RESTRAINED DIFFUSION BRIDGES: TARGETED DENOISER TRAINING AND REFINEMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Diffusion models are often inaccurate in critical, data-sparse regions of the target distribution. Sampling from restrained distributions, biased distributions used in classical enhanced sampling to concentrate data near chosen configurations, can control where training data are collected but alters the target distribution. We establish an exact cancellation that occurs when forward noising from restrained data is conditioned on its value at an appropriately chosen intermediate time. This allows a pretrained model fine-tuned on restrained data to achieve high accuracy in any chosen region of the target distribution, regardless of its probability, using standard denoising or flow matching objectives without importance weights. When no target data are available, Gibbs sampling enables inference from a model trained exclusively on restrained data.

1 INTRODUCTION

A fundamental challenge in diffusion-based generative modeling is poor accuracy in regions of state space that have low probability under the (possibly unnormalized) target measure π and are poorly represented in training data. These regions, such as transition states in molecular simulation, rare posterior configurations, or low-frequency image modes, are often of critical importance in downstream analysis. A potential remedy is to reweight the training distribution toward rare regions or collect more data there. However, simple reweighting or oversampling causes the model to learn the wrong target measure. Correcting for this with importance weights, when they are available, does not resolve the problem. By design, those weights are small in low probability regions.

Classical umbrella sampling uses harmonic biases to restrain simulations near selected configurations or collective-variable values (Torrie & Valleau, 1977; Kästner, 2011). The resulting densities $\pi(x)\exp(-\|x-z\|^2/2s^2)$ suppress distant configurations and keep a chain near z, facilitating sampling even when z is highly atypical under π. As for reweighting and oversampling, a diffusion model trained directly on restrained data will learn the restrained density rather than π.

In this article we show that conditioning the forward noising process on restrained data exactly cancels the biasing effect of those data, allowing a model trained on restrained data, using standard denoising or flow matching objectives and no other modifications, to recover the unconditional score of π with no importance weights or correction terms (Theorem 1). This cancellation exploits the observation that the restrained density at scale s coincides exactly with the posterior distribution $\pi_{0|t^*}(x_0 \mid x_{t^*})$ inside a diffusion model when the noise level t^* is matched to s (Lemma 1).

These results motivate a targeted training algorithm that requires no additional samples from π and allows precise control over where training data is concentrated without introducing any bias. It can be used as a fine-tuning algorithm (Section 4) that takes a pretrained diffusion model that degrades near a rare configuration and refines it on $[0, t^*]$ and locally in sample space using restrained samples near that region. When $\nabla \log \pi$ is available, we sample the restrained distributions using Langevin sampling. We also propose an alternative dataset resampling strategy when only samples from π are available (Appendix C). The fine-tuned model produces independent samples via standard inference at no extra cost. When no reasonable pretrained model is available, our targeted training algorithm can be used to train a model valid on $[0, t^*]$ only. The model on $[0, t^*]$ can then be used to generate samples from π using a learned Gibbs sampler (Section 5.2). We validate our approach on the Müller-Brown potential, where analytic ground truth is available, and on two molecular benchmarks, alanine

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

dipeptide in explicit solvent and Chignolin in implicit solvent, that test the approach in realistic simulation settings (Section 6).

2 Related work

Our goal is unconditional sampling from π. Diffusion posterior sampling (Chung et al., 2023) and classifier or classifier-free guidance (Dhariwal & Nichol, 2021; Ho & Salimans, 2022) condition or steer inference without refining the marginal score in selected regions. Methods that learn samplers from an unnormalized target include path-space variational and stochastic-control objectives (Zhang & Chen, 2022; Vargas et al., 2023; Berner et al., 2024; Richter & Berner, 2024; Domingo-Enrich et al., 2024; Sendera et al., 2024; Havens et al., 2025) and iterative energy matching (Akhound-Sadegh et al., 2024). These methods use π or $\nabla \log \pi$ in the loss. Ours uses the target only to generate restrained data and an ordinary denoising loss. It therefore also applies to implicit marginals, and its dataset-resampling variant requires samples but no energy evaluations (Appendix C).

Restricted-Gaussian conditionals underlie proximal Gibbs sampling (Lee et al., 2021) and related diffusion-assisted MCMC methods (Kim & Ye, 2023; Chen et al., 2024). Generative Gibbs methods (Wang et al., 2026c;a;b) similarly exploit the freedom to learn a conditional $p(x \mid y)$ under a convenient marginal over y and compose it with a different y-distribution at inference. Very recent work by Chen et al. (2026a) implements a Gibbs transition through learned denoising and a Metropolis correction to remove score and discretization bias. Diffusion bridge methods construct endpoint-conditioned generative dynamics (De Bortoli et al., 2021; Zhou et al., 2024). Here, posterior-matched restrained samples are bridged to fixed endpoints and the ordinary denoising target recovers the unconditional score. Endpoint frequencies can therefore emphasize rare regions. This targeted use of restrained components is related to stratification, which can improve inverse-temperature scaling of classical sampling for multimodal targets (Dinner et al., 2020).

3 Setup

We use x_t for forward noised states. We reserve bare π for the target density. Subscripted π denotes the finite-dimensional laws induced by the forward process initialized with $x_0 \sim \pi$. For example, π_t, $\pi_{0,t}$, $\pi_{0|t}$, and $\pi_{t|0,t^*}$ denote its marginal, joint, posterior, and bridge densities, respectively.

For definiteness, we consider any Gaussian forward process with marginals

$$x_t = \alpha_t\, x_0 + \sigma_t\, \varepsilon, \qquad \varepsilon \sim \mathcal{N}(0, I), \quad x_0 \sim \pi, \tag{1}$$

where $\alpha_t > 0$, $\sigma_t > 0$, and α_t/σ_t is strictly decreasing, with $\alpha_0 = 1$, $\sigma_0 \approx 0$, and $\alpha_T/\sigma_T \approx 0$ so that $x_T \approx \sigma_T \varepsilon$ is noise-dominated. Our results generalize beyond Gaussian and continuous space settings with appropriate re-definition of the restrained densities. We write $\alpha_{t|s} = \alpha_t/\alpha_s$ for the transition ratio and $\sigma^2_{t|s} = \sigma^2_t - \alpha^2_{t|s}\sigma^2_s$ for the conditional variance. The latter is non-negative when α_t/σ_t is decreasing. The forward process (1) induces the joint and marginal densities

$$\pi_{0,t}(x_0, x_t) = \pi(x_0)\mathcal{N}(x_t; \alpha_t x_0, \sigma^2_t I), \qquad \pi_t(x_t) = \int \pi_{0,t}(x_0, x_t)\, \mathrm{d}x_0,$$

whose score $\nabla_{x_t} \log \pi_t(x_t)$ drives the reverse dynamics.

The marginal (1) is realized by the Itô stochastic differential equation (SDE)

$$\mathrm{d}x_t = \frac{\dot\alpha_t}{\alpha_t}\, x_t\, \mathrm{d}t + g_t\, \mathrm{d}W_t, \qquad g^2_t \;=\; \dot\sigma^2_t - \frac{2\dot\alpha_t\, \sigma^2_t}{\alpha_t}, \tag{2}$$

where W_t is standard Brownian motion, overdots denote time derivatives ($\dot\sigma^2_t \equiv d(\sigma^2_t)/dt$, $\dot\alpha_t \equiv d\alpha_t/dt$), and $g^2_t \geq 0$ follows from α_t/σ_t decreasing. The corresponding reverse-time SDE is

$$\mathrm{d}x_t = \Big[\frac{\dot\alpha_t}{\alpha_t}\, x_t - g^2_t\, \nabla_{x_t} \log \pi_t(x_t)\Big]\mathrm{d}t + g_t\, \mathrm{d}\bar W_t, \tag{3}$$

where $\bar W_t$ is a reverse-time Brownian motion (Song et al., 2021). Alternatively, the probability flow ordinary differential equation (ODE)

$$\dot x_t = f^\pi_t(x_t), \qquad f^\pi_t(x_t) \;=\; \frac{\dot\alpha_t}{\alpha_t}\, x_t - \frac{g^2_t}{2}\, \nabla_{x_t} \log \pi_t(x_t), \tag{4}$$

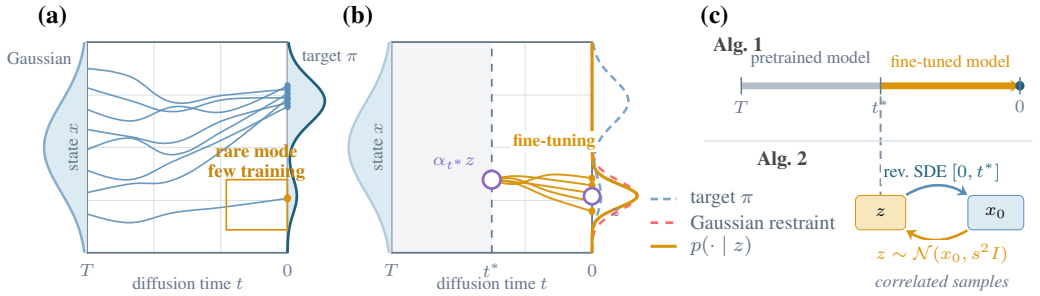

Figure 1: Conceptual overview of Restrained Diffusion Bridges. **(a)** Standard training provides many trajectories to the dominant mode of π but few to the rare mode. **(b)** Multiplying π (blue dashed) by the Gaussian restraint (red dashed) gives the restrained target $p(\cdot \mid z)$ (orange). Restrained endpoints and bridges pinned at $x_{t^*} = \alpha_{t^*} z$ focus fine-tuning on the rare region while recovering the score of π. **(c)** Algorithm 1 uses the pretrained model from T to t^* and the fine-tuned model from t^* to 0. Algorithm 2 alternates the reverse bridge with $z \sim \mathcal{N}(x_0, s^2 I)$ using only the fine-tuned model

shares the same marginals as (3) (Song et al., 2021) and generates samples from π deterministically. We refer to both (3) and (4) as the *reverse dynamics*. The choice between stochastic and deterministic integration is independent of training. The variance preserving schedule ($\alpha_t^2 + \sigma_t^2 = 1$) and the flow-matching schedule ($\alpha_t = 1 - t$, $\sigma_t = t$) (Lipman et al., 2023) are special cases. With the latter schedule, the ODE (4) recovers standard flow-matching inference. A network $\psi_\theta(x_t, t)$ is trained to approximate a chosen parameterization of either the *score* $\nabla_{x_t} \log \pi_t(x_t)$ or the *velocity field* $f_t^\pi(x_t)$ and then, at inference time, used to drive the corresponding reverse dynamics (3) or (4) starting from initial conditions sampled from $\mathcal{N}(0, \sigma_T^2 I)$. The pretrained model is accurate where training data is dense. Our goal is to refine it in low-density regions for $t \le t^*$.

