Exploring Half-Denoising and Annealed Langevin Dynamics for Generative Sampling

Faissal Izermine & Dhia Garbaya
Master MVA

January 7, 2026

Abstract

In this report, we explore a new strategy for sampling data from generative models. We combine two interesting ideas from recent research: "half-denoising" proposed by (4) and "annealed Langevin dynamics" proposed by (1). The goal is to see if we can get better samples by mixing these techniques. By employing the metrics established in (1), we aim to verify if this new algorithm yields better scores. Specifically, we investigate if half-denoising can help achieve a distribution closer to the true data as claimed in (4) , while combining it with annealing helps the model explore the data space better. We acknowledge that these methods are not the current State-of-the-Art (SOTA) compared to modern Diffusion Models, but revisiting these mathematical roots provides useful insights. We implemented these algorithms and tested them on image datasets (MNIST, CelebA, and CIFAR-10). Since generating images via Langevin dynamics is computationally expensive, we generated a limited number of samples (approx. 7,500–10,000 depending on the dataset). Despite the limited sample size, our experiments provide a promising signal. We evaluate our results using Inception Score (IS) and Fréchet Inception Distance (FID).

1 Introduction

Generative modeling is a field in Artificial Intelligence where we train a model to create new data, like images, that have a similar distribution to our real data. One popular way to do this is called "score-based modeling."

Instead of trying to learn the complex probability density directly, score-based models learn the "score function." Simply put, the score function is the gradient of the log-density. It acts like a vector field pointing in the direction where the data is most likely to be found.

However, learning this score function for real data is difficult. Real data often lives on a **manifold**, which means it only occupies a small slice of the total space. In the empty spaces, the score is undefined. To fix this, researchers usually add noise (like Gaussian noise) to the data. This smears the data out so the score is defined everywhere. This technique is called Denoising Score Matching (DSM).

The problem is that if we learn the score of noisy data, our generated samples will also be biased—they will remain slightly noisy. We need a way to remove this "noise bias." This report investigates a method called "half-denoising" to fix this. We combine it with "annealing" (starting with high noise and slowly lowering it) to see if we can get the best of both worlds: good exploration from annealing and low noise from half-denoising.

2 Related Work and Theoretical Background

In this section, we review the important methods used and the theory behind them.

2.1 Denoising Score Matching (DSM)

(2) proposed a simple and efficient way to learn the score function. The main idea is to avoid the difficult calculation of the normalizing constant Z. Instead, we take a clean image x, add random noise to create a noisy image $\tilde{x}$, and train a neural network to predict the clean image from the noisy one.

The objective function (loss) we minimize is the expected squared difference between our model's output score $s_\theta(\tilde{x})$ and the true gradient of the noising process:

$$J(\theta) = \mathbb{E}_{x,\tilde{x}} \left[\frac{1}{2} \left\| s_\theta(\tilde{x}) - \nabla_{\tilde{x}} \log q(\tilde{x}|x) \right\|^2 \right] \tag{1}$$

where $q(\tilde{x}|x)$ is the noise distribution. This trick works because minimizing this term is equivalent to matching the score of the noisy data density $q(\tilde{x})$ (3).

If we use Gaussian noise where $\tilde{x} = x + \sigma z$ (with standard deviation σ), the "true" gradient is very easy to calculate. It is just the vector pointing back to the clean image:

$$\nabla_{\tilde{x}} \log q(\tilde{x}|x) = \frac{x - \tilde{x}}{\sigma^2} \tag{2}$$

2.2 Annealed Langevin Dynamics

Once we have the score function, we need a way to generate samples. The standard method is **Langevin Dynamics**. It is an iterative process that starts from random noise x_0 and slowly moves towards high-density areas using the gradient (score):

$$x_{t+1} = x_t + \frac{\mu}{2} \nabla_x \log p(x_t) + \sqrt{\mu} z_t \tag{3}$$

where μ is the step size and $z_t \sim \mathcal{N}(0, I)$ is random noise.

