Midpoint Generative Models

Daniil Shlenskii [*1 2] **Nikita Gushchin** [*2 1] **Lev Novitskiy** [3 2] **Dmitry V. Dylov** [2 1] **Alexander Korotin** [2 1]

Abstract

We introduce *Midpoint Generative Models* (MGM), a principled framework for training one-step generative models. MGM is based on a simple symmetry of Flow Matching with linear interpolation: when the two endpoint distributions coincide, the corresponding drift field vanishes at the midpoint time, $t = 1/2$. We show that the norm of this field defines a valid discrepancy between distributions, which we call the *Midpoint Divergence*. We extend this discrepancy beyond the midpoint by introducing randomly flipped interpolations and further generalize it by replacing deterministic linear Flow Matching interpolations with symmetric stochastic interpolants, yielding a generalized Midpoint Divergence. Finally, we derive a variational formulation of our generalized divergence, yielding a tractable objective for training a one-step generator. The resulting MGM algorithm offers an effective and theoretically grounded approach to generative modeling, achieving competitive performance against existing one-step generative modeling methods.

1. Introduction

Diffusion and flow models are now at the core of many generative systems, enabling a wide range of practical applications: image generation (Dhariwal & Nichol, 2021; Rombach et al., 2022), video generation (Ho et al., 2022), audio generation (Kong et al., 2021), image editing (Meng et al., 2022), and image-to-image translation (Saharia et al., 2022). However, whenever diffusion models are applied, the next question is often the same: how can we make them fast enough for practical use? Several lines of work address this question, including the development of more accurate numerical solvers (Song et al., 2020; Karras et al., 2022; Lu et al., 2022), the distillation of pre-trained diffusion or flow models into one- or few-step generators (Salimans & Ho, 2022; Song et al., 2023; Yin et al., 2024b; Sauer et al., 2024; Gu et al., 2023), and the design of new training schemes that learn one-step generators from scratch using properties of diffusion and flow models (Song et al., 2023; Song & Dhariwal, 2024; Frans et al., 2025; Geng et al., 2025a).

In this work, we propose a novel and effective framework based on a previously unused symmetry of Flow Matching with linear interpolation (Lipman et al., 2022; Liu et al., 2022a). We observe that Flow Matching with linear interpolation between identical endpoint distributions $p_0 = p_1$ has zero velocity vector field at the midpoint time $t = 1/2$:

$$v_{1/2}(x) \equiv 0.$$

We show that this property can be used as the foundation of a divergence between two probability distributions p_0 and p_1: if the midpoint velocity field is nonzero, then the two endpoint distributions are different. More precisely, the expected squared norm of this field defines a discrepancy that vanishes exactly when the two distributions coincide.

In its basic form, this observation applies only at the midpoint time $t = 1/2$. We further show that, by introducing a random time flip $t \mapsto 1 - t$ in the interpolation, one can overcome this problem of $v_t(x) \not\equiv 0$ for arbitrary time t and extend the construction to a time-integrated divergence. Moreover, the same idea applies beyond deterministic linear paths and yields a generalized Divergence for symmetric stochastic interpolants (Albergo & Vanden-Eijnden, 2023; Albergo et al., 2025). This divergence provides a principled way to distinguish between two endpoint distributions and serves as the basis for our final training algorithm, which we call Midpoint Generative Models (MGM).

Thus, our main contributions are as follows:

- We introduce the *Midpoint Divergence*, a new distributional discrepancy derived from the midpoint symmetry of linear interpolant Flow Matching (3.1). We prove that this divergence is definite: it is nonnegative and vanishes if and only if the two endpoint distributions are equal. We further extend this construction to time-integrated objectives (3.2) and symmetric stochastic interpolants (3.3) by introducing a random time-flip.
- We develop *Midpoint Generative Models* (MGM), a practical training framework for one-step generative modeling

[1]AXXX, Russia [2]Applied AI Institute, Russia [3]Kandinsky Lab, Russia. Correspondence to: Daniil Shlenskii <daniil.shlenskii@gmail.com>.

Published as a paper at the 1st FoGen workshop, ICML 2026, Seoul, South Korea, 2026. Copyright 2026 by the author(s).

based on the proposed divergence. We derive a variational formulation (3.4) with a learned displacement field yielding an effective, practical algorithm for training generators directly from data, without requiring a pre-trained diffusion or flow teacher. We show that our method is competitive with the other approaches, learning one-step generators based on diffusion/flow properties (5).

2. Background

We briefly review Flow Matching (2.1) and stochastic interpolants (2.2), which provide the vector-field learning and path-construction framework underlying our method. This fixes notation for the learned ODE dynamics and motivates the interpolant-based discrepancy introduced later.

2.1. Flow Matching

Flow Matching learns a time-dependent vector field that defines an ordinary differential equation (ODE) transporting samples between distributions (Lipman et al., 2022; Liu et al., 2022a). Given independent endpoint samples $X_0 \sim p_0$ and $X_1 \sim p_1$, a basic choice is the linear interpolant

$$X_t = (1-t)X_0 + tX_1, \qquad t \in [0,1].$$

This interpolant defines a path of marginal distributions $p_t = \mathrm{Law}(X_t)$. These marginals are recovered by the probability flow ODE

$$\frac{dY_t}{dt} = v_t^\star(Y_t), \qquad Y_0 \sim p_0,$$

where the drift is given by the conditional expectation:

$$v_t^\star(x) = \mathbb{E}_{X_0,X_1}\left[X_1 - X_0 \mid X_t = x\right]. \tag{1}$$

Thus, Flow Matching reduces generative modeling to learning the drift $v_t^\star(x)$ that transports samples along the marginal path from p_0 to p_1 through the ODE.

2.2. Stochastic Interpolants

The stochastic interpolants framework is a generalization of the linear one, allowing more flexible, possibly noisy paths between endpoint samples (Albergo et al., 2025). Let $X_0 \sim p_0$, $X_1 \sim p_1$, and $\epsilon \sim \mathcal{N}(0, Id)$ be independent. A stochastic interpolant is defined as

$$X_t = I_t(X_0, X_1) + \sigma_t \epsilon, \qquad t \in [0,1],$$

where

$$I_0(X_0,X_1) = X_0, \;\; I_1(X_0,X_1) = X_1, \;\; \sigma_0 = \sigma_1 = 0,$$

with I_t twice continuously differentiable in both time and space and $\sigma_t^2 \in C^2([0,1])$.

The standard Flow Matching construction is recovered by setting $I_t(X_0,X_1) = (1-t)X_0 + tX_1$ and $\sigma_t = 0$, giving the linear interpolant above.

3. Midpoint Generative Models

In this section, we introduce **M**idpoint **G**enerative **M**odels (MGM), a principled framework for training one-step generative models. We begin in §3.1 by showing that the velocity field induced by linear Flow Matching contains a discriminative signal at the midpoint $t = 1/2$. In particular, this velocity field vanishes when the endpoint distributions coincide, and its failure to vanish gives rise to the Midpoint Divergence. Next, in §3.2, we extend this idea beyond the midpoint. Since non-midpoint times introduce an orientation bias, we introduce a randomly flipped interpolation that restores symmetry and leads to a time-integrated Midpoint Divergence. In §3.3, we further generalize the construction from deterministic linear interpolations to symmetric stochastic interpolants, yielding a generalized Midpoint Divergence. Finally, in §3.4, we show how this generalized divergence can be used as a training objective for generative modeling. We derive a variational formulation with a learned displacement field model and summarize the resulting practical training algorithm for one-step generators. All proofs are provided in Appendix A.

3.1. Midpoint Divergence

Consider the Flow Matching linear interpolation between two distributions p_0 and p_1 and the corresponding velocity field v_t in (1). A key property of this field is that it vanishes at the midpoint $t = 1/2$, i.e., $v_{1/2} \equiv 0$, whenever $p_0 = p_1$. We illustrate this in Figure 1 and state it formally below.

Proposition 3.1 (Midpoint symmetry)**.** *Let X_0 and X_1 be independent samples from the same distribution p with finite first moment. Then for $p_{1/2}$-a.e. x*

$$v_{1/2}(x) = \mathbb{E}_{X_0,X_1}\big[X_1 - X_0 \mid X_{1/2} = x\big] = 0.$$

This suggests using the failure of the velocity to vanish at the midpoint as a measure of discrepancy between p_0 and p_1.

Definition 3.1 (Midpoint Divergence)**.** *For distributions p_0 and p_1, define the Midpoint Divergence*

$$D_{\mathrm{mid}}(p_0,p_1) := \mathbb{E}_{X_{1/2}} \left\| \mathbb{E}_{X_0,X_1}\big[X_1 - X_0 \mid X_{1/2}\big] \right\|_2^2. \tag{2}$$

By Proposition 3.1, we immediately have $D_{\mathrm{mid}}(p_0,p_1) = 0$ whenever $p_0 = p_1$. To justify the term "divergence," we now show the converse: the Midpoint Divergence vanishes only when the endpoint distributions coincide.

