Why Adversarial Diffusion Trains More Stably Than GANs: A Local Jacobian View

author names withheld

Under Review for the Workshop on High-dimensional Learning Dynamics, 2026

Abstract

Diffusion models are widely observed to train more reliably than GANs. We ask whether this stability comes primarily from the ELBO scalar objective or from the step-wise denoising structure itself. Building on the local dynamical-systems analysis of Mescheder et al. [5], we show that min–max training induces rotational components, whereas the diffusion MSE objective yields a real spectrum near optima. For adversarial diffusion, the generator-discriminator coupling term is averaged across timesteps, reducing the effective rotational strength. Lightweight 2D experiments support the improved learning-rate robustness of adversarial diffusion over GANs, while ELBO-trained diffusion remains most stable in our setting.

1. Introduction

Many modern generative models achieve high sample quality but differ sharply in training reliability and sampling speed: GANs [1] excel at quick one-step sampling, but their adversarial training is notoriously unstable and prone to mode collapse. Diffusion models [2, 9] on the other hand, achieve great mode coverage while maintaining stable training due to their well-defined scalar objective. Their parametrization needs many small steps however, significantly slowing sampling. To address this, previous works have tried to reduce the required number of denoising steps, such as iteratively distilling models with more steps to models with fewer steps [7, 8] or using adversarial training, which does not need many small steps [3, 11, 12]. [11] conjectured that in the diffusion models with step-wise denoising, even purely adversarial training is stable, which raises the question whether the reliability of diffusion models is primarily due to their KL-based scalar objective, or the step-wise denoising structure can improve the local optimization geometry of adversarial training.

Contributions. (1) We give a unified local-Jacobian comparison of GAN, adversarial diffusion, and ELBO diffusion training. (2) For adversarial diffusion we derive the c_t-weighted timestep averaging of the coupling term that controls rotational dynamics. (3) We validate these predictions with novel diagnostic plots (Jacobian spectra and $c_t A_t S_t$ proxies) and learning-rate sweeps on 2D mixtures.

Related work. There has been previous work analyzing the training stability of GANs, from a theoretical and optimization standpoint [4–6], but to the best of our knowledge, we are the first to analyze how the step-wise denoising process helps the optimization geometry of adversarial training.

Notation. We write $p_{\mathcal{D}}$ for the data distribution, $p_{\mathcal{N}}$ for the noise prior, and $\phi(\cdot)$ for a standard normal distribution of appropriate dimension. As discussed in (App. A) we will always assume to work with the non-singular Nash equilibrium $u^* = 0$.

2. Background

2.1. GANs

GANs [1] have many different variations, but the basic principle is that two networks, a generator G_θ and discriminator D_ψ compete in a min–max game, where the generator network tries to generate samples to fool the discriminator and the discriminator tries to discern between real and synthetic data. As in [5], we can write the objective function as

$$\min_\theta \max_\psi \ \mathbb{E}_{p_\mathcal{N}(z)}[f(D_\psi(G_\theta(z)))] + \mathbb{E}_{p_\mathcal{D}(x)}[f(-D_\psi(x))],$$

where f is a smooth function with $f'(0) \neq 0$ and $f''(0) < 0$. Our goal in training is to find the Nash equilibrium, where both generator and discriminator cannot improve by changing their strategy one-sidedly. If the networks are expressive enough this Nash equilibrium is uniquely defined [1], but in practice it has been observed to be very hard to find that saddle point. Training is often sensitive to hyperparameters and may exhibit oscillatory behavior due to the underlying min–max geometry.

2.2. Diffusion Models

Diffusion models [9] define latent variables $x_1, ..., x_T$ by gradually adding Gaussian noise to data $x_0 \sim q(x_0)$ with a variance schedule $(\beta_t)_t$, $q(x_t|x_{t-1}) \sim \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I)$, and learn the reverse denoising process. As shown in [2], training via the ELBO reduces (up to a θ-independent constant) to the denoising score matching objective

$$\mathcal{L}(\theta) = \sum_{t=1}^{T} C_t \mathbb{E}_{x_0,\varepsilon}[\|\varepsilon - \varepsilon_\theta(x_t(x_0, \varepsilon), t)\|^2],$$

with C_t depending on the variance schedule. This is a scalar objective additive across timesteps, in contrast to a min–max game, and is empirically observed to train reliably with low hyperparameter sensitivity. The reverse process is approximately conditionally Gaussian only for many small steps, motivating recent adversarial and hybrid objectives for faster sampling.