We choose a data-space *probe point* $z \in \mathbb{R}^d$ and a bridge time $t^* \in (0, T)$, which together determine the restraint width $s = \sigma_{t^*}/\alpha_{t^*}$. The restrained density centered at z is

$$p(x_0 \mid z) \;\propto\; \pi(x_0)\, \exp\left(-\frac{\|x_0 - z\|^2}{2s^2}\right). \tag{5}$$

Gaussian forward-noise posteriors are common in diffusion-assisted sampling (Kim & Ye, 2023; Chen et al., 2024). As the following lemma shows, restrained samples at scale s are exact draws from the diffusion posterior at $x_{t^*} = \alpha_{t^*} z$, the condition under which bridge score matching recovers the unconditional score of π (Theorem 1).

Lemma 1 (Posterior identification)**.** *With $s = \sigma_{t^*}/\alpha_{t^*}$, the restrained density equals the diffusion posterior exactly: $p(x_0 \mid z) = \pi_{0|t^*}(x_0 \mid \alpha_{t^*} z)$.*

Proof. Apply Bayes' theorem to the Gaussian forward kernel, then substitute $s = \sigma_{t^*}/\alpha_{t^*}$. □

Forward noising $x_t = \alpha_t x_0 + \sigma_t \varepsilon$ from $x_0 \sim p(\cdot \mid z)$ produces a restrained marginal rather than π_t, so ordinary denoising score matching (Vincent, 2011; Ho et al., 2020) would learn the wrong score. More generally, at a fixed time t, the population denoising regressor depends on the conditional law $\pi_{0|t}(x_0 \mid x_t)$, not on the marginal distribution from which x_t is sampled. We could therefore choose any distribution for x_t and then draw x_0 from its matching posterior. In the Gaussian case this posterior is a restrained distribution, but implementing this construction directly would require a separate restrained simulation for every selected x_t and t.

A bridge amortizes these conditional simulations across time. Fix $t^* > t$ and an endpoint y, draw $x_0 \sim \pi_{0|t^*}(x_0 \mid y)$, and then draw $x_t \sim \pi_{t|0,t^*}(x_t \mid x_0, y)$. By the Markov property of the forward process, $\pi_{0|t,t^*}(x_0 \mid x_t, y) = \pi_{0|t}(x_0 \mid x_t)$. Thus ordinary denoising regression has the unconditional population optimum for every fixed endpoint y. The marginal distribution of x_{t^*} may therefore be chosen arbitrarily, allowing endpoints in rare regions to be oversampled. Lemma 1 identifies the required endpoint posterior for $y = \alpha_{t^*} z$ with the restrained ensemble $p(\cdot \mid z)$.

Given a starting point x_0 drawn from $p(\cdot \mid z)$ and the bridge endpoint $y = \alpha_{t^*} z$, the forward process conditioned on both values is a Gaussian bridge:

$$x_t \mid (x_0,\ x_{t^*} = \alpha_{t^*} z) \ \sim\ \mathcal{N}\Big(\mu_t(x_0, z),\ \sigma^2_{\mathrm{br}}(t)\, I\Big), \quad t \in [0, t^*], \tag{6}$$

with mean and bridge variance

$$\mu_t(x_0, z) = \alpha_t\, x_0 + \frac{\alpha_{t^*|t}\, \alpha_{t^*}\, \sigma_t^2}{\sigma^2_{t^*}}\big(z - x_0\big), \qquad \sigma^2_{\mathrm{br}}(t) = \frac{\sigma_t^2\, \sigma^2_{t^*|t}}{\sigma^2_{t^*}} \ \le\ \sigma_t^2. \tag{7}$$

Because $\sigma^2_{\mathrm{br}}(t) < \sigma_t^2$ for every $t \in (0, t^*)$, bridge samples are more concentrated than ordinary forward samples around the conditional bridge tube connecting x_0 to $\alpha_{t^*} z$. They therefore provide dense training coverage of the targeted region throughout the low-noise interval.

4 Restrained bridge-based training

Algorithm 1, introduced in this section requires samples from the probe-point restrained density $p(\cdot \mid z)$ for some choice of $t^* \in (0, T)$ to train a diffusion model on $[0, t^*]$. Algorithm 1 additionally supports training using samples from the CV joint density (8) as described below. To generate training data we assume access to the true score $\nabla_{x_0} \log \pi(x_0)$ (e.g. the negative gradient of a potential energy), so the restrained score is $\nabla_{x_0} \log p(x_0 \mid z) = \nabla_{x_0} \log \pi(x_0) - (x_0 - z)/s^2$. Overdamped Langevin dynamics with this score confines the chain to a neighborhood around z, regardless of how atypical z is under π, at the same per-step cost as Langevin on π itself (one call to $\nabla_{x_0} \log \pi$). An alternative training procedure that leverages existing samples from π instead of access to the true score is provided in Appendix C.

The restraint width $s = \sigma_{t^*}/\alpha_{t^*}$ controls the mixing time of the restrained Langevin sampler. Fast dynamics like bond vibrations and angle bending in molecular dynamics (MD) simulations equilibrate rapidly under any confining potential and impose no constraint on s. The operative upper bound comes from the slowest degrees of freedom. If $p(\cdot \mid z)$ spans a free-energy barrier, restrained Langevin must cross it and mixes no faster than the unrestrained chain. Numerical stability of fast modes and free-energy barrier geometry are thus independent constraints, and the ratio of these length scales can span many orders of magnitude, giving wide latitude in the choice of s. In the Gibbs sampling context s plays the same role as the umbrella force constant, but is determined by the noise schedule rather than set by hand.

Theorem 1 (Training correctness)**.** *Let $z \in \mathbb{R}^d$, let $x_0 \sim p(\cdot \mid z)$, and let x_t be drawn from the bridge (6) with endpoint $y = \alpha_{t^*} z$. Then for any π, any $t^* \in (0, T)$, and every $t \in (0, t^*)$:*

(i) The L^2-optimal regressor of the score target $v = -(x_t - \alpha_t x_0)/\sigma_t^2$ against x_t equals $\nabla_{x_t} \log \pi_t(x_t)$.

(ii) The L^2-optimal regressor of the velocity target $u_t(x_t \mid x_0) = \dot\alpha_t\, x_0 + \dfrac{\dot\sigma_t}{\sigma_t}(x_t - \alpha_t\, x_0)$ against x_t equals $f_t^\pi(x_t)$ (4).

Proof sketch. Lemma 1 identifies $x_0 \sim \pi_{0|t^*}(\cdot \mid y)$, so the construction is a posterior-initialized likelihood bridge. The general cancellation identity, Proposition 1, gives part (i), and part (ii) follows by an affine change of parameterization. The full proof is in Appendix A. □

Theorem 1 extends immediately to a low-dimensional collective variable (CV) $\mathcal{C} : \mathbb{R}^d \to \mathbb{R}^k$ constrained to a target value $y \in \mathbb{R}^k$ (e.g. a dihedral angle or end-to-end distance). Define the CV-restrained distribution to be the joint conditional law

$$\pi_{0,t^*|C_{t^*}}(x_0, x_{t^*} \mid y) := \mathcal{L}(x_0, x_{t^*} \mid \mathcal{C}(x_{t^*}) = y)\,. \tag{8}$$

For any draw (x_0, x_{t^*}) from (8), the conditional $x_0 \mid x_{t^*}$ is exactly the probe-point restrained density (5) with implicit probe $z = x_{t^*}/\alpha_{t^*}$. The Gaussian bridge (6) from x_0 to x_{t^*} is therefore a standard probe-point bridge, and Theorem 1 applies directly for any $\mathcal{C}$ and y. Appendix B gives the exact linear-CV construction. For nonlinear CVs the sampler must target (8).

The model need not represent every degree of freedom of the system. It is sufficient that the restraint on the full system depend only on the modeled variables (e.g. protein backbone angles). Let the full configuration $X \sim \Pi$ reduce to the modeled variable $x_0 = R(X)$, with target marginal $\pi = R_{\#}\Pi$. The CV conditioning in (8) multiplies π by a factor $b_y(x_0)$ depending only on x_0, with $p(x_0 \mid y) \propto \pi(x_0) b_y(x_0)$ the x_0-marginal of (8). Biasing the full system by $b_y(R(X))$ therefore produces this restrained marginal from training data generated in the full space.

Several strategies are available to select the probe points z_j. If an initial set of training samples from π is available and contains one or more samples x_{tar} in the targeted region, then we can choose $z_j = x_{\mathrm{tar}} + s\,\eta_j$ with $\eta_j \sim \mathcal{N}(0, I)$. The points x_{tar} can be selected deterministically, or according to any chosen reweighting of the initial training samples from π. Note that each x_{tar} can correspond to many z_j. The added noise is not strictly necessary, but is often important for training an accurate denoiser. If few or no samples from π are available, several exploration strategies can be used to generate probe points x_{tar} covering the target region. Perhaps the simplest is to run Langevin dynamics on π at reduced inverse temperature $\beta < 1$ (scaling the score by β samples π^{β}). Other options include adiabatic-bias MD (Marchi & Ballone, 1999), which can steer configurations to prescribed regions of sample space, as well as temperature-accelerated MD (Maragliano & Vanden-Eijnden, 2006), and metadynamics (Laio & Parrinello, 2002).