Theorem 3.1 (Definiteness of the Midpoint Divergence)**.** *Let $X_0 \sim p_0$ and $X_1 \sim p_1$ be independent and bounded almost surely. Then*

$$D_{\mathrm{mid}}(p_0,p_1) = 0 \quad \Longleftrightarrow \quad p_0 = p_1.$$

Thus, vanishing of the midpoint Flow Matching velocity field characterizes equality of the endpoint distributions. A

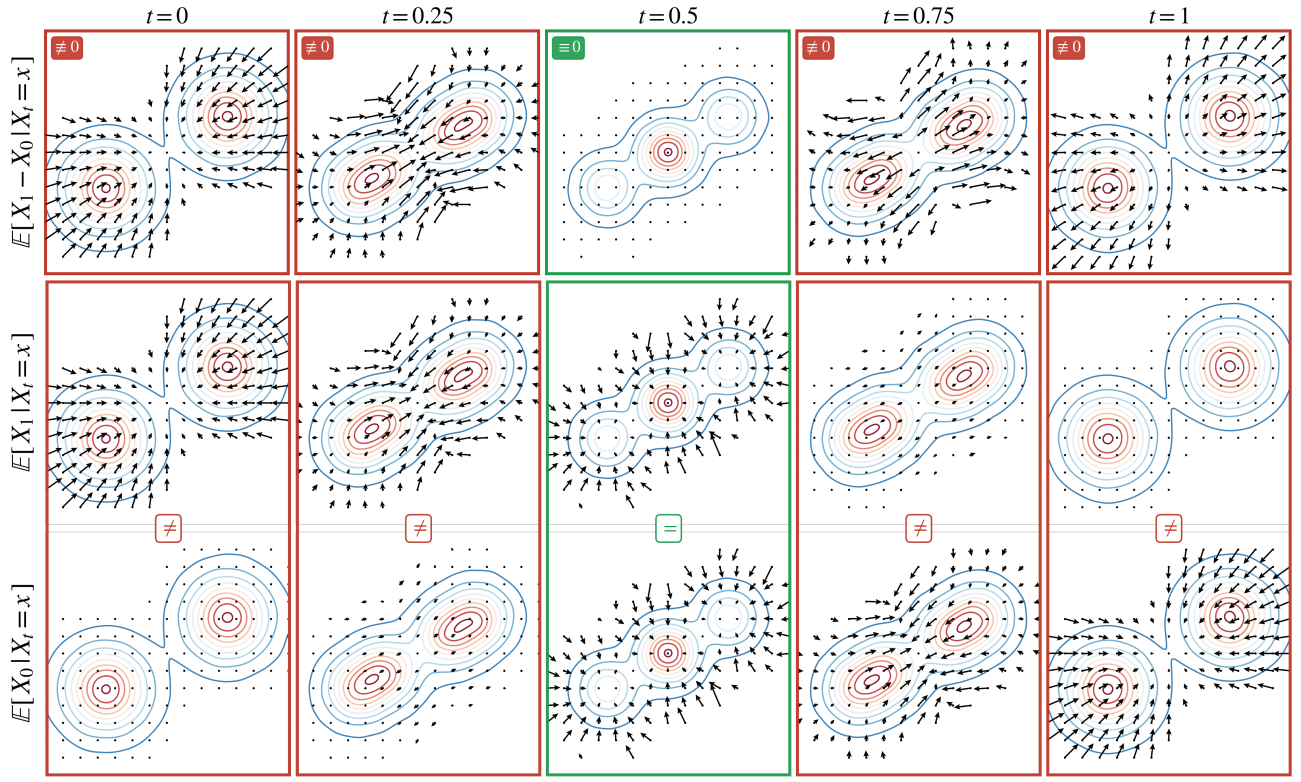


Figure 1. Linear interpolation for a Gaussian toy example. Contours show the interpolated density p_t. *Top row:* The velocity field $v_t(x) = \mathbb{E}[X_1 - X_0 \mid X_t = x]$ vanishes only at $t = 1/2$ and is nonzero away from the midpoint. *Bottom rows:* the endpoint denoisers are related by time reversal, $\mathbb{E}[X_0 \mid X_t = x] = \mathbb{E}[X_1 \mid X_{1-t} = x]$, rather than by equality at a fixed time.

natural next question is whether analogous discrepancies can be constructed using non-midpoint times.

3.2. Time-integrated Midpoint Divergence

A naive way to extend the midpoint construction over the full Flow Matching path is to integrate the squared norm of the Flow Matching velocity field over time:

$$D^{\text{naive}}_{\text{t-mid}}(p_0, p_1) := \int_0^1 \mathbb{E}_{X_t} \left\| \mathbb{E}_{X_0, X_1}[X_1 - X_0 \,|\, X_t] \right\|_2^2 \, dt. \quad (3)$$

However, this naive functional is not a valid discrepancy: for $t \neq 1/2$, the Flow Matching velocity field v_t generally does not vanish even when $p_0 = p_1$. Figure 1 (top row) illustrates this failure mode in a Gaussian toy example: under linear interpolation, the conditional velocity field vanishes at the midpoint but remains nonzero at the other displayed times. This also can be seen by rewriting the velocity as a difference of two posterior endpoint means, which we refer to as *denoisers*:

$$\begin{aligned} v_t(x) &= \mathbb{E}_{X_0, X_1}\left[X_1 - X_0 \mid X_t = x\right] \\ &= \mathbb{E}_{X_1}\left[X_1 \mid X_t = x\right] - \mathbb{E}_{X_0}\left[X_0 \mid X_t = x\right]. \end{aligned}$$

Even when $p_0 = p_1$, these two denoisers are generally different away from the midpoint. For example, at $t = 0$ we have $X_t = X_0$, and hence

$$\mathbb{E}_{X_1}\left[X_1 \mid X_0 = x\right] = \mathbb{E}_{X_1} X_1, \;\; \mathbb{E}_{X_0}\left[X_0 \mid X_0 = x\right] = x.$$

Thus, given $X_0 = x$, the squared-error optimal estimator of X_0 is x itself, whereas the optimal reconstruction of the independent endpoint X_1 is only its mean. The same phenomenon occurs at other non-midpoint times: the closer t is to one endpoint, the easier it is to reconstruct that endpoint from X_t, and the harder it is to reconstruct the other endpoint. Thus the two denoisers are intrinsically asymmetric away from $t = 1/2$. This phenomenon is visually illustrated in Figure 1 (bottom rows). To remove this orientation bias, we introduce a *randomly flipped interpolation*. Let $B \sim \text{Bernoulli}(1/2)$ be independent of (X_0, X_1), and define

$$\widetilde{X}_t = \begin{cases} X_t = (1-t)X_0 + tX_1, & B = 0, \\ X_{1-t} = tX_0 + (1-t)X_1, & B = 1. \end{cases} \quad (4)$$

Importantly, the flip variable B is not observed: we condition only on the value of $\widetilde{X}_t$. This randomization makes the two endpoints equally close, in distribution, to the observed interpolation point. Consequently, under the case $p_0 = p_1$, neither endpoint has an intrinsic reconstruction advantage, and the conditional displacement vanishes. This is illustrated in Figure 2 and formalized in the following proposition.

Proposition 3.2 (Flip-induced symmetry)**.** *Let X_0 and X_1 be independent samples from the same distribution p with finite first moment. Then, for every $t \in [0, 1]$, the symmetrized*