However, (1) showed that this struggles in low-density regions where gradients are inaccurate. They proposed **Annealed Langevin Dynamics**, where we use a sequence of noise levels $\{\sigma_1, \sigma_2, \ldots, \sigma_L\}$ from large to small. We run the Langevin update for T steps at each noise level σ_i, using the step size $\alpha_i = \epsilon \cdot \sigma_i^2 / \sigma_L^2$ (where ϵ is a hyperparameter). This helps the model explore the data space globally before focusing on fine details.

2.3 Half-Denoising

While Annealed Langevin works well, (4) pointed out that the samples are slightly biased because we are using the score of *noisy* data. He proposed "Noise-Corrected Langevin" or **Half-Denoising**.

In standard Langevin, the process converges to the noisy distribution $p(\tilde{\mathbf{x}})$. However, since our data $\tilde{\mathbf{x}}$ already contains noise variance σ^2, Hyvärinen showed that we can adjust the injected noise to target the *clean* distribution $p(\mathbf{x})$.

Derivation of the Noise-Corrected Langevin Dynamics We define the relationship between the noisy observation and the clean variable as:

$$\tilde{\mathbf{x}}_t = \mathbf{x}_t + \sigma \mathbf{n}_t, \qquad \mathbf{n}_{t+1} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}).$$

The noise-corrected update for the underlying variable $\mathbf{x}$, using the noisy score $\mathbf{\Psi}_{\tilde{\mathbf{x}}}$, is defined as:

$$\mathbf{x}_{t+1} = \tilde{\mathbf{x}}_t + \mu \mathbf{\Psi}_{\tilde{\mathbf{x}}}(\tilde{\mathbf{x}}_t) + \sqrt{2\mu - \sigma^2}\, \boldsymbol{\nu}_t, \tag{4}$$

where $\boldsymbol{\nu}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is independent of $\mathbf{n}_{t+1}$. This requires $2\mu \geqslant \sigma^2$.

By substituting this into the definition of $\tilde{\mathbf{x}}_{t+1}$, the independent Gaussian noise terms combine ($\sqrt{2\mu - \sigma^2}$ and σ) into a single term $\sqrt{2\mu}\, \bar{\boldsymbol{\nu}}_t$:

$$\tilde{\mathbf{x}}_{t+1} = \tilde{\mathbf{x}}_t + \mu \mathbf{\Psi}_{\tilde{\mathbf{x}}}(\tilde{\mathbf{x}}_t) + \sqrt{2\mu}\, \bar{\boldsymbol{\nu}}_t. \tag{5}$$

This recovers standard Langevin dynamics for $\tilde{\mathbf{x}}$, proving the validity of the update.

At the limit $\mu = \sigma^2/2$, the added noise term $\sqrt{2\mu - \sigma^2}$ vanishes. This yields the "Half-Denoising" deterministic update:

$$\mathbf{x}_{t+1} = \tilde{\mathbf{x}}_t + \frac{\sigma^2}{2} \mathbf{\Psi}_{\tilde{\mathbf{x}}}(\tilde{\mathbf{x}}_t). \tag{6}$$

Notice the coefficient is $\sigma^2/2$. This method goes only "halfway" compared to full denoising. Theoretically, this removes the first-order bias, allowing convergence to the clean data distribution.

2.4 Smooth vs. Singular Data

(5) analyzed the theoretical limits of this "Half-Denoising" versus "Full-Denoising". They measured the Wasserstein distance W_2 between the generated distribution and the true distribution. Their findings depend on the "regularity" of the data:

- If the data density is **smooth** (regular enough), **Half-Denoising** is optimal and yields a smaller error.
- If the data is **singular** (like a sharp manifold or Dirac masses), **Full-Denoising** is better because half-denoising retains too much noise variance.

Since real images often lie on a manifold (singular) but have pixel noise (smooth), the best method is likely a hybrid, which motivates our experiments.

