter, 2019), weight decay 0.03
Learning-rate schedule	Base 5×10^{-5}; cosine decay with 5,000-step warmup
Timestep sampling	Uniform $(0,1)$ or $\mathrm{logit}(T) \sim \mathcal{N}(-0.8, 0.8^2)$
Mismatch conversion	Denominator $\max(1-t, 10^{-6})$ for prediction and target in both sampler cells
Sampling and evaluation	
ODE sampler / domain	Heun-50: 49 Heun updates followed by one Euler update; $t=0$ to $t=1$
Generation safeguard	`t_eps` $= 0.05$ for all x-prediction cells
Classifier-free guidance	Scale 3.0 (Ho & Salimans, 2022), active on $(0.1, 1.0)$; this is not the ODE domain
FID-50K	50,000 samples; `torch-fidelity` (Obukhov et al., 2020), Inception-V3 pool3 features

Paired early-training protocol. We compared three JiT-B/4 branches under Uniform sampling: aligned signal MSE, raw mismatched velocity MSE, and velocity MSE with a denominator floor. All three started from the same random initialization and preserved data order, augmentations, labels, class-dropout draws, timesteps, interpolation noise, and learning-rate schedule. Table 8 records the shared settings and the sole objective/conversion difference. Each record reports the norm of the accumulated minibatch gradient before the optimizer update; loss reductions averaged over output coordinates. Figure 8 shows the complete three-branch diagnostic, including a common signal-prediction MSE for comparing fitting across objectives.

[1] See the official ELF implementation.