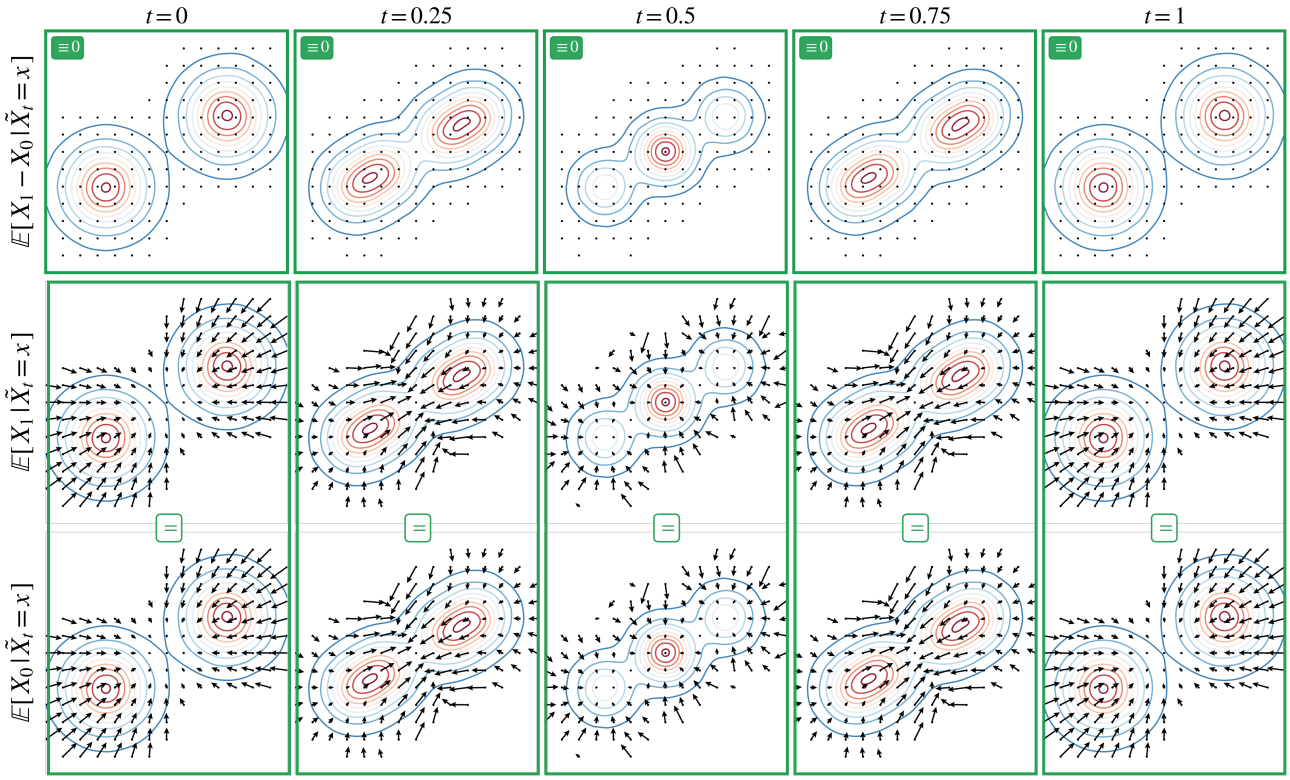


Figure 2. Randomly flipped interpolation on the same Gaussian toy example. Contours show the symmetrized density $\tilde{p}_t$. *Top row:* The symmetrized displacement field $\tilde{v}_t(x) = \mathbb{E}[X_1 - X_0 \mid \tilde{X}_t = x]$ is identically zero for every displayed t. *Bottom rows:* Under the same symmetrized observation model, the endpoint denoisers coincide pointwise for every t, $\mathbb{E}[X_0 \mid \tilde{X}_t = x] = \mathbb{E}[X_1 \mid \tilde{X}_t = x]$.

displacement field

$$\widetilde{v}_t(x) := \mathbb{E}_{X_0,X_1}[X_1 - X_0 \mid \widetilde{X}_t = x] = 0 \qquad (5)$$

for $\widetilde{p}_t$-a.e. x, where $\widetilde{X}_t \sim \widetilde{p}_t := (p_t + p_{1-t})/2$.

At $t = 1/2$, the two branches of the flipped observation coincide, so $\widetilde{X}_{1/2} = X_{1/2}$ regardless of B:

$$\begin{aligned}\widetilde{v}_{1/2}(x) &= \mathbb{E}_{X_0,X_1}[X_1 - X_0 \mid \widetilde{X}_{1/2} = x] \\ &= \mathbb{E}_{X_0,X_1}[X_1 - X_0 \mid X_{1/2} = x] = v_{1/2}(x).\end{aligned}$$

Thus, the flipped construction coincides with the original one at $t = 1/2$.

Since the symmetrized displacement field vanishes for all t, it is natural to define a discrepancy by integrating its squared norm over time.

Definition 3.2 (Time-integrated Midpoint Divergence)**.** *For any two distributions p_0 and p_1 we define time-integrated Midpoint Divergence as the following functional*

$$D_{\text{t-mid}}(p_0, p_1) := \int_0^{1/2} \mathbb{E}_{\widetilde{X}_t} \left\| \mathbb{E}_{X_0,X_1}[X_1 - X_0 \mid \widetilde{X}_t] \right\|_2^2 dt$$

We integrate only over $[0, 1/2]$ because, under the random flip construction, the discrepancy at time t is the same as at time $1-t$; integrating over $[0, 1]$ would therefore count each contribution twice. As in the single-time midpoint case, this functional is a valid divergence.

Theorem 3.2 (Definiteness of the time-integrated Midpoint Divergence)**.** *Let $X_0 \sim p_0$ and $X_1 \sim p_1$ be independent with finite second moments. Then*

$$D_{\text{t-mid}}(p_0, p_1) = 0 \quad \Longleftrightarrow \quad p_0 = p_1.$$

Figure 2 shows the same toy example after random time reversal: the symmetrized displacement field is zero for every t, and the two endpoint denoisers coincide pointwise.

3.3. Generalized Midpoint Divergence

In the flipped construction, the symmetrized displacement field $\widetilde{v}_t(x)$ in (5) is no longer the Flow Matching velocity field (1). Nevertheless, it can still be interpreted as the difference between two posterior endpoint denoisers under the symmetrized observation model:

$$\begin{aligned}\widetilde{v}_t(x) &= \mathbb{E}_{X_0,X_1}[X_1 - X_0 \mid \widetilde{X}_t = x] \\ &= \mathbb{E}_{X_1}[X_1 \mid \widetilde{X}_t = x] - \mathbb{E}_{X_0}[X_0 \mid \widetilde{X}_t = x].\end{aligned}$$

This perspective decouples the construction from the specific linear interpolation used in Flow Matching and suggests a broader class of symmetric interpolation paths.

We now extend the construction to stochastic interpolants defined in §2.2:

$$X_t = I_t(X_0, X_1) + \sigma_t \epsilon, \qquad \epsilon \sim \mathcal{N}(0, I). \tag{6}$$

We additionally require the interpolant to be symmetric, namely

$$I_t(x_0, x_1) = I_{1-t}(x_1, x_0), \qquad \sigma_t = \sigma_{1-t}.$$

Let $B \sim \text{Bernoulli}(1/2)$ be independent of (X_0, X_1, ϵ). We define the flipped observation by

$$\widetilde{X}_t = \begin{cases} X_t = I_t(X_0, X_1) + \sigma_t \epsilon, & B = 0, \\ X_{1-t} = I_{1-t}(X_0, X_1) + \sigma_{1-t}\epsilon, & B = 1, \end{cases} \tag{7}$$

This symmetrized observation model leads to the following generalized Midpoint Divergence:

$$D^{I,\sigma}_{\text{t-mid}}(p_0, p_1) := \int_0^{1/2} \mathbb{E}_{\widetilde{X}_t} \left\| \mathbb{E}_{X_0, X_1}[X_1 - X_0 \mid \widetilde{X}_t] \right\|_2^2 dt. \tag{8}$$

This reduces to the time-integrated midpoint divergence for the Flow Matching linear interpolant. The next theorem shows that this generalized construction still defines a valid divergence.

Theorem 3.3 (Definiteness of the generalized Midpoint Divergence)**.** *Let $X_0 \sim p_0$ and $X_1 \sim p_1$ be independent with finite second moments. Fix any symmetric stochastic interpolant (6). Then*

$$D^{I,\sigma}_{\text{t-mid}}(p_0, p_1) = 0 \quad \Longleftrightarrow \quad p_0 = p_1.$$

Theorem 3.3 shows that the generalized Midpoint Divergence can serve as a principled training objective: minimizing it against the data distribution identifies the target distribution uniquely. We now use this divergence to construct our Midpoint Generative Models.

3.4. Midpoint Generative Models

The generalized midpoint divergence provides a natural approach for generative modeling. Let p_θ be the distribution induced by a generator G_θ applied to a latent prior p_z:

$$Z \sim p_z, \qquad X_0 = G_\theta(Z), \qquad X_0 \sim p_\theta.$$

We pair generated samples with data samples $X_1 \sim p_{\text{data}}$, and define $\widetilde{X}_t$ using the flipped stochastic interpolant in (7). The generalized midpoint divergence then provides a natural training objective

$$\min_\theta D^{I,\sigma}_{\text{t-mid}}(p_\theta, p_{\text{data}}). \tag{9}$$

By Theorem 3.3, the divergence vanishes if and only if $p_\theta = p_{\text{data}}$. Directly optimizing (9), however, is intractable because the symmetrized displacement field

$$\mathbb{E}_{X_0, X_1}[X_1 - X_0 \mid \widetilde{X}_t]$$

is unknown. We therefore use the following variational representation.

Proposition 3.3 (Variational midpoint objective)**.** *For fixed p_θ, the generalized midpoint divergence admits the variational form $D^{I,\sigma}_{\text{t-mid}}(p_\theta, p_{\text{data}}) =$*

$$\max_{f_t} \int_0^{1/2} \mathbb{E}_{X_0, X_1, B, \epsilon} \left[2\langle f_t(\widetilde{X}_t), X_1 - X_0 \rangle - \| f_t(\widetilde{X}_t) \|_2^2 \right] dt,$$

where $\widetilde{X}_t$ is defined according to (7) and the maximum is over square-integrable vector-valued functions f_t. Moreover, the maximizer is given by

$$f_t^\star(x) = \mathbb{E}_{X_0, X_1} \left[X_1 - X_0 \mid \widetilde{X}_t = x \right] = \widetilde{v}_t(x),$$

for almost every t and $\widetilde{p}_t$-almost every x.

Using Proposition 3.3 and parameterizing the displacement field by a neural network f_ψ, we obtain the practical minimax objective

$$\min_\theta \max_\psi \ \mathcal{L}(\theta, \psi) := \tag{10}$$
$$\int_0^{1/2} \mathbb{E} \left[2\langle f_\psi(t, \widetilde{X}_t), X_1 - X_0 \rangle - \| f_\psi(t, \widetilde{X}_t) \|_2^2 \right] dt.$$

In practice, we alternate between updating the displacement field model to approximate the symmetrized displacement field and updating the generator to minimize the resulting variational midpoint objective. The full stochastic training procedure is summarized in Algorithm 1.

Importance of flipping and time integration. The variational formulation above can be instantiated not only with the generalized Midpoint Divergence, but also with simpler alternatives such as the midpoint-only divergence D_{mid} in (2) and the naive unflipped time-integrated objective $D^{\text{naive}}_{\text{t-mid}}(p_0, p_1)$ in (3); details are provided in Appendix C. These alternatives allow us to ablate the two key ingredients of MGM: random flipping and time integration.