2.3. Adversarial diffusion

To reduce the number of denoising steps, [11] replaces the ELBO with a per-timestep conditional GAN objective for $q(x_{t-1}|x_t)$. Using the diffusion posterior to parametrize $p_\theta(x_{t-1}|x_t) = \int \phi(z) q(x_{t-1}|x_t, x_0 = G_\theta(x_t, t, z))$, their training objective can be described as

$$\min_\theta \max_\psi \mathbb{E}_{\pi(t)q(x_t)} \Big[\mathbb{E}_{q(x_{t-1}|x_t)} \big[\log D_\psi(x_{t-1}, x_t, t) \big] + \mathbb{E}_{p_\theta(x_{t-1}|x_t)} \big[\log \big(1 - D_\psi(x_{t-1}, x_t, t)\big) \big] \Big], \tag{1}$$

where $\pi(t)$ is uniform on $\{1, ..., T\}$. This yields a min–max game at every timestep, while retaining the step-wise denoising structure. The original formulation was as a non-saturating GAN objective, but by [6], the local convergence geometry is the same in the equilibrium. [11] have found training to be empirically stable and avoiding mode-collapse, attributing the improved stability over one-step GANs to the conditioning providing a strong signal rather than learning from pure noise and the noising smoothing the distribution. To the best of our knowledge however, there has been no optimization-theoretic account for this.

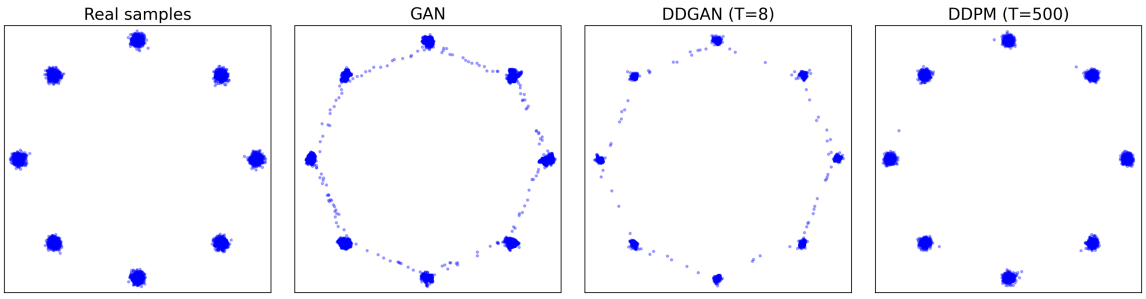

Figure 1: 8-Gaussians qualitative samples. GAN and DDGAN exhibit ring-like inter-mode mass, whereas DDPM concentrates near mixture components. Off-mode rate (3σ rule): GAN 2.1%, DDGAN 1.6%, DDPM 0.2% for 5000 samples.

3. Optimization dynamics

As in [5], denoting $p_\theta = (G_\theta)_{\#}p_{\mathcal{N}}$ as the pushforward of $p_{\mathcal{N}}$, we analyze local stability of training dynamics of the differentiable zero-sum game

$$L(\theta, \psi) = \mathbb{E}_{p_\theta(z)}[f(D_\psi(z))] + \mathbb{E}_{p_{\mathcal{D}}(x)}[f(-D_\psi(x))],$$

under simultaneous-gradient ascent, as is a standard approach in training GANs. As noted in [6], this formulation is equivalent to the regular GAN training objective with $f(x) = -\log(1 + \exp(-x))$. Under this, the training dynamics can be described by the $\mathcal{C}^1$ operator,

$$F_h(\theta, \psi) = (\theta, \psi) + h \cdot v(\theta, \psi), \text{ where } v(\theta, \psi) = \begin{pmatrix} -\nabla_\theta L(\theta, \psi) \\ \nabla_\psi L(\theta, \psi) \end{pmatrix}.$$

Here, a Nash equilibrium under is a fixed point (θ^*, ψ^*) of F_h, and by classic fixed point theory, for local convergence, it is sufficient for the spectral radius of ∇F_h in the fixed point to be smaller than one. Now, if the eigenvalues of ∇v have negative real part, we can write the eigenvalues of F_h as $\lambda = 1 + h(a + ib)$ and have $|\lambda| < 1$ if and only if $0 < h < \frac{-2a}{a^2+b^2}$. Clearly, if the spectrum of ∇v is purely real, then we have a much wider range of learning rates that lead to convergence, while if we have imaginary parts (b in example calculation), the training gets very unstable. It has significant rotational dynamics around the equilibrium and training is very dependent on the learning rate, while with eigenvalues close to the imaginary axis, there may be intractably low learning rates necessary to achieve convergence at all.