Algorithm 1 **Restrained Bridge Training**

Input: pretrained ψ_θ, bridge time t^* (giving α_{t^*}, σ_{t^*}), gradient steps G, batch size B, learning rate ρ
Input: sample pools $\{\mathcal{D}_j\}_{j=1}^{M}$ providing pairs (x_0, z_j): either $x_0 \sim p(\cdot \mid z_j)$ with explicit probe z_j [probe-point], or (x_0, x_{t^*}) from (8) with $z_j = x_{t^*}/\alpha_{t^*}$ [CV; Appendix B]
Output: fine-tuned model ψ_θ

for gradient step $g = 1$ **to** G **do**
  For each $b = 1, \ldots, B$: draw $j^{(b)} \sim \mathrm{Uniform}\{1, \ldots, M\}$ and $(x_0^{(b)}, z_{j^{(b)}}) \sim \mathcal{D}_{j^{(b)}}$ independently ▷ *one probe per item*
  Sample bridge times $t^{(b)} \sim \mathrm{Uniform}(0, t^*)$ independently
  for $b = 1$ **to** B **do**
    Compute bridge statistics for $(x_0^{(b)}, z_{j^{(b)}}, t^{(b)})$:
      $\mu^{(b)} \leftarrow \alpha_{t^{(b)}} x_0^{(b)} + \dfrac{\alpha_{t^*|t^{(b)}}\, \alpha_{t^*}\, \sigma_{t^{(b)}}^2}{\sigma_{t^*}^2}\left(z_{j^{(b)}} - x_0^{(b)}\right)$ ▷ *Eq. (7)*
      $\sigma_{\mathrm{br}}^{(b)} \leftarrow \sigma_{t^{(b)}}\, \sigma_{t^*|t^{(b)}} \,/\, \sigma_{t^*}$ ▷ *Eq. (7)*
    Sample $\eta^{(b)} \sim \mathcal{N}(0, I)$; set $x_t^{(b)} \leftarrow \mu^{(b)} + \sigma_{\mathrm{br}}^{(b)}\, \eta^{(b)}$
    Compute denoising target $\tau^{(b)}$: ▷ *Thm. 1; see below for both options*
      score: $\tau^{(b)} \leftarrow -(x_t^{(b)} - \alpha_{t^{(b)}} x_0^{(b)})/\sigma_{t^{(b)}}^2$
      velocity: $\tau^{(b)} \leftarrow \dot{\alpha}_{t^{(b)}} x_0^{(b)} + (\dot{\sigma}_{t^{(b)}}/\sigma_{t^{(b)}})(x_t^{(b)} - \alpha_{t^{(b)}} x_0^{(b)})$
  end for
  Update: $\theta \leftarrow \theta - \rho\, \nabla_\theta \dfrac{1}{B} \sum_b \left\| \psi_\theta(x_t^{(b)}, t^{(b)}) - \tau^{(b)} \right\|^2$
end for
return ψ_θ

Algorithm 1 handles multiple probe points by drawing one independent $(z_{j^{(b)}}, x_0^{(b)})$ pair per batch item, giving an unbiased estimator of $\mathbb{E}_{z\sim q}\, \mathbb{E}_{x_0 \sim p(\cdot|z)}[\ell_{\mathrm{bridge}}]$ where $\ell_{\mathrm{bridge}} = \|\psi_\theta(x_t, t) - \tau\|^2$, with no within-batch correlation from a shared bridge endpoint. The CV-restrained case (Section 3, Appendix B) fits within the same loop. Probe-point and CV pools may be mixed across $\{j\}$ with any chosen frequencies. When $\nabla \log \pi$ is unavailable but samples from π exist, dataset resampling (Appendix C) provides a third pool type that fits the same loop without any modification.

5 INFERENCE

5.1 INFERENCE FROM A FINE-TUNED MODEL

Theorem 1 guarantees that the procedure in Algorithm 1 produces the correct reverse dynamics' driving field ($\nabla_{x_t} \log \pi_t$ or f_t^π) for $t \in [0, t^*]$. Combined with the pretrained model on $[t^*, T]$, the resulting model on $[0, T]$ therefore produces samples correctly distributed according to π (up to training error). The π_{t^*} marginal smooths π by a Gaussian kernel of variance $\sigma_{t^*}^2$, so the pretrained denoiser need only capture the smoothed landscape. The fine-tuned model resolves local detail for $t \le t^*$. Errors arise only if the pretrained denoiser is inaccurate at noise levels above t^*, causing the reverse dynamics to deliver x_{t^*} far from π_{t^*}.

5.2 INFERENCE WITHOUT $[t^*, T]$

When no pretrained denoiser is available for $t > t^*$, the bridge-trained model supports a learned Gibbs sampler producing correlated samples. It implements Gibbs sampling on the joint density

$$p(x_0, z) \;=\; \pi(x_0)\,\mathcal{N}\big(z;\, x_0,\, s^2 I\big), \tag{9}$$

whose x_0-marginal is π by construction, and whose conditionals are $p(x_0 \,|\, z) = \pi_{0|t^*}(x_0 \,|\, \alpha_{t^*} z)$ by Lemma 1 and $p(z \,|\, x_0) = \mathcal{N}(z; x_0, s^2 I)$. Gibbs sampling on (9) alternates between two steps:

1. **Sample** $x_0 \sim \pi_{0|t^*}(\cdot \,|\, \alpha_{t^*} z)$: set $x_{t^*} \leftarrow \alpha_{t^*} z$ exactly and run the stochastic reverse SDE (3) from t^* to 0 with ψ_θ.
2. **Resample** $z \sim \mathcal{N}(x_0, s^2 I)$: a single Gaussian draw.

The z-update ($z \leftarrow x_0 + s\eta$, $\eta \sim \mathcal{N}(0, I)$) samples exactly from the forward kernel $x_{t^*} \,|\, x_0 \sim \mathcal{N}(\alpha_{t^*} x_0, \sigma_{t^*}^2 I)$, so the resulting x_{t^*} has the correct marginal π_{t^*}. Stochastic reverse diffusion from x_{t^*} with the correct score (Theorem 1) then recovers $\pi_{0|t^*}(x_0 \,|\, x_{t^*})$, and marginalizing over x_{t^*} returns π. Pseudocode, proximal interpretation, and a Metropolized variant are provided in Appendix D.

6 EXPERIMENTS

We evaluate targeted denoiser refinement on the two-dimensional Müller–Brown system, as well as two molecular generation problems: explicit-solvent alanine dipeptide (ALDP) demonstrates Cartesian sampling without an explicit solute-marginal energy function, while implicit-solvent Chignolin tests high-dimensional rare-state fine-tuning in the latent space of a 93-heavy-atom model. We assess both rare-state and global accuracy.

6.1 MÜLLER–BROWN POTENTIAL

The two-dimensional Müller–Brown potential (Müller & Brown, 1979) is a standard illustrative benchmark whose energy U has three metastable basins separated by substantial barriers. The Boltzmann density $\pi(x) \propto \exp(-\beta U(x))$ can be evaluated exactly on a grid. For this example we measure sampling quality against ground truth computed by quadrature. The inverse-temperature β acts as a difficulty parameter. Raising it sharpens the gradients of the target and raises the effective barriers. The relationship between inverse-temperature and our method is explored in Appendix E.

We train a diffusion model on restrained Langevin bridge endpoints alone, without pretraining, over diffusion times $t \in [0, t^*]$ for $t^* = 0.02$. 10,000 probe points z are obtained from an unrestrained Langevin trajectory at $\beta_{\text{probe}} = 0.5$, recorded every 0.25 time units. Each probe initializes a 0.3 time unit trajectory of overdamped Langevin dynamics, whose final configuration yields a single restrained sample. Integrations were performed at a time step of $\mathrm{d}t = 10^{-3}$. The baseline standard diffusion model is trained over the full $t \in [0, 1]$ range on unrestrained Langevin data. Both models use the same architecture (a three-layer MLP with 128 hidden units, trained for 10,000 iterations) and total Langevin cost (which yields 22,000 training points for the standard diffusion model using the same sampling interval).

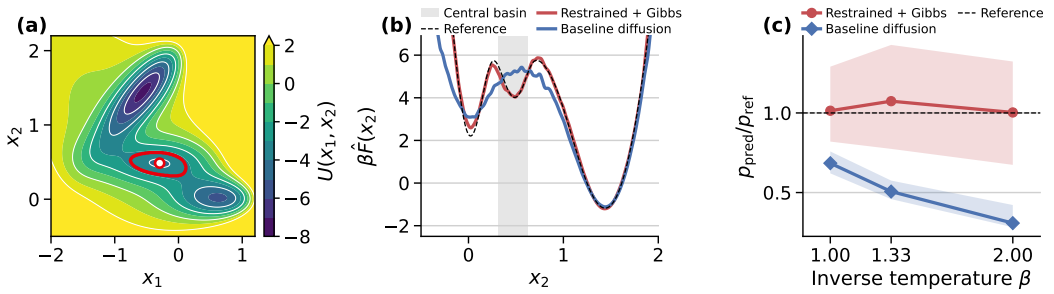

Figure 2: (a) The Müller–Brown potential with the central basin outlined in red. (b) Free-energy profile along x_2, $\beta\hat{F}(x_2) = -\ln p(x_2)$, at $\beta = 2.00$ for a representative sampling run of each method against the exact profile; the shaded band marks the extent of the central basin in x_2. The restrained sampler recovers the central minimum, whereas the baseline resolves no minimum there. (c) Central-basin occupancy relative to its exact value, $p_{\mathrm{pred}}/p_{\mathrm{ref}}$, against inverse temperature β for the restrained sampler at $t^* = 0.02$ and for the budget-matched baseline. Median and interquartile range are calculated over twenty independently seeded runs.

For the Gibbs sampler on $[0, t^*]$, we run $1{,}000$ independent chains, all initialized at the deepest minimum, for $5{,}000$ sweeps each. Each sweep noises the state forward to t^* and integrates the reverse process back to $t = 0$ in 100 steps. We treat sweeps before $1{,}000$ as burn-in and retain every tenth configuration from sweep $1{,}000$ through sweep $5{,}000$, inclusive. This yields 401 configurations per chain and $401{,}000$ pooled samples. The baseline is sampled directly from noise using the same 100 reverse steps, yielding $400{,}000$ samples.

Figure 2(a) shows that the central basin is a high-energy region carrying little probability mass, a deficit that grows exponentially with β. A diffusion model trained directly on equilibrium samples therefore sees few training points there. As shown in Figure 2(b), along x_2 the baseline washes the basin out entirely, whereas the restrained Gibbs sampler resolves the minimum accurately. Figure 2(c) shows that the failure of the baseline model becomes more acute as temperature decreases, with its interquartile range excluding exact agreement at every β. In contrast, the restrained sampler stays close to the exact occupancy throughout the temperature range, suggesting that an analogue of umbrella sampling's favorable low-temperature efficiency may apply (Dinner et al., 2020).