Figure 3 shows the resulting samples on the Swiss roll toy dataset. The midpoint-only objective learns the coarse geometry of the target, but produces noisy samples. One possible explanation is that, at $t = 1/2$, the displacement field model f_ψ only observes the mixed interpolation $x_{1/2} = (x_0 + x_1)/2$, in which the generator output x_0 appears only through an averaged state. This may provide a weaker signal for correcting fine-grained errors in generator samples. In contrast, the flipped time-integrated objective

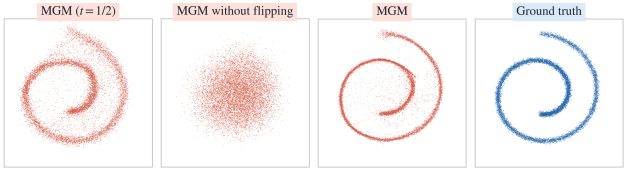


Figure 3. Toy samples illustrating the role of the MGM construction. Full MGM recovers the target distribution, while the midpoint-only objective at $t = 1/2$ remains valid but produces noisier samples. In contrast, naively integrating the unflipped midpoint objective over time fails.

includes times closer to the endpoints. For small t, the symmetrized observation contains samples close to either X_0 or X_1, while preserving endpoint symmetry through the random flip. Thus, time integration gives the displacement field access to a richer family of observations, including near-generator samples, which can provide a stronger signal for fine-grained details.

The naive unflipped time-integrated objective fails, as expected from the orientation bias of non-midpoint Flow Matching velocities. The flipped time-integrated objective, however, accurately recovers the target distribution. This comparison supports the full MGM construction: random flipping makes the displacement field vanish when $p_0 = p_1$, while time integration provides a stronger training signal than the midpoint alone.

4. Related Work

We discuss three main axes of work related to MGM: distillation of pre-trained diffusion and flow models, direct training of fast generators from diffusion/flow principles, and GANs.

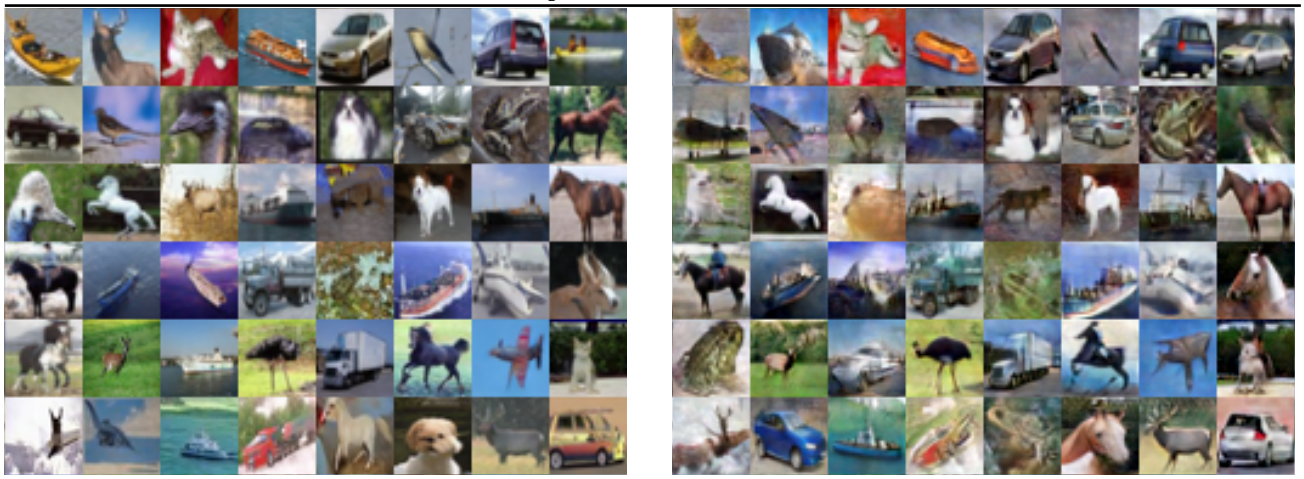
Distillation from pre-trained diffusion and flow models. A common route to fast generation is to distill a pre-trained diffusion or flow model into a one- or few-step student. One branch exploits PF-ODE consistency: teacher samples follow deterministic trajectories, so a pair (x_t, t) identifies the remaining trajectory including the final point. These methods train a student to reproduce two or more teacher-sampler steps or to make predictions at different points on the same teacher trajectory agree (Salimans & Ho, 2022; Song et al., 2023; Song & Dhariwal, 2024; Lu & Song, 2025; Kim et al., 2024; Berthelot et al., 2023; Geng et al., 2025b; Lee et al., 2025). A second branch parameterizes $p_\theta(x_0)$ through a stochastic generator and optimizes losses derived from a pre-trained teacher (Yin et al., 2024b; Sauer et al., 2024; Luo et al., 2023; Yin et al., 2024a; Xu et al., 2025; Zhou et al., 2024; Luo et al., 2024; Gushchin et al., 2025; Huang et al., 2024; Shlenskii & Korotin, 2026). These losses usually lead to adversarial optimization and often require an additional estimator for the generator distribution, such as a generator score, generator flow, or critic. Many of these works can be viewed through the inverse-distillation lens of Universal Inverse Distillation (Kornilov et al., 2026). *In contrast, MGM does not distill a pre-trained diffusion or flow teacher; it trains directly from data using the proposed Midpoint Divergence between generated samples and real data, without any teacher.*

Training one- and few-step generators from scratch using diffusion/flow properties. Another line of work trains fast samplers directly from data, without a pretrained teacher as the training signal. These methods use the fact that finite-time updates along the same generative ODE trajectory must be mutually consistent: an update from s to t should agree with updates that pass through intermediate times. Their common goal is to obtain such finite-time integrators, parameterized as endpoint-consistent predictors, two-time trajectory maps, finite-interval updates, or average velocities (Song et al., 2023; Song & Dhariwal, 2023; Lu & Song, 2025; Kim et al., 2024; Frans et al., 2025; Geng et al., 2025a). Some methods, such as Consistency Models and Consistency Trajectory Models, also support teacher distillation, but their direct-training variants use only data and noise samples (Song et al., 2023; Kim et al., 2024). Flow Map Matching gives a general view of many such objectives as learning maps between pairs of times in diffusion or stochastic-interpolant dynamics (Boffi et al., 2025). MGM is close in spirit because it also trains from scratch using diffusion and flow principles. *Unlike these approaches, MGM does not learn a finite-time integrator; it directly parameterizes the generated data distribution and compares it to real data samples using the proposed Midpoint Divergence.*

GANs. Classical GANs (Goodfellow et al., 2014) train a generator against a scalar discriminator, with f-GAN (Nowozin et al., 2016) and Wasserstein GAN (Arjovsky et al., 2017) connecting discriminator-generator minimax objectives to standard divergences. Diffusion-GAN variants (Xiao et al., 2022; Wang et al., 2023) combine adversarial objectives with diffusion ideas, for example by discriminating noised samples or modeling large denoising steps adversarially. MGM also has a minimax training form, but uses a vector-valued critic $f_\psi(t, x)$ trained toward endpoint displacements under a symmetrized stochastic interpolant. *Thus, MGM optimizes a regression-based generalized Midpoint Divergence, not a classification-based adversarial loss.*

5. Experiments

In this section, we evaluate MGM on unconditional CIFAR-10 32×32 generation. In §5.1, we describe the dataset, evaluation protocol, and model parameterization used in our CIFAR-10 experiments. In §5.2, we ablate the main design choices of MGM, including warmup duration, interpolant stochasticity, critic stochasticity, and the use of the generalized Midpoint Divergence. Finally, in §5.3, we compare MGM against reported 1-NFE CIFAR-10 baselines.



(a) MGM *(b)* MGM, $t = 1/2$ only

Figure 4. Crops from uncurated CIFAR-10 sample grids generated with one network evaluation. The time-integrated MGM objective preserves more fine-grained structure than the midpoint-only variant.

Additional experimental details and results are provided in Appendix B.

5.1. Experimental Setup

Datasets and Evaluation Protocol. We evaluate MGM on unconditional CIFAR-10 32×32 (Krizhevsky et al., 2009). Following prior works, we report the Frechet Inception Distance (FID) (Heusel et al., 2017) computed from 50K generated samples against the CIFAR-10 training set statistics.

Implementation Details. We use the EDM `ddpm++` backbone for unconditional CIFAR-10 (Karras et al., 2022) for both the generator G_θ and the displacement field f_ψ models. We use generalized Midpoint Divergence (§3.3) with the linear interpolant $I_t(x_0, x_1) = (1-t)x_0 + tx_1$ and the stochastic schedule $\sigma_t = \sqrt{\sigma t(1-t)}$. Before MGM training, we initialize the one-step generator with a warmup stage that trains the denoising objective restricted to diffusion times $t \in [0, 2.5]$. The generator is then initialized from the model evaluated at time $t = 2.5$, while the MGM displacement field receives interpolation times $t \in [0, 0.5]$. Additional warmup details are provided in Appendix B.