3.1. GANs

For GANs, [5] showed that without regularization, training dynamics can be purely cyclic and non-convergent. Adding the R_1 penalty $R_1(\theta, \psi) = \frac{\gamma}{2}\mathbb{E}_{x\sim p_{\mathcal{D}}}\|\nabla_x D_\psi(x)\|_2^2$ and assuming sufficient expressivity of D_ψ and G_θ, the regularized Jacobian $\tilde{v}'(\theta^*, \psi^*)$ takes a block form (full statement in App. B) whose imaginary parts are bounded by the generator-discriminator mixed gradient $\|K_{DG}\|_2$, where

$$K_{DG} = f'(0)\mathbb{E}_{p(z)}\left[\nabla_{\psi,x} D_{\psi^*}(G_\theta(z))\nabla_\theta G_\theta(z)\right]|_{\theta=\theta^*}. \tag{2}$$

The eigenvalues have negative real part, guaranteeing local convergence at small learning rates, but significant imaginary parts make this convergence highly learning-rate sensitive.

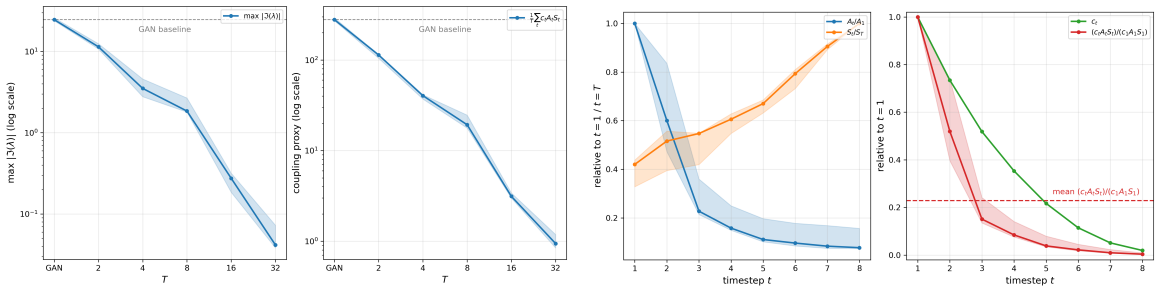

Figure 2: Optimization-geometry diagnostics. (a) Scaling with denoising depth T (GAN corresponds to $T = 1$): both the rotational magnitude $\max|\Im(\lambda)|$ and the coupling proxy $\frac{1}{T}\sum_t c_t A_t S_t$ decrease (IQR). (b) DDGAN timestep diagnostics (median with IQR across seeds), showing the decay of c_t and the normalized product $(c_t A_t S_t)/(c_1 A_1 S_1)$ over t.

3.2. DDGAN

Recall the training objective of DDGANs in (1). By denoting $p_{\mathcal{D}}(x_t, x_{t-1}, t) := \pi(t)q(x_t)q(x_{t-1}|x_t)$ and $p_\theta(x_{t-1}, x_t, t) := \pi(t)q(x_t)p_\theta(x_{t-1}|x_t)$ we can write this as a GAN objective

$$L(\theta, \psi) = \mathbb{E}_{p_\theta(z)}[f(D_\psi(z))] + \mathbb{E}_{p_{\mathcal{D}}(x)}[f(-D_\psi(x))]$$

and recover the minimax-game interpretation described by [5]. For their convergence theorem to hold, in addition to smoothness assumption on the Nash equilibrium manifold $\mathcal{M}$, we require that $d_{\psi^*} \equiv 0$ on a neighborhood of the support of the data distribution $p_{\mathcal{D}}$ and that the stationary generator induces the data distribution: $p_{\theta^*}(x_{t-1}|x_t) = q(x_{t-1}|x_t)\ q(x_t)\ a.s.$ for all t.