6.2 Explicit-solvent ALDP

Our ALDP experiment involves a 22-atom solute and 628 TIP3P waters (Jorgensen et al., 1983). Solvent variables appear only in the MD simulations used to generate training data. The diffusion model represents the marginal distribution of Kabsch-aligned (Kabsch, 1976) solute coordinates. The Ramachandran angles (ϕ, ψ) (Ramachandran et al., 1963) summarize the conformational distribution and define five metastable free-energy basins, including the rarest, R_5 (Figure 3(a)). We assess global fidelity using the Jensen–Shannon (JS) distance between normalized two-dimensional (ϕ, ψ) histograms, defined as the square root of the JS divergence (Lin, 1991). Rare-state accuracy is assessed by comparing the generated population $p_{\mathrm{gen}}(R_5)$ with the reference population.

We compare two ways to improve rare-basin sampling against a baseline diffusion model. The baseline model is trained on five 300 K equilibrium MD trajectories of $5.05\,\mu$s each, including a 50 ns warm-up per trajectory, for a total MD budget of $25.25\,\mu$s. It captures the global distribution but underestimates the population of R_5. The two approaches use the same model architecture and are compared at matched numbers of training updates.

Setting I: refinement without additional MD. We use the equilibrium dataset to construct synthetic restraint pairs and fine-tune the base model from its $20{,}000$-update checkpoint for another $30{,}000$ updates. On a 50×50 Ramachandran grid, $10{,}000$ probes are allocated in proportion to n_{ij}^{γ}, where n_{ij} is the equilibrium count in cell (i, j) and $\gamma = 0.5$. For each probe, we select 30 synthetic restrained samples without further MD or energy evaluations (Appendix C). The restraints act on the Cartesian coordinates of the solute heavy atoms (Appendix B); the Ramachandran angles are used

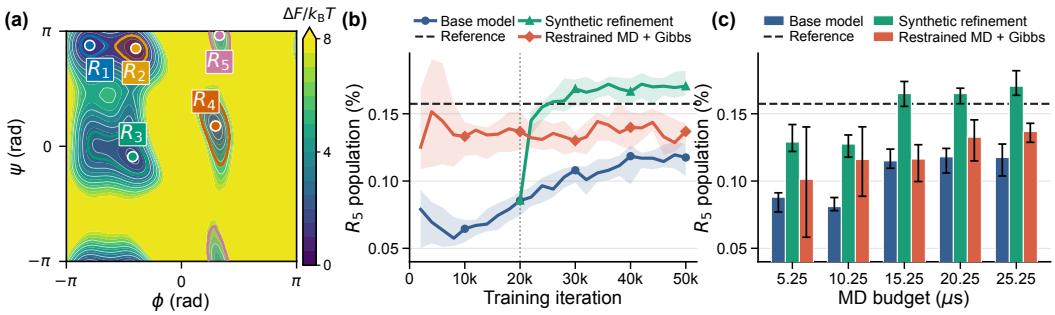

Figure 3: Explicit-solvent ALDP results. (a) Five low-free-energy basins in Ramachandran space. (b) Generated R_5 population versus training updates at a fixed $25.25\,\mu$s MD budget; synthetic refinement branches at 20k updates. (c) Generated R_5 population versus matched MD budget at 50k updates. Curves and bars show medians over 20 runs, and bands and error bars show interquartile ranges. Dashed lines indicate the reference value.

only to allocate probes and evaluate the generated samples. One-shot generation switches from the base 20k-update checkpoint to the fine-tuned denoiser at $t^* = 0.40$.

Setting II: generation at fixed MD cost. When starting from an energy function and no pre-trained model, we instead spend the same $25.25\,\mu$s MD budget on a $250\,$ns trajectory at 500 K to generate $5{,}000$ probes and $5{,}000 \times 5\,$ns of restrained MD at 300 K. Although total simulation time is matched, the short restrained trajectories allow much greater parallelism and can substantially reduce wall-clock time. The resulting pairs train a denoiser on $[0, t^*]$, with $t^* = 0.40$, from scratch. Equilibrium samples are then obtained using a Gibbs chain.

At the full $25.25\,\mu$s MD budget, Figure 3(b) compares two variants of targeted denoiser refinement. The synthetic-pair variant branches from the base model at 20k updates and rapidly brings the R_5 population close to the reference. The restrained-MD-pair variant provides a more accurate estimate than the baseline model even at the earliest training checkpoints. To assess sensitivity to simulation cost, we then fix training at $50{,}000$ updates and vary the matched MD budget from 5.25 to $25.25\,\mu$s (Figure 3(c)). At each budget, we shorten all five equilibrium trajectories used by the base model and synthetic-pair refinement, and adjust the number of restrained-MD probes from $1{,}000$ to $5{,}000$. Synthetic refinement improves the R_5 estimate across all budgets. Restrained-MD refinement improves it relative to the matched-budget baseline model and continues to improve with increasing MD budget while the baseline model accuracy appears to saturate. The 2D JS distances remain similar across methods and budgets. Their trends and further details are reported in Appendix F.

6.3 Implicit-solvent Chignolin

Next we study CLN025 (YYDPETGTWY; 93 heavy atoms, 279 Cartesian coordinates) (Honda et al., 2008) using a $22\,\mu$s trajectory at $340\,$K, sampled every $10\,$ps. We discard the first $2\,\mu$s as warm-up, and evaluate every model against the remaining $20\,\mu$s reference ensemble. The distances $\mathrm{NO1} = d(\mathrm{Asp3\,N}, \mathrm{Gly7\,O})$ and $\mathrm{NO2} = d(\mathrm{Asp3\,N}, \mathrm{Thr8\,O})$ distinguish the native π-turn (short NO2) from the misfolded α-turn (short NO1, longer NO2) (Maruyama et al., 2020). The misfolded region, $\mathrm{NO1} < 3.7\,$Å and $5.0 < \mathrm{NO2} < 8.0\,$Å, has reference population 0.3747% (Figure 4(a)).

For this system, we did not obtain a satisfactory Cartesian baseline diffusion model within a reasonable training budget. We therefore train one autoencoder to define a 128-dimensional latent space. That representation is shared across independently seeded diffusion runs at the same MD budget (Appendix G). Using a learned encoder as a restraint CV would require repeated encoder evaluation and differentiation during MD, so we consider only refinement without additional MD (Setting I). As for ALDP, we allocate $10{,}000$ probes to occupied $(\mathrm{NO1}, \mathrm{NO2})$ cells in proportion to $n_{ij}^{0.5}$ and construct 30 synthetic posterior endpoints per probe. The base model is trained for $160{,}000$ updates. Synthetic refinement branches from its 80k checkpoint and performs 80k fine-tune updates at t^*. One-shot generation switches from the base-80k model to the fine-tuned denoiser at t^*.

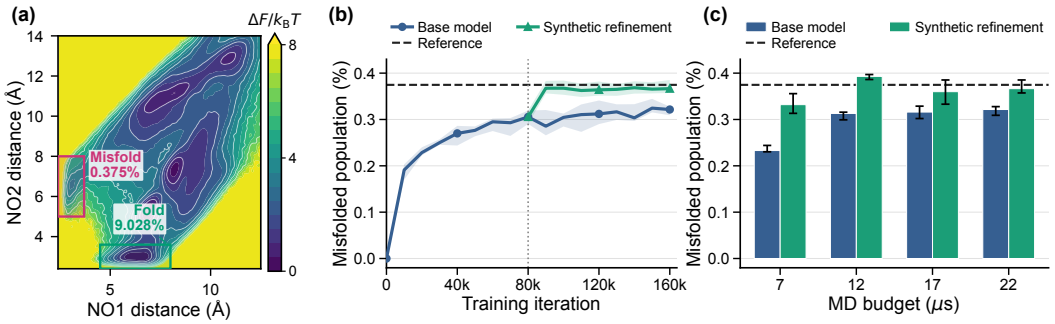

Figure 4: Implicit-solvent CLN025 results. (a) Free energy in the $(\mathrm{NO1}, \mathrm{NO2})$ plane for the $20\,\mu$s reference ensemble, with folded and misfolded regions outlined. (b) Generated misfolded population versus training updates at a fixed $22\,\mu$s MD budget. Synthetic refinement branches at 80k updates. (c) Generated misfolded population versus total MD budget at 160k total updates. Synthetic refinement uses $t^* = 0.30$. Curves and bars show medians over 10 runs, and bands and error bars show interquartile ranges. Dashed lines indicate the reference value.

At the full $22\,\mu$s MD budget, the median generated misfolded population rises from 0.3055% at the 80k branch point to 0.3675% after only 10k refinement updates and remains close to the reference thereafter (Figure 4(b)). At 160k total updates, synthetic refinement improves the median from 0.3215% for the base model to 0.3665%, reducing the absolute relative error from 14.2% to 2.2%. Across the 7, 12, 17, and $22\,\mu$s MD budgets, the corresponding synthetic errors are 11.4%, 4.9%, 3.9%, and 2.2%, compared with 37.8%, 16.5%, 15.7%, and 14.2% for the base model (Figure 4(c)). Again the baseline model accuracy appears to saturate as the MD budget increases. At the full MD budget, the median 2D NO JS distance increases slightly from 0.10875 to 0.11673. Additional JS-distance and bridge-time results are reported in Appendix G.

7 Conclusion

Leveraging the fact that the restrained density at scale s is the exact diffusion posterior at the noise level t^* matched to s (Lemma 1), we showed that conditioning the forward noising process on restrained data exactly cancels the biasing effect of those data. As a result, a model trained on restrained distributions, using standard denoising or flow matching objectives, recovers the unconditional score of π with no importance weights or correction terms (Theorem 1).

Algorithm 1 uses restrained Langevin samples to fine-tune a pretrained model on $[0, t^*]$ near chosen probe points. The result produces independent samples from π via standard reverse dynamics at no extra cost. A likelihood-free variant replaces Langevin sampling with importance-weighted resampling from an existing dataset when $\nabla \log \pi$ is unavailable (Appendix C). When no pretrained model is available, the learned Gibbs sampler can use only the bridge-trained model on $[0, t^*]$ to generate correlated samples from π.