5.2. Ablation Study

We ablate four design choices in MGM: how long to run the denoising-objective warmup, how much noise to use in the generator interpolant, whether the displacement field should observe noisy interpolants, and whether the objective should integrate over time or use only the midpoint. The corresponding FID results are reported in Table 1, with the four ablations separated in Tables 1.a–1.d.

Warmup duration. We vary the number of denoising-objective warmup steps before MGM training (Table 1.a). Performance improves rapidly from 5K to 50K warmup steps (25% of the original denoising training procedure), while longer warmups do not provide further gains.

Interpolant stochasticity. We ablate different values of σ in the stochastic schedule $\sigma_t = \sqrt{\sigma t(1-t)}$ (Table 1.b). The deterministic setting $\sigma = 0$ performs poorly, indicating that noise is important for stable training. This is consistent with the common observation in image-to-image diffusion models that deterministic bridges are weak and that adding bridge noise improves performance (Liu et al., 2022b; Zhou et al., 2023; Zheng et al., 2024; He et al., 2024; Wang et al., 2025b). Small nonzero noise gives the best results, while excessive stochasticity degrades FID.

Critic stochasticity. We next decouple the generator and critic interpolants (Table 1.c). When updating the generator, we sample $x_t^{\text{gen}} = I_t(x_0, x_1)$ without adding noise. When training the displacement field, we use noisy observations $x_t^{\text{critic}} = I_t(x_0, x_1) + \sqrt{\sigma t(1-t)}z$ with $z \sim \mathcal{N}(0, I)$, and vary the critic noise scale σ. This critic-only stochasticity gives the lowest FID in our ablations.

Generalized Midpoint Divergence. We compare the time-integrated objective with a midpoint-only variant that fixes $t = 1/2$ (Table 1.d). The midpoint-only setting is substantially worse across all critic noise levels. Using only the midpoint loses fine-grained details and degrades sample quality (see Fig. 4), while integrating over multiple interpolation times provides a stronger learning signal.

5.3. Baseline Comparison

We compare MGM with other one-step CIFAR-10 generators, separating methods by the training signal used to learn the one-step model. A method is counted as using a teacher-model training signal if a separately pretrained model, or trajectories/targets generated by such a model, appears in the student generator loss. We do not count mere initialization or self-pretraining unless the pretrained model is

Warmup	FID
5K	6.24
10K	3.73
20K	2.69
50K	**2.34**
100K	2.42
200K	2.66

(a) **Warmup duration**

σ	FID
0.00	5.87
0.01	2.58
0.05	**2.46**
0.20	2.78
0.50	3.55

(b) **Stochasticity**

Critic σ	FID
0.00	5.87
0.01	**2.27**
0.05	2.34
0.20	2.98
0.50	56.58

(c) **Critic-only noise**

Critic σ	FID
0.00	14.75
0.01	**13.55**
0.05	19.76
0.20	68.62
0.50	120.48

(d) **Midpoint-only training**

Table 1. **Ablation study on unconditional CIFAR-10** 32×32 **generation.** FID-50K is reported for 1-NFE sampling, and the best entry within each ablation is marked in gray.

w/ teacher-model training signal		w/o teacher-model training signal	
Method	FID	Method	FID
CTM (Kim et al., 2024)	**1.87**	MGM (Ours)	**2.27**
SiD (Zhou et al., 2024)	1.92	DiffRatio-DiJS (Chen et al., 2025)	2.39
TCM (Lee et al., 2025)	2.46	iCT-deep (Song & Dhariwal, 2024)	2.51
CMT w/ ECT (Hu et al., 2025)	2.74	iCT (Song & Dhariwal, 2024)	2.83
CD (Song et al., 2023)	3.55	sCT (Lu & Song, 2025)	2.85
sCD (Lu & Song, 2025)	3.66	MF (Geng et al., 2025a)	2.92
DMD (Yin et al., 2024b)	3.77	Stable CT (Wang et al., 2025a)	2.92
DFNO (Zheng et al., 2023)	3.78	StyleGAN2 w/ ADA (Karras et al., 2020)	2.92
TRACT (Berthelot et al., 2023)	3.78	IMM (Zhou et al., 2025)	3.20
2-Rectified Flow (Liu et al., 2022a)	4.85	VCT (Silvestri et al., 2025)	3.26
PD (Salimans & Ho, 2022)	8.34	ECT (Geng et al., 2025b)	3.60

Table 2. Baseline comparison on CIFAR-10 32×32 one step generation (FID-50K). Diffusion/flow baseline values are from CMT (Hu et al., 2025), the StyleGAN2+ADA value is from Xiao et al. (2022), and the DiffRatio-DiJS value is from Chen et al. (2025). For each group, the best entry is marked in gray.

subsequently used as a loss target, boundary condition, or trajectory generator. The diffusion and flow baseline values are taken from the CIFAR-10 1-NFE entries collected by CMT (Hu et al., 2025), the StyleGAN2+ADA value is taken from Xiao et al. (2022), and the DiffRatio value is taken from Chen et al. (2025). Under the same grouping, MGM trains an auxiliary displacement field, but its learning signal is obtained directly from data rather than from a teacher model. As shown in Table 2, MGM achieves FID 2.27, the best result among methods without a teacher-model training signal.

6. Discussion and Limitations

We introduced *Midpoint Generative Models* (MGM), a framework for training one-step generators by minimizing the generalized Midpoint Divergence. Our construction reframes symmetric stochastic interpolants, originally used to define generative paths, as a basis for comparing endpoint distributions. This yields a definite divergence and a practical variational objective for direct generator training.

As MGM introduces a new training paradigm, several questions remain open. In particular, its performance may depend on the choice of stochastic interpolant, noise schedule, and interpolation-time sampling distribution during training. Understanding how these design choices affect optimization stability and sample quality is an important direction for future work.

Limitations. Our practical training procedure uses a minimax objective with an adversarially trained displacement field. As in other adversarial training methods, this can introduce optimization instability and sensitivity to the balance between generator and displacement-field updates.

Impact Statements

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

References

Albergo, M., Boffi, N. M., and Vanden-Eijnden, E. Stochastic interpolants: A unifying framework for flows and diffusions. *Journal of Machine Learning Research*, 26 (209):1–80, 2025.

Albergo, M. S. and Vanden-Eijnden, E. Building normalizing flows with stochastic interpolants. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=li7qeBbCR1t.

Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein generative adversarial networks. In *International conference on machine learning*, pp. 214–223. Pmlr, 2017.

Berthelot, D., Autef, A., Lin, J., Yap, D. A., Zhai, S., Hu, S., Zheng, D., Talbott, W., and Gu, E. Tract: Denoising diffusion models with transitive closure time-distillation. *arXiv preprint arXiv:2303.04248*, 2023.

Boffi, N. M., Albergo, M. S., and Vanden-Eijnden, E. Flow map matching with stochastic interpolants: A mathematical framework for consistency models. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=cqDH0e6ak2.

Chen, W., Zhang, M., He, J., Ou, Z., Hernández-Lobato, J. M., Schölkopf, B., and Barber, D. Diffratio: Training one-step diffusion models without teacher supervision. *arXiv e-prints*, pp. arXiv–2502, 2025.

Dhariwal, P. and Nichol, A. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Frans, K., Hafner, D., Levine, S., and Abbeel, P. One step diffusion via shortcut models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=OlzB6LnXcS.

Geng, Z., Deng, M., Bai, X., Kolter, J. Z., and He, K. Mean flows for one-step generative modeling. *arXiv preprint arXiv:2505.13447*, 2025a.

Geng, Z., Pokle, A., Luo, W., Lin, J., and Kolter, J. Z. Consistency models made easy. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL https://openreview.net/forum?id=xQVxo9dSID.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

Gu, J., Zhai, S., Zhang, Y., Liu, L., and Susskind, J. M. BOOT: Data-free distillation of denoising diffusion models with bootstrapping. In *ICML 2023 Workshop on Structured Probabilistic Inference & Generative Modeling*, 2023. URL https://openreview.net/forum?id=ZeM7S01Xi8.

Gushchin, N., Li, D., Selikhanovych, D., Burnaev, E., Baranchuk, D., and Korotin, A. Inverse bridge matching distillation. 2025.

He, G., Zheng, K., Chen, J., Bao, F., and Zhu, J. Consistency diffusion bridge models. *Advances in Neural Information Processing Systems*, 37:23516–23548, 2024.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M., and Fleet, D. J. Video diffusion models. *Advances in neural information processing systems*, 35:8633–8646, 2022.

Hu, Z., Lai, C.-H., Mitsufuji, Y., and Ermon, S. Cmt: Mid-training for efficient learning of consistency, mean flow, and flow map models. *arXiv preprint arXiv:2509.24526*, 2025.

Huang, Z., Geng, Z., Luo, W., and Qi, G.-j. Flow generator matching. *arXiv preprint arXiv:2410.19310*, 2024.

Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J., and Aila, T. Training generative adversarial networks with limited data. *Advances in neural information processing systems*, 33:12104–12114, 2020.

Karras, T., Aittala, M., Aila, T., and Laine, S. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35: 26565–26577, 2022.

Kim, D., Lai, C.-H., Liao, W.-H., Murata, N., Takida, Y., Uesaka, T., He, Y., Mitsufuji, Y., and Ermon, S. Consistency trajectory models: Learning probability flow ODE trajectory of diffusion. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=ymjI8feDTD.