Plugging the DDGAN parametrization of $p_\theta(x_{t-1}|x_t)$ into the Jacobian formula (4) and applying the diffusion posterior identities of [2] (full derivation in App. C), the mixed gradient decomposes as a c_t-weighted average over timesteps:

$$K_{DG} = \frac{f'(0)}{T}\sum_t c_t K^t_{DG}, \qquad \|K_{DG}\|_2 \le \frac{|f'(0)|}{T}\sum_{t=1}^{T} c_t A_t S_t, \tag{3}$$

with diffusion-posterior coefficients $c_t = \frac{\sqrt{\bar\alpha_{t-1}}\beta_t}{1-\bar\alpha_t}$, $\alpha_t = 1-\beta_t$, per-timestep mixed gradient (equivalent to a single-step GAN Jacobian) $\bar\alpha_t = \prod_{i\le t}\alpha_i$, K^t_{DG}, and per-timestep discriminator and generator gradient norms A_t, S_t. Under the VP-SDE schedule of [10], $c_t \le 1$, and $\sum_t c_t \lesssim \log T$, so timestep-averaging shrinks the effective coupling in K_{DG} by a factor $O(\log T/T)$ (assuming $\|K^t_{DG}\|_2$ does not grow with t). This yields a tighter bound on the imaginary parts of the Jacobian than for single-step GANs, with better conditioning the more noising steps you take (Fig. 2a).

3.3. DDPM

Finally, for the regular diffusion model, the training dynamics under gradient descent can be described as $F_h(\theta) = \theta - h\nabla_\theta\mathcal{L}(\theta)$ and thus have the symmetric Jacobian $J = I - h\nabla^2_\theta\mathcal{L}(\theta)$ with *real* eigenvalues. So gradient descent lacks the rotational (imaginary) components characteristic of

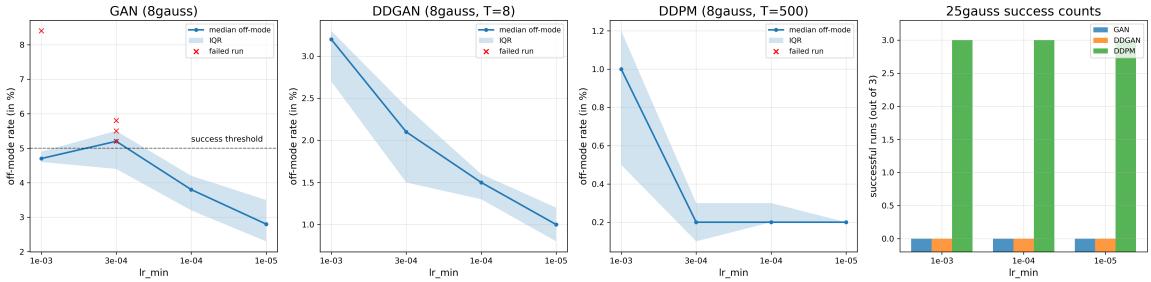

Figure 3: LR sweep on 8-Gaussians: median off-mode rate (line) with IQR (shaded) across seeds; red × denote failed runs (off-mode above the success threshold). Right: number of successful runs on 25-Gaussians (3 seeds per LR).

min–max dynamics. At a local minimizer θ^* with $\varepsilon_{\theta^*} = \varepsilon$, the residual term vanishes and

$$\nabla^2_\theta \mathcal{L}(\theta^*) = 2 \sum_{t=1}^{T} C_t \, \mathbb{E}_{x_t}[J_\theta \varepsilon_\theta(x_t, t)^\top J_\theta \varepsilon_\theta(x_t, t)] \succeq 0.$$