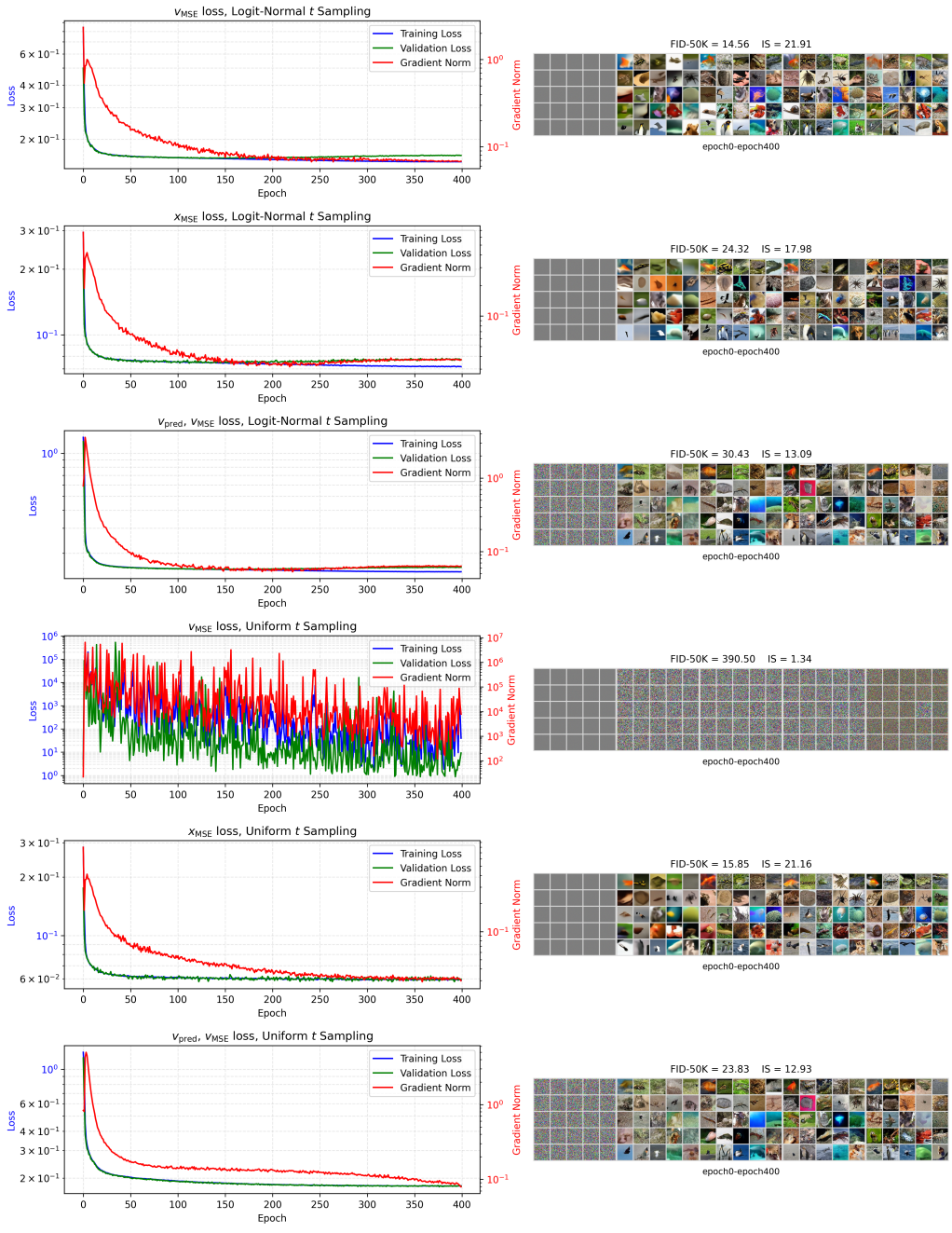

Figure 7: Expanded JiT-B/4 Tiny-ImageNet diagnostics for all six objective–sampler configurations in Figure 4. This archived view retains the complete epoch-0-to-400 sample evolution together with the original per-run loss and gradient trajectories; the main-text figure shows only the final epoch-400 samples.

Table 8: Paired JiT-B/4 early-training diagnostic configuration.

Item	Setting
Backbone / data	JiT-B/4; Tiny-ImageNet, 64×64
Sampler / initialization	Uniform $(0, 1)$; common initialization seed 0
Compared branches	Signal MSE; velocity MSE with exact denominator $1 - t$; velocity MSE with floor 10^{-6}
Budget / batch	5,000 updates; four microbatches of 64, effective batch 256
Learning-rate schedule	Base 5×10^{-5}; 5,000-step warmup on a shared 156,400-step schedule
Weight decay / label dropout	0.03 / 0.1
Precision	BF16 autocast; FP32 loss accumulation
Gradient recording	Pre-update accumulated minibatch-gradient norm; no gradient clipping or timestep truncation
Loss reduction	Mean over examples and output coordinates

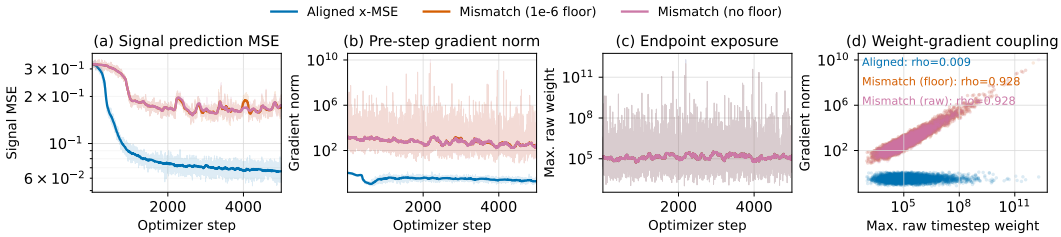

Figure 8: Supplementary paired JiT-B/4 diagnostic under Uniform sampling. Three branches shared initialization and stochastic inputs, using aligned signal MSE, raw mismatched velocity MSE, or velocity MSE with a 10^{-6} denominator floor. Panels show signal MSE, pre-update minibatch-gradient norms, endpoint exposure, and weight–gradient associations. Thin curves retain all observations; thick curves show 101-step rolling medians.

The median pre-update gradient norm was 0.306 for alignment and 590.98 for raw mismatch. The supplementary 10^{-6} floor activated at step 2,192, where the largest sampled timestep was 0.9999992847. The largest gradient norms were 5.44×10^9 with the floor and 1.06×10^{10} without it; their medians were 596.91 and 590.98, respectively. Both mismatched branches retained higher early signal MSE than alignment. Thus the floor changed the extreme event, while the qualitative training contrast persisted. Correlations in Figure 8 use all 5,000 paired records and the maximum applied mismatch weight in each minibatch; the aligned branch is compared with the corresponding raw endpoint exposure. These associations describe the co-variation of sampled endpoint exposure and raw gradients along the recorded training trajectories. All three branches remained numerically finite over the 5,000-step horizon.