Our approach cannot repair errors due to an inadequate choice of network architecture. We are also not yet able to provide more than heuristic guidance on the choice of t^*, though we find that our results are relatively insensitive to that choice (Appendix G.6). For the Gibbs scheme there is an important tradeoff between higher sampling efficiency of the restrained distributions for lower t^* and faster mixing at inference for larger t^* (Appendix E).

The most immediate application is to molecular simulation, where large pretrained structure models degrade at transition states and rare configurations, exactly where biological function is determined. Algorithm 1 addresses this failure without global retraining.

Several directions remain open: whether an analog of the umbrella sampling exponential data efficiency result in Dinner et al. (2020) holds, extensions of the framework to discrete diffusion, and broader applications of the likelihood-free resampling approach to high-dimensional generative settings such as image generation.

AI USE STATEMENT

Generative AI tools were used to identify relevant literature, create and revise scientific figures, and write and revise code used in the execution of the research. Generative AI tools were also used to draft and revise portions of the manuscript and to improve its clarity, structure, grammar, and style. The diffusion models and synthetic posterior-pair constructions studied in this paper are components of the research methodology and do not constitute the use of external generative AI tools to generate the research datasets. The authors take responsibility for the final content of this paper, including text, claims, and artifacts produced with the assistance of generative AI.

REPRODUCIBILITY STATEMENT

The assumptions and derivations underlying restrained bridge training are stated in Sections 3 and 4, with the complete proof of the main correctness result provided in Appendix A. Algorithm 1 specifies the bridge construction and training objective, while the collective-variable construction, likelihood-free dataset resampling procedure, and learned Gibbs sampler are detailed in Appendices B, C, and D, respectively. For the Müller–Brown, alanine-dipeptide, and Chignolin experiments, Section 6 and Appendices E, F, and G report the target systems, simulation and preprocessing procedures, restraint and probe construction, model architectures, diffusion schedules, optimization and sampling settings, matched data or simulation budgets, random seeds, generated sample counts, evaluation metrics, and repeated-run aggregation procedures. The corresponding appendices also detail the fixed reference distributions and finite-sample baselines used for evaluation. These details are intended to support independent implementation and reproduction of the theoretical and numerical results.

REFERENCES

Tara Akhound-Sadegh, Jarrid Rector-Brooks, Joey Bose, Sarthak Mittal, Pablo Lemos, Cheng-Hao Liu, Marcin Sendera, Siamak Ravanbakhsh, Gauthier Gidel, Yoshua Bengio, Nikolay Malkin, and Alexander Tong. Iterated denoising energy matching for sampling from boltzmann densities. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, pp. 760–786, 2024.

Julius Berner, Lorenz Richter, and Karen Ullrich. An optimal control perspective on diffusion-based generative modeling. *Transactions on Machine Learning Research*, 2024.

Han Chen, Sifan Liu, and Jun Yang. Markov chain monte carlo with diffusion paths, 2026a. URL https://arxiv.org/abs/2607.11631.

Wenlin Chen, Mingtian Zhang, Brooks Paige, José Miguel Hernández-Lobato, and David Barber. Diffusive Gibbs sampling. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pp. 7731–7747, 2024.

Yifan Chen, Xiaoou Cheng, Jonathan Niles-Weed, and Jonathan Weare. Convergence of unadjusted langevin in high dimensions: Delocalization of bias. *Communications on Pure and Applied Mathematics*, 79(6):1467–1491, 2026b. doi: 10.1002/cpa.70032. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/cpa.70032.

Hyungjin Chung, Jeongsol Kim, Michael T. McCann, Marc L. Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *International Conference on Learning Representations (ICLR)*, 2023.

Tom Darden, Darrin York, and Lee Pedersen. Particle mesh ewald: An nlog(n) method for ewald sums in large systems. *The Journal of Chemical Physics*, 98(12):10089–10092, 06 1993. ISSN 0021-9606. doi: 10.1063/1.464397. URL https://doi.org/10.1063/1.464397.

Valentin De Bortoli, James Thornton, Jeremy Heng, and Arnaud Doucet. Diffusion schrödinger bridge with applications to score-based generative modeling. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pp. 17695–17709, 2021.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat GANs on image synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pp. 8780–8794, 2021.

Aaron R. Dinner, Erik H. Thiede, Brian Van Koten, and Jonathan Weare. Stratification as a general variance reduction method for Markov chain Monte Carlo. *SIAM/ASA Journal on Uncertainty Quantification*, 8(3):1139–1188, 2020.

Carles Domingo-Enrich, Jiequn Han, Brandon Amos, Joan Bruna, and Ricky T. Q. Chen. Stochastic optimal control matching. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 2024.

Peter Eastman, Jason Swails, John D. Chodera, Robert T. McGibbon, Yutong Zhao, Kyle A. Beauchamp, Lee-Ping Wang, Andrew C. Simmonett, Matthew P. Harrigan, Chaya D. Stern, Rafal P. Wiewiora, Bernard R. Brooks, and Vijay S. Pande. Openmm 7: Rapid development of high performance algorithms for molecular dynamics. *PLOS Computational Biology*, 13(7): 1–17, 07 2017. doi: 10.1371/journal.pcbi.1005659. URL https://doi.org/10.1371/journal.pcbi.1005659.

Aaron J. Havens, Benjamin Kurt Miller, Bing Yan, Carles Domingo-Enrich, Anuroop Sriram, Daniel S. Levine, Brandon M. Wood, Bin Hu, Brandon Amos, Brian Karrer, Xiang Fu, Guan-Horng Liu, and Ricky T. Q. Chen. Adjoint sampling: Highly scalable diffusion samplers via adjoint matching. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, pp. 22204–22237, 2025.

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus), 2023. URL https://arxiv.org/abs/1606.08415.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv:2207.12598*, 2022.

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

Shinya Honda, Toshihiko Akiba, Yusuke Kato, Yoshito Sawada, Masakazu Sekijima, Miyuki Ishimura, Ayako Ooishi, Hideki Watanabe, Takayuki Odahara, and Kazuaki Harata. Crystal structure of a ten-amino acid protein. *Journal of the American Chemical Society*, 130:15327–31, 11 2008. doi: 10.1021/ja8030533.

William Jorgensen, Jayaraman Chandrasekhar, Jeffry Madura, Roger Impey, and Michael Klein. Comparison of simple potential functions for simulating liquid water. *J. Chem. Phys.*, 79:926–935, 07 1983. doi: 10.1063/1.445869.

W. Kabsch. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A*, 32(5):922–923, 1976. doi: 10.1107/S0567739476001873. URL https://onlinelibrary.wiley.com/doi/abs/10.1107/S0567739476001873.

Johannes Kästner. Umbrella sampling. *WIREs Computational Molecular Science*, 1(6):932–942, 2011. doi: 10.1002/wcms.66.

Beomsu Kim and Jong Chul Ye. Denoising MCMC for accelerating diffusion-based generative models. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, volume 202 of *Proceedings of Machine Learning Research*, pp. 16955–16977, 2023.

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

Alessandro Laio and Michele Parrinello. Escaping free-energy minima. *Proceedings of the National Academy of Sciences*, 99(20):12562–12566, 2002.

Yin Tat Lee, Ruoqi Shen, and Kevin Tian. Structured logconcave sampling with a restricted gaussian oracle. In *Proceedings of the 34th Conference on Learning Theory (COLT)*, volume 134 of *Proceedings of Machine Learning Research*, pp. 2993–3050, 2021.

J. Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151, 1991. doi: 10.1109/18.61115.

Kresten Lindorff-Larsen, Stefano Piana, Kim Palmo, Paul Maragakis, John L. Klepeis, Ron O. Dror, and David E. Shaw. Improved side-chain torsion potentials for the amber ff99sb protein force field. *Proteins: Structure, Function, and Bioinformatics*, 78(8):1950–1958, 2010. doi: 10.1002/prot.22711. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/prot.22711.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations (ICLR)*, 2023.

James A. Maier, Carmenza Martinez, Koushik Kasavajhala, Lauren Wickstrom, Kevin E. Hauser, and Carlos Simmerling. ff14sb: Improving the accuracy of protein side chain and backbone parameters from ff99sb. *Journal of Chemical Theory and Computation*, 11(8):3696–3713, 08 2015. ISSN 1549-9618. doi: 10.1021/acs.jctc.5b00255. URL https://doi.org/10.1021/acs.jctc.5b00255.

Luca Maragliano and Eric Vanden-Eijnden. A temperature accelerated method for sampling free energy and determining reaction pathways in rare events simulations. *Chemical Physics Letters*, 426(1-3):168–175, 2006. doi: 10.1016/j.cplett.2006.05.062.

Massimo Marchi and Pietro Ballone. Adiabatic bias molecular dynamics: A method to navigate the conformational space of complex molecular systems. *The Journal of Chemical Physics*, 110(8):3697–3702, 1999. doi: 10.1063/1.478259.

Yutaka Maruyama, Shunpei Koroku, Misaki Imai, Koh Takeuchi, and Ayori Mitsutake. Mutation-induced change in chignolin stability from pi-turn to alpha-turn. *RSC Advances*, 10(38):22797–22808, 06 2020. ISSN 2046-2069. doi: 10.1039/d0ra01148g. URL https://doi.org/10.1039/d0ra01148g.

Klaus Müller and Leo D. Brown. Location of saddle points and minimum energy paths by a constrained simplex optimization procedure. *Theoretica Chimica Acta*, 53(1):75–93, 1979.

Hai Nguyen, Daniel R. Roe, and Carlos Simmerling. Improved generalized born solvent model parameters for protein simulations. *Journal of Chemical Theory and Computation*, 9(4):2020–2034, 04 2013. ISSN 1549-9618. doi: 10.1021/ct3010485. URL https://doi.org/10.1021/ct3010485.

Jerome P. Nilmeier, Gavin E. Crooks, David D. L. Minh, and John D. Chodera. Nonequilibrium candidate Monte Carlo is an efficient tool for equilibrium simulation. *Proceedings of the National Academy of Sciences*, 108(45):E1009–E1018, 2011.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, pp. 3942–3951, 2018. doi: 10.1609/aaai.v32i1.11671.

G. N. Ramachandran, C. Ramakrishnan, and V. Sasisekharan. Stereochemistry of polypeptide chain configurations. *Journal of Molecular Biology*, 7:95–99, 1963. doi: 10.1016/S0022-2836(63)80023-6.