4. Experiments

We evaluate our theoretical findings on 2D toy mixtures: *8-Gaussians* (modes on a circle) and *25-Gaussians* (modes on a grid). All models share an MLP backbone with the VP-SDE schedule of [10], full hyperparameters in App. F. Fig. 1 shows qualitative samples on 8-Gaussians. To test learning-rate robustness we sweep the final annealed rate $lr_{\min} \in \{10^{-3}, 3{\cdot}10^{-4}, 10^{-4}, 10^{-5}\}$ over 5 seeds on 8-Gaussians (Fig. 3). We report *off-mode rate* (fraction of samples outside 3σ of every mode) where we declare success if all modes are covered and off-mode rate is below 5%. DDPM is robust across the sweep and achieves near-baseline off-mode mass, while GAN exhibits more failures at larger $lr_{\min}$ and improves as $lr_{\min}$ decreases; DDGAN is less sensitive than GAN in this toy regime. On 25-Gaussians (3 seeds; $lr_{\min} \in \{10^{-3}, 10^{-4}, 10^{-5}\}$), only DDPM succeeds under this lightweight setup. Finally, Fig. 2 (a) shows that treating GAN as $T = 1$, increasing denoising depth T sharply reduces both $\max|\Im(\lambda)|$ and the coupling proxy $\frac{1}{T}\sum_t c_t A_t S_t$. Fig. 2(b) shows that within DDGAN ($T = 8$), c_t and $(c_t A_t S_t)/(c_1 A_1 S_1)$ decay over timesteps.

5. Conclusion

We compared GAN, adversarial diffusion and ELBO-trained diffusion through the local Jacobian of their training dynamics. Min–max objectives show rotational components (imaginary eigenvalues), which explains learning-rate sensitivity even when convergent. In adversarial diffusion, timestep averaging and c_t-weighting reduce the effective generator-discriminator coupling, yielding smaller imaginary components than in one-step GANs. ELBO-trained diffusion is the most stable in our setting, consistent with its scalar objective yielding a real spectrum near optima. Our toy-mixture experiments support these claims via learning-rate sweeps, Jacobian spectra and per-timestep coupling proxies. Overall, adversarial diffusion can improve GAN-style stability via timestep averaging, but scalar objectives are the cleanest route to eliminating rotational dynamics near optima.

References

[1] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, page 2672–2680, Cambridge, MA, USA, 2014. MIT Press.

[2] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf.

[3] U-Chae Jun, Jaeeun Ko, and Jiwoo Kang. Generative adversarial diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16786–16796, October 2025.

[4] Lars Mescheder, Sebastian Nowozin, and Andreas Geiger. The numerics of gans. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 1823–1833, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

[5] Lars M. Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for gans do actually converge? In *International Conference on Machine Learning*, 2018. URL https://api.semanticscholar.org/CorpusID:3345317.

[6] Vaishnavh Nagarajan and J. Zico Kolter. Gradient descent gan optimization is locally stable. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 5591–5600, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

[7] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=TIdIXIpzhoI.

[8] Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol, editors, *Computer Vision – ECCV 2024*, pages 87–103, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-73016-0.

[9] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 2256–2265, Lille, France, 07–09 Jul 2015. PMLR. URL https://proceedings.mlr.press/v37/sohl-dickstein15.html.

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

[11] Zhisheng Xiao, Karsten Kreis, and Arash Vahdat. Tackling the generative learning trilemma with denoising diffusion GANs. In *International Conference on Learning Representations (ICLR)*, 2022.

[12] Yanwu Xu, Mingming Gong, Shaoan Xie, Wei Wei, Matthias Grundmann, kayhan Batmanghelich, and Tingbo Hou. Semi-implicit denoising diffusion models (SIDDMs). In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=gaktiSjatl.

Appendix A. Local coordinates near the Nash equilibrium manifold

For neural-network parametrizations, the set of Nash equilibria typically forms a smooth manifold $\mathcal{M}$ due to reparametrization symmetries [5]. With parameter vector $u = (\theta, \psi)$ and $u^* \in \mathcal{M}$ an equilibrium, by the inverse function theorem there exists a local diffeomorphism φ mapping a neighborhood of u^* to local coordinates $\alpha = \varphi(\mathcal{M})$ in which $\mathcal{M}$ is locally flattened, i.e. $\varphi(\mathcal{M}) = \{0\}^k \times \mathbb{R}^{d-k}$. [5] showed that the Jacobians in the two coordinate systems have the same spectrum, and local convergence is determined by the restriction of the Jacobian to the normal space $N_{u^*}\mathcal{M}$. Hence, without loss of generality we work in local coordinates with $u^* = 0$ and analyze the Jacobian restricted to $N_0\mathcal{M}$.