Frozen-checkpoint protocol. For Figure 9, we evaluated the raw, non-EMA weights from two epoch-399 checkpoints (400 completed epochs): aligned x-MSE with Uniform training and mismatched x-prediction+v-MSE with Logit-Normal training. Both evaluations used the same 32 fixed validation images from distinct, evenly spaced class indices, the same noise realizations, and 15 timesteps from $t = 0$ to approximately $1 - 10^{-6}$. No model parameters were updated. Forward passes, losses, and gradient evaluations used FP32 without a denominator floor or gradient clipping.

At each timestep, we backpropagated one image at a time and averaged the squared Euclidean norms of the resulting full-parameter gradients. With $\ell_x^{\text{mean}} = \ell_x / D$, the plotted empirical aligned moment is

$$\widehat{G}_\theta(t) = \frac{1}{n} \sum_{j=1}^{n} \left\| \nabla_\theta \ell_{x,j}^{\text{mean}}(t) \right\|^2, \qquad n = 32. \tag{57}$$

It is a fixed-set empirical counterpart of $G_\theta(t)/D^2$, rather than the squared norm of an averaged gradient. The coordinate normalization changes the magnitude but not the endpoint order. Because image selection is fixed and class-stratified, these values describe the diagnostic set rather than an unbiased estimate over the full training distribution.

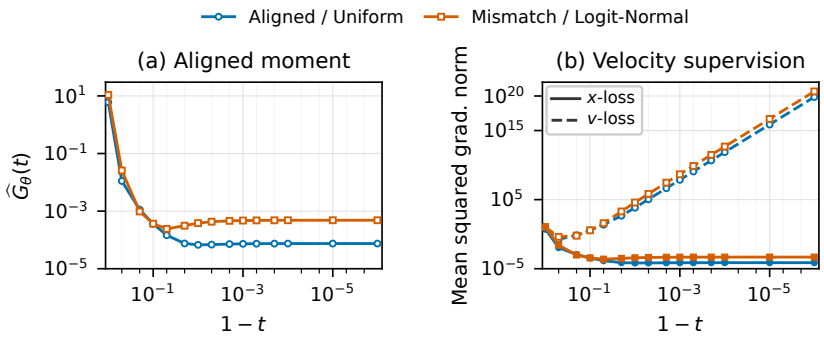

Figure 9: Frozen JiT endpoint diagnostic after training. Left: empirical aligned-gradient second moment. Right: aligned (solid) and mismatched (dashed) losses at the same checkpoints. Colors identify the training recipe. Each point averages squared full-parameter gradient norms over the same 32 fixed validation image/noise pairs, evaluated one example at a time in FP32 without updates or clipping.

For each checkpoint and input, the mismatched objective was also evaluated in the algebraically simplified form $(1-t)^{-2}\ell_x^{\text{mean}}$, with the same model output. The resulting squared-gradient ratio is prescribed by the conversion identity. Comparing this form with direct velocity division gave maximum relative squared-gradient discrepancies below 2×10^{-7} over the recorded grid; the ratio audit used the actual FP32 value of $1-t$. The empirical information is the behavior of the base moment itself: over the last three endpoint locations it approached a plateau in both checkpoints. The observed plateau concerns the recorded finite grid. The two checkpoints also differed in their training samplers, so the comparison reflects both objective and sampling choices.

For the aligned-trained checkpoint, reducing $1-t$ from approximately 10^{-4} to 10^{-6} left the aligned moment near 7.47×10^{-5}, while the mismatched moment rose from 7.47×10^{11} to 7.10×10^{19}. The observed plateau is empirical; the amplification ratio follows from Equation (5). Together, these measurements show substantial endpoint amplification at the evaluated trained states over the recorded timestep grid.

Exploratory trained-state checks. We additionally recorded two paired restarts from trained checkpoints. Switching three trained Binary-MNIST checkpoints from aligned to mismatched supervision increased the first shared-minibatch gradient norm by $306\times$–$2766\times$; after 2,000 paired updates, FID-50K was 30.08 ± 19.13 for alignment and 1554.13 ± 612.71 for mismatch. A fresh-AdamW restart from a trained JiT checkpoint produced a $19{,}896\times$ first-minibatch raw-gradient ratio; the mismatch branch had 43% higher mean signal MSE over 1,000 updates while remaining numerically finite. These finite-horizon observations describe changes in raw gradients, signal fitting, and generation following an objective switch. The interventions changed both the time weighting of the objective and the gradients presented to the optimizer.