Lorenz Richter and Julius Berner. Improved sampling via learned diffusions. In *International Conference on Learning Representations (ICLR)*, 2024.

Marcin Sendera, Minsu Kim, Sarthak Mittal, Pablo Lemos, Luca Scimeca, Jarrid Rector-Brooks, Alexandre Adam, Yoshua Bengio, and Nikolay Malkin. Improved off-policy training of diffusion samplers. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 2024.

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 (ICLR)*, 2021.

Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems*, volume 30, pp. 1195–1204, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/68053af2923e00204c3ca7c6a3150cf7-Abstract.html.

Glenn M. Torrie and John P. Valleau. Nonphysical sampling distributions in monte carlo free-energy estimation: Umbrella sampling. *Journal of Computational Physics*, 23(2):187–199, 1977. doi: 10.1016/0021-9991(77)90121-8.

Francisco Vargas, Will Grathwohl, and Arnaud Doucet. Denoising diffusion samplers. In *International Conference on Learning Representations (ICLR)*, 2023.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, pp. 5998–6008, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7):1661–1674, 2011.

Weizhou Wang, Jonathan Weare, and Aaron R. Dinner. Composing diffusion priors with explicit physical context via generative gibbs sampling. *arXiv preprint arXiv:2605.10642*, 2026a. doi: 10.48550/arXiv.2605.10642.

Weizhou Wang, Jonathan Weare, and Aaron R. Dinner. Nuclear quantum effects as a denoising problem. *arXiv preprint arXiv:2607.19680*, 2026b. doi: 10.48550/arXiv.2607.19680.

Weizhou Wang, Xuanxi Zhang, Jonathan Weare, and Aaron R. Dinner. Quantum statistics from classical simulations via generative gibbs sampling. *arXiv preprint arXiv:2601.20228*, 2026c. doi: 10.48550/arXiv.2601.20228.

Qinsheng Zhang and Yongxin Chen. Path integral sampler: A stochastic control approach for sampling. In *International Conference on Learning Representations (ICLR)*, 2022.

Zhijun Zhang, Xinzijian Liu, Kangyu Yan, Mark E. Tuckerman, and Jian Liu. Unified efficient thermostat scheme for the canonical ensemble with holonomic or isokinetic constraints via molecular dynamics. *The Journal of Physical Chemistry A*, 123(28):6056–6079, 07 2019. ISSN 1089-5639. doi: 10.1021/acs.jpca.9b02771. URL https://doi.org/10.1021/acs.jpca.9b02771.

Linqi Zhou, Aaron Lou, Samar Khanna, and Stefano Ermon. Denoising diffusion bridge models. In *International Conference on Learning Representations (ICLR)*, 2024.

A Likelihood–bridge cancellation and proof of Theorem 1

The main theorem is a Gaussian instance of the following classical Doob-transform identity.

Proposition 1 (Posterior-initialized likelihood bridge)**.** *Let a Markov process x_t have initial density π, transition density $k_{0,t}(x \mid x_0)$, and a nonnegative endpoint-observation likelihood ℓ_y. For $t \le t^*$ define*

$$h_t(x) = \mathbb{E}[\ell_y(x_{t^*}) \mid x_t = x].$$

Initialize x_0 from the posterior proportional to $\pi(x_0)h_0(x_0)$ and sample the corresponding likelihood-conditioned bridge. Then the population regressor of the ordinary denoising target $\nabla_x \log k_{0,t}(x \mid x_0)$, conditional on $x_t = x$, is $\nabla_x \log \pi_t(x)$.

Proof. Writing $Z_y = \int \pi(x_0)h_0(x_0)\,dx_0$, the posterior-initialized bridge joint density at times 0 and t is

$$q_y(x_0, x) = Z_y^{-1}\pi(x_0)k_{0,t}(x \mid x_0)h_t(x).$$

The factor $h_t(x)$ is constant in the conditional law of x_0 given $x_t = x$. Hence

$$\mathbb{E}_{q_y}[\nabla_x \log k_{0,t}(x \mid x_0) \mid x_t = x] = \nabla_x \log \int \pi(x_0) k_{0,t}(x \mid x_0)\, dx_0 = \nabla_x \log \pi_t(x).$$

Thus the likelihood, equivalently the Doob-transform factor, cancels from the ordinary regression. □

Proof of Theorem 1. For the Gaussian process, observing $x_{t^*} = y$ gives $h_t(x) = k_{t,t^*}(y \mid x)$. Lemma 1 identifies the posterior initial law $\pi(x_0)h_0(x_0)/Z_y$ with $p(x_0 \mid z)$, and the corresponding likelihood-conditioned process is the bridge (6). Moreover,

$$\nabla_{x_t} \log k_{0,t}(x_t \mid x_0) = -\frac{x_t - \alpha_t x_0}{\sigma_t^2} = v.$$

Proposition 1 proves part (i). For part (ii), $g_t^2 = \dot{\sigma}_t^2 - 2(\dot{\alpha}_t/\alpha_t)\sigma_t^2$ gives

$$u_t(x_t \,|\, x_0) = \frac{\dot{\alpha}_t}{\alpha_t} x_t - \frac{g_t^2}{2} v.$$

Taking conditional expectations and applying part (i) yields $\mathbb{E}[u_t \mid x_t] = f_t^{\pi}(x_t)$. □

The exact endpoint observation gives the Gaussian restrained construction, while the regular conditional observation $\mathcal{C}(x_{t^*}) = y$ gives (8). Thus both the probe-point and exact CV constructions are special cases of Proposition 1, which is an organizing identity rather than a novelty claim about the Doob transform.

B Sampling the CV-restrained joint density

This appendix provides procedures for generating pairs $(x_0^{(i)}, x_{t^*}^{(i)})$ from the conditional law (8), which are passed to Algorithm 1 with implicit probe $z_j = x_{t^*}^{(i)}/\alpha_{t^*}$.

Linear CVs. Let $\mathcal{C}(x) = Ax$ with $A \in \mathbb{R}^{k \times d}$ full row-rank. Marginalizing x_{t^*} in (8) gives the x_0-marginal

$$\pi_{0|C_{t^*}}(x_0 \,|\, y) \;\propto\; \pi(x_0)\, \mathcal{N}\big(y;\, \alpha_{t^*} A x_0,\, \sigma_{t^*}^2 A A^\top\big), \tag{10}$$

a harmonic restraint on Ax_0 with target y/α_{t^*} and stiffness $\alpha_{t^*}^2/\sigma_{t^*}^2$. Samples $x_0^{(i)}$ are drawn by restrained Langevin dynamics in $\mathbb{R}^d$. Given $x_0^{(i)}$, the x_{t^*}-conditional is Gaussian and is drawn analytically:

$$x_{t^*}^{(i)} \sim \mathcal{N}\Big(\alpha_{t^*} x_0^{(i)} + A^\top (AA^\top)^{-1}\big(y - \alpha_{t^*} A x_0^{(i)}\big),\; \sigma_{t^*}^2 \big(I - A^\top (AA^\top)^{-1} A\big)\Big). \tag{11}$$

Since $I - A^\top(AA^\top)^{-1}A$ is the orthogonal projector onto $\ker A$, the noise lies in $\ker A$ and $Ax_{t^*}^{(i)} = y$ holds exactly. Pairs $(x_0^{(i)}, x_{t^*}^{(i)})$ are therefore generated independently with no MCMC over x_{t^*}.

Non-linear CVs. When $\mathcal{C}$ is nonlinear, the sampler must target the regular conditional law (8). Replacing the constraint by a finite stiff harmonic penalty gives an approximation to that law that can be sampled using standard algorithms.

Setting $\mathcal{C}(x) = x$ (i.e. $A = I$, $k = d$) in (11), the covariance $\sigma_{t^*}^2(I - I^\top(II^\top)^{-1}I)$ vanishes, so the conditional distribution degenerates to the point mass $x_{t^*} = y = \alpha_{t^*} z$. The two-step procedure then reduces exactly to the probe-point construction of Section 4: sample $x_0 \sim p(\cdot \mid z)$ by restrained Langevin, then set $x_{t^*} = \alpha_{t^*} z$ deterministically.

C Dataset resampling for likelihood-free bridge-based model refinement

Algorithm 1 requires $\nabla \log \pi$ to draw restrained samples by Langevin dynamics. When samples $\mathcal{D} = \{x_0^{(i)}\}_{i=1}^m$ from π are available but gradients are not—e.g. image distributions or black-box

simulators—restrained samples can instead be obtained by importance-weighted resampling from $\mathcal{D}$.

Choose a probe by perturbing a dataset point: $z = \tilde{x} + s\eta$, $\tilde{x} \in \mathcal{D}$, $\eta \sim \mathcal{N}(0, I)$. Assign each dataset point the weight

$$w^{(i)}(z) \;=\; \exp\Big(-\frac{\big\|x_0^{(i)} - z\big\|^2}{2s^2}\Big) \tag{12}$$

and draw x_0^* proportional to the normalized weights $\bar{w}^{(i)}(z) = w^{(i)}(z)/\sum_j w^{(j)}(z)$. The pair (x_0^*, z) replaces the Langevin-drawn pair in Algorithm 1. All other steps are unchanged.

Proposition 2 (Dataset-resampling correctness)**.** *Let x_0^* be drawn with weights (12). As $m \to \infty$, x_0^* converges weakly to $p(\cdot \mid z)$, and the L^2-minimizer of the bridge training loss converges to $\nabla_{x_t} \log \pi_t$ (score parameterization) or f_t^π (velocity parameterization).*

Proof. Let $w_z(x) = \exp(-\|x - z\|^2/2s^2)$. For any bounded continuous φ, the self-normalized law of large numbers gives, under standard integrability conditions,

$$\sum_i \bar{w}^{(i)}(z)\varphi(x_0^{(i)}) \longrightarrow \frac{\mathbb{E}_\pi[w_z(x)\varphi(x)]}{\mathbb{E}_\pi[w_z(x)]} = \mathbb{E}_{p(\cdot \mid z)}[\varphi(x)].$$

Thus the empirical weights are proportional only to $w_z(x_i)$, while the factor π is supplied by drawing the dataset from π. The result then follows by Theorem 1. □

The weights in (12) are large only for dataset points within distance $\sim s$ of z. Points farther away contribute negligibly. The method therefore fails when z falls in a low-density region of π, where few dataset points lie nearby and the resampled x_0^* is determined by whichever point happens to be closest rather than by the true conditional $p(\cdot \mid z)$. Restrained Langevin handles this case because it actively drives x_0 toward z. Resampling is the better choice when the dataset already covers the neighborhood of z but the pretrained model is miscalibrated there.