Kong, Z., Ping, W., Huang, J., Zhao, K., and Catanzaro, B. Diffwave: A versatile diffusion model for audio synthesis. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=a-xFK8Ymz5J.

Kornilov, N. M., Li, D., Mavrin, T., Leonov, A., Gushchin, N., Burnaev, E., Koshelev, I. S., and Korotin, A. Universal inverse distillation for matching models with real-data supervision (no GANs). In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=8NuN5UzXLC.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

Lee, S., Xu, Y., Geffner, T., Fanti, G., Kreis, K., Vahdat, A., and Nie, W. Truncated consistency models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=ZYDEJEvCbv.

Lipman, Y., Chen, R. T., Ben-Hamu, H., Nickel, M., and Le, M. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Liu, X., Gong, C., and Liu, Q. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022a.

Liu, X., Wu, L., Ye, M., et al. Let us build bridges: Understanding and extending diffusion generative models. In *NeurIPS 2022 Workshop on Score-Based Methods*, 2022b.

Lu, C. and Song, Y. Simplifying, stabilizing and scaling continuous-time consistency models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=LyJi5ugyJx.

Lu, C., Zhou, Y., Bao, F., Chen, J., Li, C., and Zhu, J. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in neural information processing systems*, 35:5775–5787, 2022.

Luo, W., Hu, T., Zhang, S., Sun, J., Li, Z., and Zhang, Z. Diff-instruct: A universal approach for transferring knowledge from pre-trained diffusion models. *Advances in Neural Information Processing Systems*, 36:76525–76546, 2023.

Luo, W., Huang, Z., Geng, Z., Kolter, J. Z., and Qi, G.-j. One-step diffusion distillation through score implicit matching. *Advances in Neural Information Processing Systems*, 37:115377–115408, 2024.

Meng, C., He, Y., Song, Y., Song, J., Wu, J., Zhu, J.-Y., and Ermon, S. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=aBsCjcPu_tE.

Nowozin, S., Cseke, B., and Tomioka, R. f-gan: Training generative neural samplers using variational divergence minimization. *Advances in neural information processing systems*, 29, 2016.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Saharia, C., Chan, W., Chang, H., Lee, C., Ho, J., Salimans, T., Fleet, D., and Norouzi, M. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 conference proceedings*, pp. 1–10, 2022.

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

Sauer, A., Lorenz, D., Blattmann, A., and Rombach, R. Adversarial diffusion distillation. In *European Conference on Computer Vision*, pp. 87–103. Springer, 2024.

Shlenskii, D. and Korotin, A. Overclocking electrostatic generative models. 2026. URL https://openreview.net/forum?id=flo449mncA.

Silvestri, G., Ambrogioni, L., Lai, C.-H., Takida, Y., and Mitsufuji, Y. VCT: Training consistency models with variational noise coupling. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=CMoX0BEsDs.

Song, J., Meng, C., and Ermon, S. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

Song, Y. and Dhariwal, P. Improved techniques for training consistency models. *arXiv preprint arXiv:2310.14189*, 2023.

Song, Y. and Dhariwal, P. Improved techniques for training consistency models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=WNzy9bRDvG.

Song, Y., Dhariwal, P., Chen, M., and Sutskever, I. Consistency models. In *International Conference on Machine Learning*, pp. 32211–32252. PMLR, 2023.

Wang, F.-Y., Geng, Z., and Li, H. Stable consistency tuning: Understanding and improving consistency models. In *ICLR 2025 Workshop on Deep Generative Model in Machine Learning: Theory, Principle and Efficacy*, 2025a. URL https://openreview.net/forum?id=5RoPe2ShXx.

Wang, Y., Yoon, S., Jin, P., Tivnan, M., Song, S., Chen, Z., Hu, R., Zhang, L., Li, Q., Chen, Z., et al. Implicit image-to-image schrödinger bridge for image restoration. *Pattern Recognition*, 165:111627, 2025b.

Wang, Z., Zheng, H., He, P., Chen, W., and Zhou, M. Diffusion-GAN: Training GANs with diffusion. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=HZf7UbpWHuA.

Xiao, Z., Kreis, K., and Vahdat, A. Tackling the generative learning trilemma with denoising diffusion GANs. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=JprM0p-q0Co.

Xu, Y., Nie, W., and Vahdat, A. One-step diffusion models with f-divergence distribution matching. *arXiv preprint arXiv:2502.15681*, 2025.

Yin, T., Gharbi, M., Park, T., Zhang, R., Shechtman, E., Durand, F., and Freeman, B. Improved distribution matching distillation for fast image synthesis. *Advances in neural information processing systems*, 37:47455–47487, 2024a.

Yin, T., Gharbi, M., Zhang, R., Shechtman, E., Durand, F., Freeman, W. T., and Park, T. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6613–6623, 2024b.

Zheng, H., Nie, W., Vahdat, A., Azizzadenesheli, K., and Anandkumar, A. Fast sampling of diffusion models via operator learning. In *International conference on machine learning*, pp. 42390–42402. PMLR, 2023.

Zheng, K., He, G., Chen, J., Bao, F., and Zhu, J. Diffusion bridge implicit models. *arXiv preprint arXiv:2405.15885*, 2024.

Zhou, L., Lou, A., Khanna, S., and Ermon, S. Denoising diffusion bridge models. *arXiv preprint arXiv:2309.16948*, 2023.

Zhou, L., Ermon, S., and Song, J. Inductive moment matching. *arXiv preprint arXiv:2503.07565*, 2025.

Zhou, M., Zheng, H., Wang, Z., Yin, M., and Huang, H. Score identity distillation: Exponentially fast distillation of pretrained diffusion models for one-step generation. In *Forty-first International Conference on Machine Learning*, 2024.

A. Proofs

In this section, we provide proofs for all theorems and propositions in the main part.

A.1. Proof of the Proposition 3.1

Let

$$\Delta := X_1 - X_0, \qquad M := X_{1/2} = \frac{X_0 + X_1}{2}.$$

Since X_0 and X_1 are independent samples from the same distribution,

$$(X_0, X_1) \overset{d}{=} (X_1, X_0).$$

Under this exchange, M is unchanged while Δ changes sign. Hence

$$(\Delta, M) \overset{d}{=} (-\Delta, M).$$

Therefore the conditional law of Δ given M is symmetric about the origin, and so

$$\mathbb{E}[\Delta \mid M] = 0 \quad \text{a.s.}$$

Since $\Delta = X_1 - X_0$ and $M = X_{1/2}$, this is exactly

$$v_{1/2}(x) = \mathbb{E}[X_1 - X_0 \mid X_{1/2} = x] = 0$$

for $p_{1/2}$-almost every x.

A.2. Proof of the Theorem 3.1

Let

$$\Delta := X_1 - X_0, \qquad M := X_{1/2} = \frac{X_0 + X_1}{2}.$$

Then, by Definition (3.1),

$$D_{\text{mid}}(p_0, p_1) = \mathbb{E}\big[\|\mathbb{E}[\Delta \mid M]\|_2^2\big].$$

If $p_0 = p_1$, then Proposition 3.1 gives

$$\mathbb{E}[\Delta \mid M] = 0 \quad \text{a.s.}$$

and hence $D_{\text{mid}}(p_0, p_1) = 0$.

Conversely, suppose that

$$D_{\text{mid}}(p_0, p_1) = 0.$$

Since the integrand is nonnegative,

$$\mathbb{E}[\Delta \mid M] = 0 \quad \text{a.s.}$$

Thus, for every bounded measurable function φ,

$$\mathbb{E}[\Delta\, \varphi(M)] = 0.$$

Because X_0 and X_1 are bounded almost surely, for every $u \in \mathbb{R}^d$ the function

$$\varphi_u(m) := e^{\langle u, m\rangle}$$

is bounded on the support of M. Therefore

$$\mathbb{E}\left[(X_1 - X_0) e^{\langle u, (X_0 + X_1)/2\rangle}\right] = 0.$$

Using independence of X_0 and X_1, this becomes

$$\mathbb{E}\left[X_1 e^{\langle u, X_1\rangle/2}\right] \mathbb{E}\left[e^{\langle u, X_0\rangle/2}\right] = \mathbb{E}\left[X_0 e^{\langle u, X_0\rangle/2}\right] \mathbb{E}\left[e^{\langle u, X_1\rangle/2}\right].$$

Define the moment generating functions

$$G_j(v) := \mathbb{E}\Big[e^{\langle v, X_j\rangle}\Big], \qquad j \in \{0,1\}.$$

Since X_0 and X_1 are bounded, G_0 and G_1 are finite and differentiable on all of $\mathbb{R}^d$, with

$$\nabla G_j(v) = \mathbb{E}\Big[X_j e^{\langle v, X_j\rangle}\Big].$$

Setting $v = u/2$, the preceding identity yields

$$G_0(v)\nabla G_1(v) = G_1(v)\nabla G_0(v) \qquad \text{for all } v \in \mathbb{R}^d.$$