3 The First Algorithm: Annealed Langevin

This is the baseline algorithm from (1).

Algorithm 1 Annealed Langevin Dynamics

Require: $\{\sigma_i\}_{i=1}^{L}, \epsilon, T$
Initialize $\tilde{\mathbf{x}}_0$
for $i \leftarrow 1$ to L **do**
  $\alpha_i \leftarrow \epsilon \cdot \frac{\sigma_i^2}{\sigma_L^2}$
  for $t \leftarrow 1$ to T **do**
    Draw $\mathbf{z}_t \sim \mathcal{N}(0, I)$
    $\tilde{\mathbf{x}}_t \leftarrow \tilde{\mathbf{x}}_{t-1} + \alpha_i s_\theta(\tilde{\mathbf{x}}_{t-1}, \sigma_i) + \sqrt{2\alpha_i}\mathbf{z}_t$
  end for
  $\tilde{\mathbf{x}}_0 \leftarrow \tilde{\mathbf{x}}_T$
end for
return $\tilde{\mathbf{x}}_T$

4 Our Algorithm

We propose to combine the "Half-Denoising" strategy with "Annealed Langevin Dynamics" from (1).

4.1 Why combine them?

1. **Annealing**: Song and Ermon (1) showed that using a sequence of noise levels (from σ_{max} down to σ_{min}) helps the model mix better. It fixes the "sampling error" in terms of time;
2. **Half-Denoising**: Hyvärinen (4) showed this reduces the "variance" or bias at a specific noise level.

We hypothesize that by using annealing to get close to the right image, and then using a half-denoising-style update, we can get better quality samples.

4.2 The Method

We call our method "Annealed Half-Denoising."

1. We define a sequence of noise levels $\sigma_1 > \sigma_2 > \cdots > \sigma_L$.
2. We start with random noise.
3. For each noise level σ_i:
   - We run the dynamics loop.
   - Instead of the standard Langevin update, we use a modified update rule. We set the step size related to α_i (derived from σ_i^2) to cancel out the noise bias at that level, following the half-denoising principle.
4. We repeat this until we reach the lowest noise level.

4.3 Failed Attempt: Annealed Iterative Denoising

Below is our first attempt at a hybrid algorithm. This version applies the ϵ-based annealing strategy from Algorithm 1 directly to the iterative denoising logic.

Algorithm 2 Annealed Half-Denoising (Unstable)

Require: $\{\sigma_i\}_{i=1}^{L}, \epsilon, T$
Initialize $\mathbf{x}_0$
for $i \leftarrow 1$ to L **do**
  for $t \leftarrow 1$ to T **do**
    Draw $\mathbf{n}_t \sim \mathcal{N}(0, \mathbf{I})$
    $\tilde{\mathbf{x}}_{t-1} = \mathbf{x}_{t-1} + \sigma_i \mathbf{n}_t$
    $\mathbf{x}_t = \tilde{\mathbf{x}}_{t-1} + \frac{\sigma_i^2}{2} s_\theta(\tilde{\mathbf{x}}_{t-1}, \sigma_i)$
  end for
  $\mathbf{x}_0 \leftarrow \mathbf{x}_T$
end for
return $\mathbf{x}_T$