Appendix B. GAN Jacobian (full statement)

Under the regularization above, the regularized vector field is

$$\tilde{v}(\theta, \psi) = (-\nabla_\theta L(\theta, \psi),\ \nabla_\psi L(\theta, \psi) - \nabla_\psi R_1(\theta, \psi))$$

and the Jacobian of $\tilde{v}$ takes the form

$$\tilde{v}'(\theta^*, \psi^*) = \begin{pmatrix} 0 & -K_{DG}^T \\ K_{DG} & K_{DD} - L_{DD} \end{pmatrix}, \tag{4}$$

with K_{DG} as in (2) and

$$\begin{aligned} K_{DD} &= f''(0)\mathbb{E}_{p_{\mathcal{D}}(x)}\left[\nabla_\psi D_{\psi^*}(x)\nabla_\psi D_{\psi^*}(x)^T\right], \\ L_{DD} &= \gamma\mathbb{E}_{p_{\mathcal{D}}(x)}\left[\nabla_{\psi,x} D_{\psi^*}(x)\nabla_{\psi,x} D_{\psi^*}(x)^T\right], \end{aligned} \tag{5}$$

where $K_{DD} - L_{DD}$ is symmetric negative definite on $N_0\mathcal{M}$. The eigenvalues of matrices of the form

$$J = \begin{pmatrix} 0 & -B^T \\ B & -Q \end{pmatrix},$$

with Q symmetric positive definite have negative real part and satisfy

$$|\Im(\lambda)| \le \sqrt{\lambda_{\max}(B^T B)} = \|B\|_2.$$

Applying this with $B = K_{DG}$, $Q = L_{DD} - K_{DD}$ gives the bound used in the main text.

Appendix C. Detailed DDGAN derivation

We give the full chain from the DDGAN objective to the bound (3) in the main text. Starting from the GAN-form objective

$$L(\theta, \psi) = \mathbb{E}_{p_\theta(z)}[f(D_\psi(z))] + \mathbb{E}_{p_{\mathcal{D}}(x)}[f(-D_\psi(x))],$$

we use the result of (4) and (5) and recover, by plugging in the parametrization of $p_\theta(x_{t-1}|x_t)$,

$$K_{DG} = f'(0)\nabla_\theta \mathbb{E}_{\pi(t)q(x_t)\phi(z)\phi(\varepsilon)}\left[\nabla_\psi D_{\psi^*}(\mu(x_t, G_\theta(x_t, t, z), t) + \tilde{\beta}_t\varepsilon, x_t, t)\right]|_{\theta=\theta^*},$$

where we used the exact distribution $q(x_{t-1}|x_t, x_0) = \mathcal{N}(x_{t-1}; \mu(x_t, x_0, t), \tilde{\beta}_t I)$ as derived in [2], with $\tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$, $c_t = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}$ and $\mu(x_t, x_0, t) = c_t x_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}x_t$. Plugging this back into $g := \frac{1}{f'(0)}K_{DG}$, we recover

$$\nabla_\theta g(\theta) = \mathbb{E}_{\substack{\pi(t)q(x_t)\\ \phi(z)\phi(\varepsilon)}}\left[\nabla_{\psi,x_{t-1}} D_{\psi^*}(x^\theta_{t-1}(x_t, z, \varepsilon), x_t, t)\left(\frac{\partial}{\partial x_0}\mu(x_t, x_0^\theta, t)\right)\nabla_\theta G_\theta(x_t, t, z)\right]|_{\theta=\theta^*}.$$

Defining the per-timestep mixed gradient and gradient norms,

$$\begin{aligned}
K^t_{DG} &:= \mathbb{E}_{q(x_t)\phi(z)\phi(\varepsilon)}\left[\nabla_{\psi,x_{t-1}} D_{\psi^*}(x^\theta_{t-1}(x_t, z, \varepsilon), x_t, t)\nabla_\theta G_\theta(x_t, t, z)\right]|_{\theta=\theta^*},\\
A_t &= \left(\mathbb{E}_{q(x_t)q(x_{t-1}|x_t)}[\|\nabla_{\psi,x_{t-1}} D_\psi(x_{t-1}, x_t, t)\|_2^2]\right)^{1/2},\\
S_t &= \left(\mathbb{E}_{q(x_t)\phi(z)}[\|\nabla_\theta G_\theta(x_t, t, z)\|_2^2]\right)^{1/2},
\end{aligned}$$

we obtain

$$K_{DG} = \frac{f'(0)}{T}\sum_t c_t K^t_{DG}$$

and thus

$$\|K_{DG}\|_2 \le \frac{|f'(0)|}{T}\sum_{t=1}^T c_t\|K^t_{DG}\|_2 \le \frac{|f'(0)|}{T}\sum_{t=1}^T c_t A_t S_t.$$