Since $G_0(v) > 0$ and $G_1(v) > 0$,

$$\nabla \log G_1(v) = \nabla \log G_0(v).$$

Hence $\log G_1 - \log G_0$ is constant on $\mathbb{R}^d$. Evaluating at $v = 0$ gives

$$G_0(0) = G_1(0) = 1,$$

so the constant is zero. Thus

$$G_0(v) = G_1(v) \qquad \text{for all } v \in \mathbb{R}^d.$$

By uniqueness of moment generating functions, $p_0 = p_1$. Therefore

$$D_{\text{mid}}(p_0, p_1) = 0 \quad \Longleftrightarrow \quad p_0 = p_1.$$

A.3. Proof of the Proposition 3.2

Fix $t \in [0,1]$ and let

$$\Delta := X_1 - X_0.$$

We need to show that

$$\mathbb{E}[\Delta \mid \widetilde{X}_t] = 0.$$

Since X_0 and X_1 are independent samples from the same distribution and $B \sim \text{Bernoulli}(1/2)$ is independent of them,

$$(X_0, X_1, B) \overset{d}{=} (X_1, X_0, 1-B).$$

Under this transformation, Δ changes sign. On the other hand, the flipped observation

$$\widetilde{X}_t = (1-B)\big((1-t)X_0 + tX_1\big) + B\big(tX_0 + (1-t)X_1\big)$$

is unchanged. Therefore

$$(\Delta, \widetilde{X}_t) \overset{d}{=} (-\Delta, \widetilde{X}_t).$$

It follows that the conditional law of Δ given $\widetilde{X}_t$ is symmetric about the origin, and hence

$$\mathbb{E}[\Delta \mid \widetilde{X}_t] = 0 \quad \text{a.s.}$$

Equivalently,

$$\widetilde{v}_t(x) = \mathbb{E}[X_1 - X_0 \mid \widetilde{X}_t = x] = 0$$

for $\widetilde{p}_t$-almost every x.

A.4. Proof of the Theorem 3.2

Theorem 3.2 is the deterministic linear interpolant special case of Theorem 3.3. Indeed, take

$$I_t(x_0, x_1) := (1-t)x_0 + tx_1, \qquad \sigma_t := 0.$$

Then the flipped stochastic interpolant in (7) reduces to the flipped linear interpolation in (4):

$$\widetilde{X}_t = \begin{cases} (1-t)X_0 + tX_1, & B = 0, \\ tX_0 + (1-t)X_1, & B = 1. \end{cases}$$

Consequently,

$$D^{I,\sigma}_{t\text{-mid}}(p_0, p_1) = D_{t\text{-mid}}(p_0, p_1).$$

The claim therefore follows directly from Theorem 3.1.

A.5. Proof of the Theorem 3.3

Let

$$\Delta := X_1 - X_0, \qquad m_t(x) := \mathbb{E}[\Delta \mid \widetilde{X}_t = x].$$

By definition,

$$D^{I,\sigma}_{t\text{-mid}}(p_0, p_1) = \int_0^{1/2} \mathbb{E}\big[\|m_t(\widetilde{X}_t)\|_2^2\big]\, dt.$$

Nonnegativity is immediate.

First suppose that $p_0 = p_1$. Then X_0 and X_1 are exchangeable. Since $B \sim \mathrm{Bernoulli}(1/2)$ is independent of (X_0, X_1, ϵ), the transformation

$$(X_0, X_1, B, \epsilon) \mapsto (X_1, X_0, 1-B, \epsilon)$$

preserves the joint distribution. Under this transformation, Δ changes sign. On the other hand, the flipped observation (7), which due to symmetry could be rewritten as

$$\widetilde{X}_t = \begin{cases} I_t(X_0, X_1) + \sigma_t \epsilon, & B = 0, \\ I_t(X_1, X_0) + \sigma_t \epsilon, & B = 1, \end{cases}$$

is unchanged. Hence

$$(\Delta, \widetilde{X}_t) \overset{d}{=} (-\Delta, \widetilde{X}_t).$$

Therefore the conditional law of Δ given $\widetilde{X}_t$ is symmetric about the origin, so

$$m_t(\widetilde{X}_t) = \mathbb{E}[\Delta \mid \widetilde{X}_t] = 0 \quad \text{a.s.}$$

for every $t \in [0, 1/2]$. Thus

$$D^{I,\sigma}_{t\text{-mid}}(p_0, p_1) = 0.$$

Conversely, suppose that

$$D^{I,\sigma}_{t\text{-mid}}(p_0, p_1) = 0.$$

Since the integrand is nonnegative,

$$m_t(\widetilde{X}_t) = \mathbb{E}[\Delta \mid \widetilde{X}_t] = 0 \quad \text{a.s.}$$

for almost every $t \in [0, 1/2]$. Choose a sequence $t_n \downarrow 0$ from this full-measure set. By the endpoint continuity of the stochastic interpolant,

$$\widetilde{X}_{t_n} \longrightarrow Z \quad \text{a.s.},$$

where

$$Z := \begin{cases} X_0, & B = 0, \\ X_1, & B = 1. \end{cases}$$

Let $\varphi : \mathbb{R}^d \to \mathbb{R}$ be bounded and continuous. For each n, since

$$m_{t_n}(\widetilde{X}_{t_n}) = \mathbb{E}[\Delta \mid \widetilde{X}_{t_n}] = 0 \quad \text{a.s.},$$

we have, by the tower property,

$$\begin{aligned}\mathbb{E}\big[\Delta\, \varphi(\widetilde{X}_{t_n})\big] &= \mathbb{E}\big[\mathbb{E}[\Delta \mid \widetilde{X}_{t_n}]\varphi(\widetilde{X}_{t_n})\big] \\ &= \mathbb{E}\big[m_{t_n}(\widetilde{X}_{t_n})\varphi(\widetilde{X}_{t_n})\big] \\ &= 0.\end{aligned}$$

Since $\Delta \in L^1$ and φ is bounded, dominated convergence gives

$$\mathbb{E}[\Delta\, \varphi(Z)] = 0.$$

By a monotone-class argument, the same identity holds for every bounded Borel function φ.

Expanding over the two values of B, we obtain

$$0 = \frac{1}{2}\mathbb{E}\big[(X_1 - X_0)\varphi(X_0)\big] + \frac{1}{2}\mathbb{E}\big[(X_1 - X_0)\varphi(X_1)\big].$$

Let

$$\bar{x}_0 := \mathbb{E}[X_0], \qquad \bar{x}_1 := \mathbb{E}[X_1].$$

Using independence of X_0 and X_1, the previous identity becomes

$$\int \varphi(x)(\bar{x}_1 - x)\, p_0(dx) + \int \varphi(x)(x - \bar{x}_0)\, p_1(dx) = 0$$

for every bounded Borel φ. Taking $\varphi \equiv 1$ gives

$$\bar{x}_0 = \bar{x}_1.$$

Denote the common value by $\bar{x}$. Then

$$\int \varphi(x)(x - \bar{x})\, (p_1 - p_0)(dx) = 0$$

for every bounded Borel φ.

Let

$$\mu := p_1 - p_0.$$

We have shown that

$$(x - \bar{x})\, \mu(dx) = 0$$

as a vector-valued signed measure. We now show that this implies $\mu = 0$.

Fix a coordinate $j \in \{1, \dots, d\}$ and an integer $n \geq 1$, and define

$$A_{j,n} := \{x \in \mathbb{R}^d : \ |x_j - \bar{x}_j| \geq 1/n\}.$$

For any Borel set $A \subseteq A_{j,n}$, the function

$$\varphi(x) := \frac{\mathbf{1}_A(x)}{x_j - \bar{x}_j}$$

is bounded and Borel. Applying the j-th coordinate of the signed-measure identity gives

$$\mu(A) = \int \varphi(x)(x_j - \bar{x}_j)\, \mu(dx) = 0.$$

Thus μ vanishes on every Borel subset of $A_{j,n}$. Since

$$\mathbb{R}^d \setminus \{\bar{x}\} = \bigcup_{j=1}^{d} \bigcup_{n=1}^{\infty} A_{j,n},$$

we have

$$\mu(\mathbb{R}^d \setminus \{\bar{x}\}) = 0.$$

Hence μ is supported on $\{\bar{x}\}$. But μ is the difference of two probability measures, so

$$\mu(\mathbb{R}^d) = p_1(\mathbb{R}^d) - p_0(\mathbb{R}^d) = 0.$$

Therefore $\mu(\{\bar{x}\}) = 0$, and consequently $\mu \equiv 0$. Hence $p_1 \equiv p_0$.