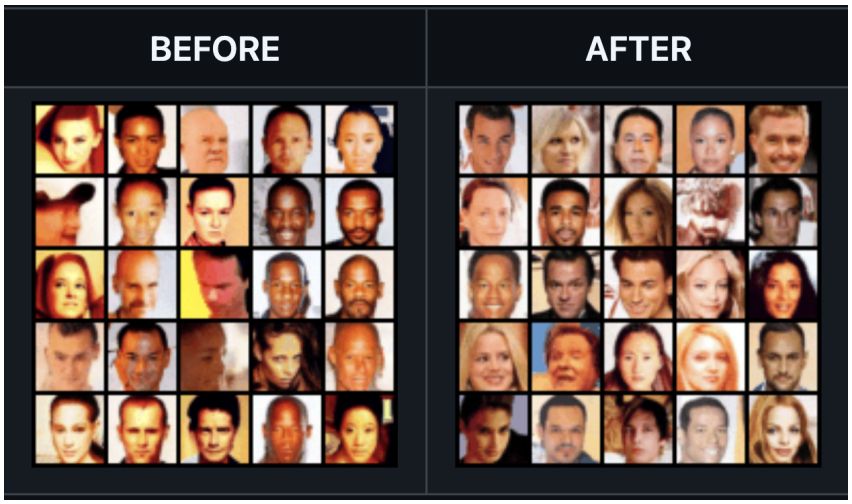
The issue with this algorithm is that the step sizes (σ_i) between noise levels are quite large The images change drastically at each step, preventing stable convergence. Therefore, we discard this approach.

4.4 Improved Algorithm: Annealed α-Half-Denoising

This version uses the deterministic update rule from half-denoising but scales it using the annealed step size α_i to ensure smooth transitions.

Algorithm 3 Annealed α-Half-Denoising (Hybrid)

Require: $\{\sigma_i\}_{i=1}^{L}, \epsilon, T$
Initialize $\mathbf{x}_0$
for $i \leftarrow 1$ to L **do**
  $\alpha_i \leftarrow \epsilon \cdot \frac{\sigma_i^2}{\sigma_L^2}$
  for $t \leftarrow 1$ to T **do**
    Draw $\mathbf{n}_t \sim \mathcal{N}(0, \mathbf{I})$
    $\tilde{\mathbf{x}}_{t-1} = \mathbf{x}_{t-1} + \sqrt{2\alpha_i}\mathbf{n}_t$
    $\mathbf{x}_t = \tilde{\mathbf{x}}_{t-1} + \alpha_i s_\theta(\tilde{\mathbf{x}}_{t-1}, \sigma_i)$
  end for
  $\mathbf{x}_0 \leftarrow \mathbf{x}_T$
end for
return $\mathbf{x}_T$



Algorithm 2 (before), Algorithm 3 (after)

5 Evaluation Metrics

To measure if our images are good, we cannot just look at them. We use computable scores commonly found in the literature:

5.1 Inception Score (IS)

The Inception Score evaluates the quality of generated images based on two key aspects:

- **Image clarity:** Each generated image should strongly correspond to a specific object class, reflected by low entropy in the conditional label distribution $p(y \mid x)$.
- **Image diversity:** The full set of generated images should cover many different classes, indicated by high entropy in the marginal distribution $p(y)$.

To compute this score, a pre-trained Inception network is used to obtain class probability predictions. The IS is defined as:

$$\text{IS} = \exp\left(\mathbb{E}_x\left[\text{KL}\left(p(y \mid x) \,\|\, p(y)\right)\right]\right)$$

Higher Inception Scores indicate better image quality and diversity.

5.2 Fréchet Inception Distance (FID)

FID is generally considered better than IS. It compares the statistics of generated images to real images. It treats the features of the images (from the Inception network) as if they are Gaussian distributions. It calculates the distance between the Gaussian of real images (μ_r, Σ_r) and generated images (μ_g, Σ_g).

Optimal Transport Interpretation: FID is actually the **Wasserstein-2 distance** between these two Gaussian distributions. In Optimal Transport theory, this measures the "cost" to move the mass of one distribution to the other.

$$FID = ||\mu_r - \mu_g||^2 + \text{Tr}(\Sigma_r + \Sigma_g - 2(\Sigma_r\Sigma_g)^{1/2}) \tag{7}$$

A lower FID means the generated images are statistically very close to the real images.