D GIBBS SAMPLING DETAILS

Algorithm 2 **Generative Gibbs Sampler**

Input: fine-tuned ψ_θ, α_{t^*}, $s = \sigma_{t^*}/\alpha_{t^*}$, initial probe z_0, iterations K, thinning κ
Output: samples $\{x_0^{(k)}\}_{k=1}^K$ from π (exactly when ψ_θ is exact)

Initialize $z \leftarrow z_0$
for $k = 1$ **to** $K\kappa$ **do**
  Set $x_{t^*} \leftarrow \alpha_{t^*} z$ ▷ *exact bridge endpoint; bypasses $t > t^*$ entirely*
  Run the stochastic reverse SDE (3) from t^* to 0 with ψ_θ to obtain x_0 ▷ *Thm. 1*
  Sample $\eta \sim \mathcal{N}(0, I)$; set $z \leftarrow x_0 + s\,\eta$ ▷ *$z \mid x_0 \sim \mathcal{N}(x_0, s^2 I)$, no score evaluation*
  if $\kappa \mid k$ **then** record $x_0^{(k/\kappa)} \leftarrow x_0$ **end if**
end for
return $\{x_0^{(k)}\}$

Algorithm 2 is a learned proximal Gibbs sampler: its first step samples the restricted-Gaussian conditional with stochastic reverse diffusion, while its second step is an exact Gaussian update (Lee et al., 2021; Kim & Ye, 2023; Chen et al., 2024). Step 1 is called a *proximal sample* of π at z because $p(\cdot \mid z) \propto \pi \exp(-\|\cdot - z\|^2/2s^2)$ is the stochastic analog of the proximal operator familiar from convex optimization, $\mathrm{prox}_{f,s}(z) = \arg\min_x\{f(x) + \frac{1}{2s^2}\|x - z\|^2\}$, with $f = -\log \pi$. Rather than minimizing a penalized objective, we sample from the corresponding penalized distribution. The construction here, including Metropolization, is essentially identical to that introduced recently in Chen et al. (2026a). The key distinction is that here we use restrained data to train the sampler, avoiding the need for π or $\nabla \log \pi$ during training and permitting direct training of marginal samplers.

Algorithm 2 targets π exactly when ψ_θ is exact (Theorem 1), and in practice the error is entirely attributable to the quality of the fine-tuned model and time discretization error, not to the Gibbs

structure. Mixing speed depends on how broadly the probe points $\{z_j\}$ cover the relevant region during training and on the choice of s. Matching s to the local length scale of π improves both coverage and mixing. Algorithm 2 produces correlated rather than independent samples, but because t^* is small relative to T, each Step 1 requires only a fraction of the function evaluations of a full diffusion trajectory.

Though we do not employ it in any of our experiments in this paper, when model error is a concern, Metropolization via Nonequilibrium Candidate Monte Carlo (NCMC) (Nilmeier et al., 2011) is straightforward and eliminates asymptotic bias entirely. The Metropolized chain then targets π exactly, regardless of model quality, at the cost of occasional rejected proposals. As with all Metropolized samplers, the rejection rate will increase with dimension, even when the un-Metroplized sampler's accuracy does not necessarily degrade rapidly (Chen et al., 2026b).

Metropolization by NCMC requires ψ_θ to use a score parameterization ($\psi_\theta \approx \nabla_{x_t} \log \pi_t$), written s_θ in what follows. We write $p_\theta(\cdot \,|\, x_{t_n})$ for the one-step reverse-transition density obtained by an Euler–Maruyama step of the reverse SDE (3) with score s_θ, and $\overleftarrow{X}$ for the time-reversal of a discrete trajectory (i.e. the sequence with its elements in reversed order). Computing the path log-probabilities $\log p_\theta$ requires the SDE drift, which depends explicitly on the score.

To construct the Metropolized scheme, we replace the exact marginal noising step ($z \leftarrow x_0 + s\eta$, $\eta \sim \mathcal{N}(0, I)$) of Algorithm 2 with a discrete simulation of the forward SDE over the time grid $0 = t_0 < t_1 < \cdots < t_N = t^*$, with Gaussian transition kernel

$$q(x_{t_n} \,|\, x_{t_{n-1}}) \;=\; \mathcal{N}\big(x_{t_n};\, \alpha_{t_n|t_{n-1}}\, x_{t_{n-1}},\, \sigma^2_{t_n|t_{n-1}} I\big).$$

This replaces the explicit probe anchor z with a stochastic coupling state x_{t^*} whose path probability is tractable. Denote the forward trajectory $X_{\text{fwd}} = (x_0, x_{t_1}, \ldots, x_{t^*})$ and the reverse denoising trajectory $X_{\text{rev}} = (x'_{t^*}, x'_{t_{N-1}}, \ldots, x'_0)$ produced by running s_θ via Euler–Maruyama from the shared coupling state $x'_{t^*} = x_{t^*}$. The full round-trip from x_0 to x'_0 constitutes a single NCMC proposal with acceptance probability $A = \min(1, \exp(-W))$, where the nonequilibrium work is

$$\begin{aligned} W \;=\; & \underbrace{\big[\log \pi(x_0) - \log \pi(x'_0)\big]}_{W_{\text{energy}}} \;+\; \underbrace{\big[\log q(X_{\text{fwd}} \,|\, x_0) - \log p_\theta(\overleftarrow{X_{\text{fwd}}} \,|\, x_{t^*})\big]}_{W_{\text{out}}} \\ & +\; \underbrace{\big[\log p_\theta(X_{\text{rev}} \,|\, x_{t^*}) - \log q(\overleftarrow{X_{\text{rev}}} \,|\, x'_0)\big]}_{W_{\text{in}}}, \end{aligned} \tag{13}$$

Algorithm 3 **Metropolized Generative Gibbs Sampling (NCMC-Gibbs)**

Input: fine-tuned s_θ (score parameterization), true log-target $\log \pi$, time grid $0{=}t_0{<}t_1{<}\ldots{<}t_N{=}t^*$, initial x_0, iterations K, thinning κ
Output: samples $\{x_0^{(k)}\}_{k=1}^K$ targeting π exactly

for $k = 1$ **to** $K\kappa$ **do**
▷ *1. Forward noising (outward leg)*
  Initialize $X_{\text{fwd}} \leftarrow (x_0)$
  for $n = 1$ **to** N **do**
    Sample $x_{t_n} \sim q(\cdot \,|\, x_{t_{n-1}})$ ▷ *analytic Gaussian kernel*
    Append x_{t_n} to X_{fwd}
  end for; set coupling state $x_{t^*} \leftarrow x_{t_N}$
▷ *2. Reverse denoising (inward leg)*
  Initialize $X_{\text{rev}} \leftarrow (x_{t^*})$; set $x'_{t_N} \leftarrow x_{t^*}$
  for $n = N$ **downto** 1 **do**
    Sample $x'_{t_{n-1}} \sim p_\theta(\cdot \,|\, x'_{t_n})$ ▷ *Euler–Maruyama with s_θ*
    Append $x'_{t_{n-1}}$ to X_{rev}
  end for; set candidate $x'_0 \leftarrow x'_{t_0}$
▷ *3. Nonequilibrium work (Eq. (13))*
  $W_{\text{energy}} \leftarrow \log \pi(x_0) - \log \pi(x'_0)$
  $W_{\text{out}} \leftarrow \log q(X_{\text{fwd}} \,|\, x_0) - \log p_\theta(\overleftarrow{X_{\text{fwd}}} \,|\, x_{t^*})$

$W_{\text{in}} \leftarrow \log p_\theta(X_{\text{rev}} \mid x_{t^*}) - \log q(\overleftarrow{X_{\text{rev}}} \mid x_0')$
$W \leftarrow W_{\text{energy}} + W_{\text{out}} + W_{\text{in}}$
▷ *4. Metropolis–Hastings step*
Draw $u \sim \text{Uniform}(0,1)$; **if** $u < \min\{1, \exp(-W)\}$ **then** $x_0 \leftarrow x_0'$ **end if**
if $\kappa \mid k$ **then** store $x_0^{(k/\kappa)} \leftarrow x_0$ **end if**
end for; **return** $\{x_0^{(k)}\}$

E MÜLLER–BROWN: SETUP AND ADDITIONAL RESULTS

All experiments are run at target inverse temperatures $\beta \in \{1.00, 1.33, 2.00\}$, with the main text reporting $t^* = 0.02$. Densities and reference occupancies are evaluated on a 299×299 quadrature grid over $x \in [-1.5, 1.2]$, $y \in [-0.5, 2.0]$, which contains all three minima. A basin is defined as the connected sub-level region $\{x : U(x) \le U_{\min} + \Delta u\}$ containing the corresponding minimum, with $\Delta u = 0.85$. The reference occupancy p_{ref} is the Boltzmann weight integrated over that region by grid quadrature. For the central basin this gives $p_{\text{ref}} = 3.97 \times 10^{-2}$, 1.89×10^{-2} and 2.99×10^{-3} at $\beta = 1.00$, 1.33 and 2.00, respectively, so the mode of interest carries under 4% of the mass even at the highest temperature.

The two pipelines are matched on total Langevin simulated time, burn-in included. The total budget for restrained data is $25 + 10{,}000 \times 0.25 = 2{,}525$ time units for probe generation and $10{,}000 \times 0.3 = 3{,}000$ on restrained runs, for $5{,}525$ in total. The budget for the baseline model is the same $5{,}525$ time units, spent on one unrestrained trajectory at the target β, seeded from a randomly chosen probe point: 25 units of burn-in followed by recording at the same 0.25 interval, which yields $22{,}000$ points. The baseline therefore receives $2.2\times$ more training data at identical simulation cost.

For the forward diffusion process in equation (1), we use the linear variance-preserving (VP) schedule (Song et al., 2021):

$$\alpha_t^2 = \exp(-0.1t - 9.95t^2), \qquad \sigma_t^2 = 1 - \alpha_t^2. \tag{14}$$

Both models use Adam (Kingma & Ba, 2015) with learning rate 10^{-3}, batch size $1{,}000$ and an Exponential Moving Average (EMA) (Tarvainen & Valpola, 2017) with decay 0.99. Models are trained for $10{,}000$ iterations and evaluated with EMA weights. At sampling time all Gibbs chains start from the same point $(-0.558, 1.442)$, so the chains must locate the remaining basins unaided. Each replicate of the pipeline is run at its own seed, 20 in total, and the t^* sweep below uses the first five of them.