B.7 MIMO DETECTION AS A SUPPLEMENTARY APPLICATION

We used the conditional flow-matching protocol and SGDiT backbone introduced by Jiang et al. (2026). This application provides supplementary evidence for the mismatch mechanism in conditional inference and additionally checks checkpoint-selection sensitivity. We consider the real-valued equivalent system

$$Y = HX + N, \qquad X \in \{-1, +1\}^{2N}, \qquad N \sim \mathcal{N}(0, \sigma^2 I), \tag{58}$$

obtained from QPSK transmission over an i.i.d. Rayleigh-fading channel. The backbone combines a Soft Graph Transformer (Hong et al., 2026) with DiT-style timestep conditioning (Peebles & Xie, 2023); it conditions on Y, H, and the timestep and transports an initial Gaussian sample toward the conditional signal distribution. In every x-prediction configuration, the backbone directly outputs $\hat{X}_\theta(Z_t, t, Y, H)$ and the detector computes the implied velocity $(\hat{X}_\theta - Z_t)/(1-t)$ only for integration. In direct v-prediction configurations, the backbone output is the velocity itself. All objectives used the same two-step Euler detector. Training and integration truncated time at $t_{\max} = 0.99$.

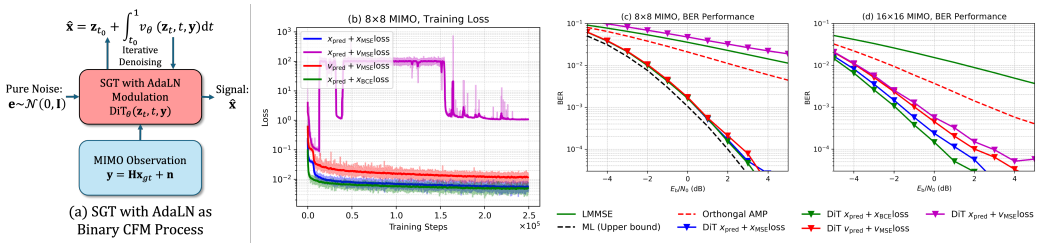

Figure 10: Supplementary conditional MIMO detection. Aligned signal objectives avoided the severe instability of the tested x-prediction/velocity-loss mismatch and achieved lower BER under the reported recipes.

The 8×8 system used learning rate 10^{-3} and the 16×16 system used 10^{-4}. We evaluated both the final 250K checkpoint and the early 13K checkpoint favorable to the mismatched baseline. Aligned objectives retained a BER advantage under both checkpoint rules. The performance gaps persisted under the evaluated truncation and checkpoint settings.

C Additional Scope Clarifications

What "robust" means. We use robustness narrowly to mean reduced sensitivity of training dynamics to terminal timestep exposure and sampler choice. It does not refer to adversarial robustness, distribution-shift robustness, or a guarantee of convergence. In the final paper title and abstract, every robustness statement is tied to this optimization setting.

Relation to discrete baselines. The binary experiments use an analog-bit representation within a continuous Gaussian path. This setting allows prediction–loss pairing and timestep sampling to be varied within the same continuous-state formulation. Categorical diffusion and discrete flow matching address the complementary question of selecting among different generative pipelines; the present comparisons focus on the internal objective–sampler mechanism.

D Compute and Assets

Table 9 gives the published reference specifications of the GPU family used for the diagnostics. These nominal specifications describe the reference design, rather than host-specific memory configurations or measured training clocks.

Table 9: Published reference GPU specifications.

Specification	Reference value
FP32 compute cores	16,384
Base clock (GHz)	2.23
Boost clock (GHz)	2.52
Standard memory	24 GB GDDR6X
Memory interface	384 bit
Graphics power (W)	450

Each diagnostic job used one GPU and executed its configurations serially; jobs ran on multiple hosts. The full JiT-B/4 matrix comprised six 400-epoch runs. One source run took approximately 26 hours, excluding queue time and some evaluation overhead. Each 5,000-step JiT smoke leg took about 48 minutes, and each 1,000-step paired restart leg took about 10 minutes. These are measured per-run times in the corresponding environments, not total exploratory compute.

MNIST (LeCun et al., 1998) and Tiny-ImageNet (Le & Yang, 2015) are established public research datasets. Binary MNIST is obtained by deterministic thresholding of MNIST. Tiny-ImageNet is derived from ImageNet and is used under its research and educational access terms. No human-subject data were collected, and no new dataset is released.