6 Experiments and Results

We ran our experiments on three standard datasets: MNIST, CelebA, and CIFAR-10. Note: Generating samples with Langevin dynamics is very slow because it requires hundreds of steps for every image. Due to time constraints, we generated approximately $k \approx 7{,}500$ to 10,000 samples depending on the dataset. While small compared to some large-scale studies, it provides a sufficient signal for comparison.

We compared three methods:

1. **Baseline:** Standard Annealed Langevin Dynamics (1).
2. **Raw Half-Denoising:** (4)'s method combined with annealing.
3. **Our Hybrid:** α-Annealed Half-Denoising.

6.1 Results Tables

Method	**Noise Levels**	**Inception Score (IS) ↑**	**FID ↓**
Baseline (Ordinary Annealed)	Multiple	1.9238 ± 0.0178	54.4459
Our Hybrid (Annealed Half)	Multiple	**1.9252** ± 0.0164	**46.8944**

Table 1: Comparison of Inception Scores and FID for 10k sampled MNIST.

Method	**Noise Levels**	**Inception Score (IS) ↑**	**FID ↓**
Baseline (Ordinary Annealed)	Multiple	**2.3222** ± 0.0494	16.0085
Our Hybrid (Annealed Half)	Multiple	2.2537 ± 0.0724	**14.6682**

Table 2: Comparison of Inception Scores and FID for 7500 sampled CelebA.

Method	Noise Levels	Inception Score (IS) ↑	FID ↓
Baseline (Ordinary Annealed)	Multiple	8.7243 ± 0.2801	32.8998
Our Hybrid (Annealed Half)	Multiple	**8.8674** ± 0.1931	**27.6049**

Table 3: Comparison of Inception Scores and FID for 10k sampled CIFAR-10.

6.2 Interpretation

The results across all three datasets show a consistent positive trend for our proposed method. On CIFAR-10, the **FID score improved significantly to 27.60**, compared to the baseline's 32.89. Similarly, we achieved lower FID scores on both CelebA (14.66 vs 16.00) and MNIST (46.89 vs 54.44). Inception Scores (IS) also remained comparable or slightly better than the baseline across all experiments.

We interpret these results as confirmation of our hypothesis: while the annealing strategy successfully helps the model explore the global structure of the data (fixing the "mixing" problem), the **Half-Denoising** update rule provides a better local correction for the noise bias. By not over-correcting the noise (as full denoising might do), the hybrid method produces samples that are statistically closer to the real distribution.

7 Conclusion

In this report, we went back to the mathematical roots of score-based generative models. We explored if combining "Half-Denoising" (bias correction) with "Annealing" (better exploration) could improve sampling.

Our experiments, conducted on 7,500 to 10,000 samples, showed a positive signal. Specifically, the Hybrid "Annealed Half-Denoising" method consistently improved the FID score across MNIST, CelebA, and CIFAR-10. This suggests that the noise correction proposed by (4) is indeed effective when combined with the robust exploration of annealing.

While these methods are not the absolute SOTA compared to modern diffusion models, understanding the fundamental trade-off between noise smoothing (5) and bias correction (4) remains highly relevant for designing more efficient sampling algorithms.

References

[1] Song, Y., & Ermon, S. (2019). *Generative Modeling by Estimating Gradients of the Data Distribution*. Advances in Neural Information Processing Systems (NeurIPS), 32.

[2] Vincent, P. (2011). *A Connection Between Score Matching and Denoising Autoencoders*. Neural Computation, 23(7), 1661-1674.

[3] Hyvärinen, A. (2005). *Estimation of Non-Normalized Statistical Models by Score Matching*. Journal of Machine Learning Research, 6(4), 695–709.

[4] Hyvärinen, A. (2025). *A noise-corrected Langevin algorithm and sampling by half-denoising*. arXiv preprint arXiv:2410.05837v3 [cs.LG].

[5] Beyler, E., & Bach, F. (2025). *Optimal Denoising in Score-Based Generative Models: The Role of Data Regularity*. arXiv preprint arXiv:2503.12966v2 [cs.LG].