Combining both directions,

$$D^{I,\sigma}_{t\text{-mid}}(p_0, p_1) = 0 \quad \Longleftrightarrow \quad p_0 = p_1.$$

A.6. Proof of the Proposition 3.3

Let

$$\Delta := X_1 - X_0, \qquad m_t(x) := \mathbb{E}[\Delta \mid \widetilde{X}_t = x].$$

Then

$$D^{I,\sigma}_{t\text{-mid}}(p_\theta, p_{\text{data}}) = \int_0^{1/2} \mathbb{E}\big[\|m_t(\widetilde{X}_t)\|_2^2\big]\, dt.$$

Fix $t \in [0, 1/2]$ and let f_t be any square-integrable vector-valued function. Recall that

$$m_t(\widetilde{X}_t) = \mathbb{E}[\Delta \mid \widetilde{X}_t].$$

Since $f_t(\widetilde{X}_t)$ is measurable with respect to $\widetilde{X}_t$, the tower property gives

$$\begin{aligned}
\mathbb{E}\big[\langle f_t(\widetilde{X}_t), \Delta\rangle\big] &= \mathbb{E}\Big[\mathbb{E}\big[\langle f_t(\widetilde{X}_t), \Delta\rangle \mid \widetilde{X}_t\big]\Big] \\
&= \mathbb{E}\Big[\Big\langle f_t(\widetilde{X}_t), \mathbb{E}[\Delta \mid \widetilde{X}_t]\Big\rangle\Big] \\
&= \mathbb{E}\big[\langle f_t(\widetilde{X}_t), m_t(\widetilde{X}_t)\rangle\big].
\end{aligned}$$

The second term does not involve Δ, so it is unchanged. Therefore

$$\mathbb{E}\left[2\langle f_t(\widetilde{X}_t), \Delta\rangle - \|f_t(\widetilde{X}_t)\|_2^2\right] = \mathbb{E}\left[2\langle f_t(\widetilde{X}_t), m_t(\widetilde{X}_t)\rangle - \|f_t(\widetilde{X}_t)\|_2^2\right].$$

Completing the square,

$$2\langle f_t, m_t\rangle - \|f_t\|_2^2 = \|m_t\|_2^2 - \|f_t - m_t\|_2^2.$$

Therefore

$$\mathbb{E}\left[2\langle f_t(\widetilde{X}_t), \Delta\rangle - \|f_t(\widetilde{X}_t)\|_2^2\right] = \mathbb{E}\big[\|m_t(\widetilde{X}_t)\|_2^2\big] - \mathbb{E}\big[\|f_t(\widetilde{X}_t) - m_t(\widetilde{X}_t)\|_2^2\big].$$

The first term is nonnegative and does not depend on f_t, so the expression is maximized when

$$f_t(\widetilde{X}_t) = m_t(\widetilde{X}_t) \quad \text{a.s.}$$

Equivalently,

$$f_t^\star(x) = m_t(x) = \mathbb{E}[X_1 - X_0 \mid \widetilde{X}_t = x]$$

for $\widetilde{p}_t$-almost every x.

Integrating the pointwise variational identity over $t \in [0, 1/2]$ yields

$$D^{I,\sigma}_{t\text{-mid}}(p_\theta, p_{\text{data}}) = \max_{f_t} \int_0^{1/2} \mathbb{E}\left[2\langle f_t(\widetilde{X}_t), X_1 - X_0\rangle - \|f_t(\widetilde{X}_t)\|_2^2\right] dt.$$

This proves the variational representation and identifies the maximizer.

B. Additional Experimental Details

We provide additional implementation details for the experiments reported in the main text. Our implementation builds on the EDM[1] and SiD[2] codebases.

Setup. For CIFAR-10 32×32 experiments, we use the `ddpm++` architecture from EDM for both the generator and the displacement field. Each run is trained on two H100 GPUs for aobut two days.

Hyperparameters. We use the same optimizer settings for both networks. Specifically, both the generator and the displacement fieldare optimized with Adam using the default hyperparameters, except that we set $\beta_1 = 0.0$ and use a learning rate of 10^{-5}. The batch size is 256. Unless otherwise stated, the generator and displacement field are updated with a one-to-one update ratio. For evaluation, we maintain an exponential moving average of the generator weights with decay 0.999.

Warmup. Before MGM training, we initialize the generator with a short denoising warmup based on the EDM training objective and hyperparameters. Unlike standard EDM training, which samples diffusion noise levels over a much wider range, we restrict the diffusion time/noise level to $\tau \in (0, 2.5]$ instead of $\tau \in (0, 80]$. This warmup is used only to obtain a reasonable initialization for the one-step generator; we do not aim to train a full diffusion model. Obtained warmuped network weights $P_\eta(x, t)$ we use it to initialize displacement field: $f_\psi(x, t) \leftarrow P_\eta(x, t)$ and generator $G_\theta(z) \leftarrow P_\eta(z, t = 2.5)$ and $z \sim \mathcal{N}(0, Id)$.

Concretely, EDM samples log-noise values

$$n \sim \mathcal{N}(-1.2, 1.2^2), \qquad \tau = \exp(n).$$

To restrict the sampled values to $\tau \leq \tau_{\max}$ with $\tau_{\max} = 2.5$, we reflect the log-noise value around $\log \tau_{\max}$:

$$s = \begin{cases} n, & n \leq \log \tau_{\max}, \\ 2 \log \tau_{\max} - n, & n > \log \tau_{\max}, \end{cases} \qquad \tau = \exp(s).$$

This folded log-normal sampling rule preserves the low-noise part of the EDM distribution while ensuring that all warmup samples satisfy $\tau \in (0, 2.5]$.

In our main experiments, we initialize from warmup weights obtained after approximately six hours of denoising warmup. This stage accounts for only about 18% of the total wall-clock training time, so most of the computation is spent on MGM training rather than on the initialization stage.

C. Algorithm

In this section, we provide a detailed description of the Midpoint Generative Models training procedure. Algorithm 1 gives pseudocode, while Algorithm 2 provides a corresponding Python implementation sketch.

Simpler variants of our objective. The same training template can also be adapted to simpler variants of our objective, including the midpoint-only divergence D_{mid} in (2) and the naive unflipped time-integrated objective $D^{\text{naive}}_{\text{t-mid}}$ in (3). To obtain the midpoint-only variant, one fixes $t = 1/2$ throughout training instead of sampling $t \sim \mathcal{U}[0, 1/2]$. To obtain the naive unflipped time-integrated variant, one removes the random flip by setting $B \equiv 0$ and samples t over the full interval $[0, 1]$ rather than $[0, 1/2]$.

[1] https://github.com/NVlabs/edm
[2] https://github.com/mingyuanzhou/SiD

Algorithm 1 Midpoint Generative Models (MGM)

Generator G_θ, displacement field f_ψ, symmetric interpolant (I_t, σ_t)

repeat

 # Update displacement field f_ψ

Sample $z \sim p_z$, $x_1 \sim p_{\text{data}}$, $t \sim \mathcal{U}[0, 1/2]$, $b \sim \text{Bernoulli}(1/2)$, $\epsilon \sim \mathcal{N}(0, I)$.

Set $x_0 = G_\theta(z)$

Define

$$x_t = \mathcal{I}_t(x_0, x_1) + \sigma_t \epsilon \qquad x_{1-t} = \mathcal{I}_{1-t}(x_0, x_1) + \sigma_{1-t}\epsilon$$

Set the flipped observation $\tilde{x}_t = (1-b)x_t + b x_{1-t}$

Update ψ via:

$$\hat{L}_\psi = \|f_\psi(\tilde{x}_t, t) - (x_1 - x_0)\|_2^2$$

$$\psi \leftarrow \psi - \eta \nabla_\psi \hat{L}_\psi$$

 # Update generator G_θ

Sample a fresh batch and recompute x_0, x_1, and $\tilde{x}_t$ as above.

Update θ via:

$$\hat{L}_\theta = 2\langle f_\psi(\tilde{x}_t, t), x_1 - x_0\rangle - \|f_\psi(\tilde{x}_t, t)\|_2^2$$

$$\theta \leftarrow \theta - \eta \nabla_\theta \hat{L}_\theta$$

until convergence

Return G_θ

Algorithm 2 Midpoint Generative Models (MGM)

```
# G: generator network,      (z) -> x_0
# f: field / critic network, (x, t) -> drift
# eps: strength for interpolant noise (scalar)

def mgm_loss_f(G, f, eps):
  # -------- update field network f --------
  z  = sample_noise(N)   # [N, d_z]
  x_1 = sample_data(N)   # [N, D]
  t  = rand(N, 1) * 0.5  # [N, 1], uniform in [0, 1/2]
  noise = randn(N, D)    # [N, D]
  b  = rand(N, 1) < 0.5  # [N, 1], Bernoulli(0.5)

  x_0 = stop_grad(G(z))  # [N, D]

  # symmetric midpoint observations
  x_t  = x_0 * (1.0 - t) + x_1 * t
  x_1mt = x_0 * t + x_1 * (1.0 - t)

  # add stochastisity to the interpolant
  sigma_t = sqrt(eps * t * (1.0 - t))
  x_t = x_t + noise * sigma_t
  x_1mt = x_1mt + noise * sigma_t

  # hide which branch was observed
  x_tilde = (1 - b) * x_t + b * x_1mt # [N, D]

  # regress field onto displacement x_1 - x_0
  target = stop_grad(x_1 - x_0) # [N, D]
  pred = f(x_tilde, t) # [N, D]
  L_f  = (pred - target).pow(2).sum(-1).mean()

  return L_f


def mgm_loss_G(G, f, eps):
   # -------- update generator G --------
   z  = sample_noise(N)
   x_1 = sample_data(N)
   t  = rand(N, 1) * 0.5
   noise = randn(N, D)
   b  = rand(N, 1) < 0.5

   x_0 = G(z)            # [N, D]
   x_t  = x_0 * (1.0 - t) + x_1 * t
   x_1mt = x_0 * t + x_1 * (1.0 - t)

   sigma_t = sqrt(eps * t * (1.0 - t))
   x_t = x_t + noise * sigma_t
   x_1mt = x_1mt + noise * sigma_t

   x_tilde = (1 - b) * x_t + b * x_1mt

   # variational objective for the generator
   drift = f(x_tilde, t) # [N, D]
   target = x_1 - x_0      # [N, D]
   L_G = (drift * (2*target - drift)).sum(-1).mean()

   return L_G
```