Appendix D. Bounds on c_t

Bound $c_t \le 1$. From $\bar{\alpha}_t = \bar{\alpha}_{t-1}(1-\beta_t)$, we have $\bar{\alpha}_{t-1}\beta_t = \bar{\alpha}_{t-1} - \bar{\alpha}_t$, so

$$c_t = \frac{\bar{\alpha}_{t-1} - \bar{\alpha}_t}{\sqrt{\bar{\alpha}_{t-1}}(1-\bar{\alpha}_t)}.$$

Since $\bar{\alpha}_t \in [0,1]$ is decreasing, $\bar{\alpha}_{t-1} - \bar{\alpha}_t \le \sqrt{\bar{\alpha}_{t-1}} - \bar{\alpha}_t \le \sqrt{\bar{\alpha}_{t-1}}(1-\bar{\alpha}_t)$, hence $c_t \le 1$.

Logarithmic bound on $\sum_t c_t$. Under the VP-SDE schedule of [10],

$$1 - \bar{\alpha}_t = \sigma^2(t/T) = 1 - e^{-\beta_{\min}t/T - \frac{1}{2}(\beta_{\max}-\beta_{\min})(t/T)^2} \ge \tfrac{1}{2}\beta_{\min}\tfrac{t}{T},$$

using $1 - e^{-x} \ge x/2$ for $x \in [0,1]$. Since $\sqrt{\bar{\alpha}_{t-1}} \le 1$ and $\beta_t \le \beta_{\max}/T$,

$$c_t = \frac{\sqrt{\bar{\alpha}_{t-1}}\,\beta_t}{1-\bar{\alpha}_t} \le \frac{\beta_t}{1-\bar{\alpha}_t} \le \frac{\beta_{\max}/T}{\beta_{\min}t/(2T)} = \frac{2\beta_{\max}}{\beta_{\min}}\cdot\frac{1}{t}.$$

Summing the harmonic series,

$$\sum_{t=1}^T c_t \le \frac{2\beta_{\max}}{\beta_{\min}}\sum_{t=1}^T\frac{1}{t} = O(\log T).$$

Appendix E. Assumptions and limitations

Our analysis is local (near equilibria) and relies on the standard expressivity/smoothness assumptions of [5]. For DDGAN we additionally assume that the per-timestep mixed-gradient factors K^t_{DG} do not grow in t, which we empirically support via the proxy $(c_t A_t S_t)$ in Fig. 2b; in practice, controlling per-timestep factors (e.g. via weight decay or spectral normalization) may help prevent their growth. Finally, our Jacobian analysis assumes gradient steps while experiments use Adam, but the empirical trends agree.

Appendix F. Implementation details

Architecture. All models use the same MLP architecture (3 hidden layers, width 64); DDGAN/DDPM use sinusoidal timestep embeddings.
Training. We train for 50k steps with Adam and batch size 1024. For DDGAN ($T = 8$) and DDPM ($T = 500$) we use the VP-SDE schedule of [10] as in [11] and set the R_1 weight to $\gamma = 0.01$.
Logging. We implement GAN, DDGAN and DDPM in PyTorch with MLP backbones. Experiments log off-mode rate, mode coverage, Jacobian eigenvalue estimates based on Arnoldi iteration on the linearized vector field, and per-timestep and averaged coupling proxies (A_t, S_t), $(c_t A_t S_t)$ and $\frac{1}{T}\sum_t c_t A_t S_t$. Figures are generated from these logs with fixed seeds.

Appendix G. AI Use Declaration

In accordance with workshop policy, we disclose our use of AI tools. ChatGPT 5.2 Edu was used during literature search and ideation (summarizing candidate papers, discussing scope and feasibility of topic directions), for grammar and spelling corrections, and for sentence-level clarity suggestions adopted without changing the meaning or structure of any paragraph; the literature review, outline, and all conceptual content are our own. The 2D toy implementations of GAN, DDGAN, and DDPM, the training-run logging, and the plotting code were drafted by Claude Opus 4.5 and Codex coding agents under our specification, and we verified correctness. The theoretical analysis, derivations, and experimental design are our own work.