Diffusion Generative Flow Samplers: Improving Learning Signals Through Partial Trajectory Optimization

Dinghuai Zhang*, Ricky T. Q. Chen, Cheng-Hao Liu, Aaron Courville & Yoshua Bengio

Thomas Mousseau

October 20, 2025

Overview

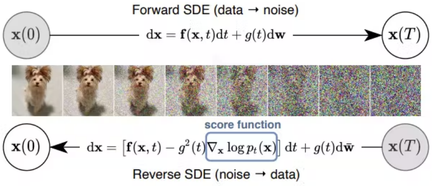
Generative Modeling

Task

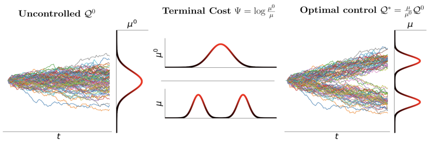
Sample from a complex (high-dimensional and multimodal) distribution D

D can be given under the form of:

- A dataset of samples $\{x_i\}_{i=1}^N \sim D$ (e.g., images, text, audio)



- An unnormalized density $\mu(x)$ where D has density $\pi(x) \propto \mu(x)$ (e.g., energy-based models, physics/chemistry)



Sampling from Unnormalized Densities

Goal. Sample from a D-dimensional target with unnormalized density $\mu(x)$ where $\mathbb{R}^D \to \mathbb{R}^+$.

$$\pi(x) = \frac{\mu(x)}{Z}, \qquad Z = \int_{\mathbb{R}^D} \mu(x)\, dx \text{ (unknown)}.$$

We assume we can evaluate $\mu(x)$, but we have no samples from π and do not know Z.

Context. We seek a *sampler* (similar to MCMC/VI) that produces calibrated samples and, ideally, estimates of $\log Z$, *without* any dataset from π.

Chemistry (small-molecule conformers). Different 3D conformations have a formation energy from force-field terms (bonds, angles, dihedrals, nonbonded); lower energy $\Rightarrow$ higher Boltzmann probability. A well-calibrated sampler is needed to draw conformers in proportion to these probabilities, which is important in binding-pose ranking/free-energy estimation, Boltzmann-weighted property prediction (e.g., NMR shifts), and generating diverse realistic 3D conformers for screening.

Diffusion Generative Flow Samplers

Idea. We will *reframe sampling* from an unnormalized target $\pi(x) \propto \mu(x)$ as a *stochastic optimal control (SOC)* problem: learn a control that steers a simple reference diffusion so its *terminal marginal* matches π.

Why this helps.

- Gives a *path-space* training objective/metric: a KL on trajectories $\mathrm{KL}(Q \,\|\, P)$ where P is the reference paths reweighted by $\mu(x_T)$.
- The *partition function Z cancels* inside this objective, so we can train using only μ (and optionally $\nabla \log \mu$).
- Lets us optimize *without samples from π* and still measure closeness to the true normalized endpoint.

Caveat (sets up DGFS). This path-KL places supervision *only at the terminal time* $\Rightarrow$ poor *credit assignment* and high-variance gradients.

DGFS fix (preview). Inject *intermediate* learning signals via a GFlowNet-inspired *learned flow* and *subtrajectory balance*, enabling partial-trajectory training and more stable learning.

Steps 1–2: Forward & Reference in discrete time

Controlled forward transition (learned drift).

$$P_F(x_{n+1} \mid x_n) = \mathcal{N}\big(x_{n+1};\ x_n + h\, f(x_n, n),\ h\sigma^2 I\big)$$

Controlled path law.

$$Q(x_{0:N}) = p_0^{\text{ref}}(x_0) \prod_{n=0}^{N-1} P_F(x_{n+1} \mid x_n)$$

Uncontrolled (reference) transition (zero drift).

$$P_F^{\text{ref}}(x_{n+1} \mid x_n) = \mathcal{N}\big(x_{n+1};\ x_n,\ h\sigma^2 I\big)$$

Reference path law and marginals.

$$Q^{\text{ref}}(x_{0:N}) = p_0^{\text{ref}}(x_0) \prod_{n=0}^{N-1} P_F^{\text{ref}}(x_{n+1} \mid x_n), \qquad p_n^{\text{ref}}(x) \text{ is closed form.}$$

Goal. Learn f so that the terminal marginal $Q(x_N)$ matches $\pi(x) = \mu(x)/Z$ (no data, Z unknown).

Step 3: Path target & KL $\Rightarrow$ SOC objective

Target path measure via terminal reweighting.

$$P(x_{0:N}) \;\propto\; Q^{\mathrm{ref}}(x_{0:N})\,\frac{\mu(x_N)}{p_N^{\mathrm{ref}}(x_N)} \qquad \Longrightarrow \qquad P(x_N) \propto \mu(x_N).$$

KL decomposition.

$$\mathrm{KL}(Q\|P) = \mathbb{E}_Q\left[\log \frac{Q}{Q^{\mathrm{ref}}}\right] + \mathbb{E}_Q\left[\log \frac{p_N^{\mathrm{ref}}(x_N)}{\pi(x_N)}\right].$$

Running control cost (Gaussian mean-shift) Girsanov theorem.

$$\mathbb{E}_Q\left[\log \frac{Q}{Q^{\mathrm{ref}}}\right] = \mathbb{E}_Q \sum_{n=0}^{N-1} \frac{h}{2\sigma^2}\,\|f(x_n,n)\|^2.$$

Terminal potential from the target Girsanov theorem.

$$\mathbb{E}_Q\left[\log \frac{p_N^{\mathrm{ref}}(x_N)}{\pi(x_N)}\right] = \mathbb{E}_Q\big[\log p_N^{\mathrm{ref}}(x_N) - \log \mu(x_N)\big] \;+\; \log Z.$$

SOC objective

SOC objective (discrete-time).

$$\boxed{\min_f \ \mathbb{E}_Q\Big[\sum_{n=0}^{N-1} \tfrac{h}{2\sigma^2}\|f(x_n,n)\|^2 \ + \ \log p_N^{\text{ref}}(x_N) - \log \mu(x_N)\Big]}$$

we will be using this discrete-time objective as our loss function to optimize the drift f. Thus we can model the drift with a neural network f_θ and optimize the parameters θ backpropagating through time (BPTT) while avoiding to use the stochastic-adjoint method necessary for a neural SDE.

Credit Assignment Problem in Path Space

Conclusion