The hyperparameter t^* sets the restraint width through $s = \sigma_{t^*}/\alpha_{t^*}$, so increasing it weakens the harmonic restraint. Its influence is most pronounced at large β, where the target has sharper gradients and taller barriers. Figure 5 reports the central-basin occupancy relative to its exact value, $p_{\text{pred}}/p_{\text{ref}}$, against $t^* \in \{0.02, 0.05, 0.1, 0.2\}$ at each β, with every other parameter held at the values given in Section 6.1; in particular the restrained trajectory length is fixed at 0.3 time units for every t^*, so the simulation budget does not vary along the horizontal axis. This sweep uses the first five of the seeded replicates.

At fixed restrained-trajectory length the occupancy is close to exact at $t^* = 0.02$, with median ratios of 0.81, 0.78 and 0.91 at $\beta = 1.00$, 1.33 and 2.00, respectively. As t^* grows the basin is increasingly overpopulated, especially at larger β. The restrained density is smoothed into a single well by the harmonic term only when that term dominates. As t^* grows the restraint weakens, the barrier structure of βU survives in the restrained density, and sampling training data from it reintroduces the barrier-crossing problem. The fixed-length trajectories used to generate training data then no longer reach the intended conditional, so the training pairs are biased and the learned model inherits that bias. Recovering the correct conditional in this regime would require longer restrained trajectories, or more bridge points per probe, than at small t^*.

Small t^* smooths the restrained density more strongly and so makes the inner sampling problem easier, with an important drawback. Successive Gibbs iterations become strongly correlated, since each sweep perturbs the state only by an amount set by t^* and the effective number of independent samples per iteration therefore falls. In particular, transitions between basins become slower during

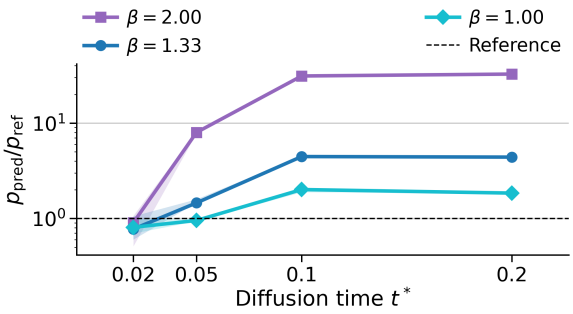

Figure 5: Central-basin occupancy relative to its exact value, $p_{\text{pred}}/p_{\text{ref}}$, against t^* at each inverse temperature β, after 10,000 training iterations. Median and interquartile range are calculated over five independently seeded runs.

sampling. The method thus exhibits a trade-off between the cost of generating the training data and the efficiency of the resulting sampler.

F Alanine dipeptide: setup and additional results

F.1 System, preprocessing, and reference distribution

Explicit-solvent molecular dynamics. The solvated system contains a 22-atom alanine-dipeptide solute and 628 TIP3P water molecules in a cubic box of side length 2.7222 nm (1906 atoms total). We use the OpenMM (Eastman et al., 2017) implementations of the AMBER99SB-ILDN protein force field (Lindorff-Larsen et al., 2010) and TIP3P water, particle-mesh Ewald (Darden et al., 1993) electrostatics with a 0.9 nm real-space cutoff, constrained solute hydrogen bonds, rigid water, and center-of-mass motion removal. All trajectories use a Langevin-middle integrator (Zhang et al., 2019) with friction $1\,\text{ps}^{-1}$ and step size 2 fs. After at most 2000 energy-minimization iterations, velocities are drawn at the simulation temperature and the system is propagated for 10 ps before the first stored frame. Production calculations use single-precision CUDA.

Five independent 300 K trajectories are generated with seeds 9000–9004 and saved every 1 ps. The reported experiment uses each trajectory over 0–5.05 μs, treats the first 50 ns as warm-up, and hence has 25.0 μs of production data and a total reported MD cost of 25.25 μs. A separate 250 ns 500 K trajectory (seed 9500) stores the full solvated state every 10 ps, supplying the 5000 probe candidates used in Setting II. Water is retained during MD but discarded before diffusion-model training.

Each equilibrium solute frame is Kabsch-aligned, using all 22 solute atoms, to a fixed reference structure used consistently across all experiments. The restrained-MD preprocessing described below additionally makes each solute whole under periodic boundary conditions. Coordinates are expressed in nm and multiplied by 20 before entering the model. Base-model training uses a 50 ps subsample of the production interval.

Ramachandran reference and basins. The reference histogram uses all 25,000,000 raw 1 ps frames in $(0.05, 5.05]$ μs from the five 300 K trajectories. We form a 300×300 periodic histogram over $[-\pi, \pi)^2$, smooth its density with a wrapped Gaussian of standard deviation two bins, and define the dimensionless free energy $F = -\log p + \text{const}$. We identify the five distinct local minima listed in Table 1. Basin R_i is the four-neighbor, periodically connected component containing minimum i and satisfying $F - F_i \le \Delta F_i$. This construction gives $p_{\text{ref}}(R_5) = 0.001574$ (0.1574%).

Basin	minimum ϕ (rad)	minimum ψ (rad)	$\Delta F/k_{\mathrm{B}}T$
R_1	-2.52374610	2.75412956	0.6
R_2	-1.24616509	2.67035376	1.0
R_3	-1.30899694	-0.30368729	1.5
R_4	0.93200582	0.53407075	3.0
R_5	1.03672558	3.02640092	2.0

Table 1: Local-minimum coordinates and basin-specific free-energy barriers used for all ALDP evaluations.

F.2 RESTRAINED-PAIR CONSTRUCTION

Shared Cartesian CV. Let $A \in \mathbb{R}^{30\times 66}$ select the Cartesian coordinates of the ten solute heavy atoms with zero-based atom indices $(1, 4, 5, 6, 8, 10, 14, 15, 16, 18)$. We use the same VP schedule (14) for the diffusion model. At $t^* = 0.40$, $(\alpha_{t^*}, \sigma_{t^*}) = (0.4421969, 0.8969180)$ and the matched restraint width is $s = \sigma_{t^*}/\alpha_{t^*} = 2.028323$ in scaled model coordinates, or 0.101416 nm in physical coordinates.

Setting I: synthetic pairs. For each MD-budget level, only its matching equilibrium pool is used. The Ramachandran plane is divided into a 50×50 grid. If an occupied cell contains n_{ij} frames, its probe quota is proportional to $n_{ij}^{0.5}$, with integer remainders assigned by largest fractional part and quotas capped by cell occupancy. We then draw 10,000 distinct equilibrium probes without replacement within their cells. For each training run and MD-budget level, the 10,000 probes and their posterior-matched synthetic pairs are independently resampled using seed-specific random streams.

Each clean probe produces 30 pairs. In scaled coordinates, a target CV is drawn as

$$y = A\tilde{x} + s\xi, \qquad \xi \sim \mathcal{N}(0, AA^\top).$$

A source frame x_0 is drawn by discrete resampling from the linear-CV marginal in Equation (10), with the endpoint target in that equation set to $\alpha_{t^*} y$. Equivalently, its unnormalized resampling weight is

$$w(x_0 \mid y) \propto \exp\left[-\frac{1}{2s^2}(Ax_0 - y)^\top (AA^\top)^{-1}(Ax_0 - y)\right].$$

Given x_0, the conditional distribution in Equation (11), again evaluated at endpoint target $\alpha_{t^*} y$, gives the normalized endpoint

$$z = \frac{x_{t^*}}{\alpha_{t^*}} = A^\top (AA^\top)^{-1} y + \left[I - A^\top (AA^\top)^{-1} A\right](x_0 + s\epsilon), \qquad \epsilon \sim \mathcal{N}(0, I).$$

Therefore $Az = y$ exactly. Every budget contains 300,000 synthetic pairs and requires no additional MD or energy evaluations. The equilibrium-pool mean is subtracted before lower-time training and restored after sampling.

Setting II: restrained-MD pairs. The 250 ns of the 500 K trajectory is sampled every 50 ps, giving 5000 candidates that include the complete solvent state and periodic box. For each candidate, Gaussian noise of standard deviation 0.101416 nm is added only to the ten selected heavy atoms to define a restraint center. Starting from the corresponding solvated state, velocities are redrawn at 300 K and a 5 ns trajectory is run without minimization using the same physical model as above. We implement the harmonic factor in Equation (10) in physical coordinates using minimum-image displacements:

$$U_{\mathrm{rest}}(x) = \frac{k}{2} \sum_{a\in\mathcal{H}} \mathrm{periodicdistance}(x_a, z_a)^2, \qquad k = \frac{RT}{s^2} = 242.52\ \mathrm{kJ\,mol^{-1}\,nm^{-2}}.$$

Frames are stored every 10 ps, but only the 30 frames at 2.1,2.2,...,5.0 ns are retained. Before pair construction, each solute is made whole, the restraint displacement is taken by the minimum-image convention, and the configuration and target share the same framewise Kabsch transform. The unconstrained component of z is then resampled analytically as above, which preserves $Az = y$ exactly. The full dataset contains 150,000 pairs, and its centering shift is the mean of the 5000 aligned high-temperature candidates.

F.3 MATCHED MD BUDGETS AND REPEATED RUNS

For the MD-budget curve, all five equilibrium trajectories are retained and their production intervals are shortened together. The restrained method uses a matching number of 5 ns trajectories, so both sides have cost $5L + 0.25\,\mu$s at level L (Table 2). The $25.0\,\mu$s reference distribution is used to evaluate every level and is not treated as training data at smaller budgets.

	Base-model training dataset		Restrained-MD-model training dataset	
Matched MD budget (μs)	Production interval per trajectory (μs)	Training configurations	Restrained-MD probes	Retained training pairs
5.25	$(0.05, 1.05]$	100,000	1,000	30,000
10.25	$(0.05, 2.05]$	200,000	2,000	60,000
15.25	$(0.05, 3.05]$	300,000	3,000	90,000
20.25	$(0.05, 4.05]$	400,000	4,000	120,000
25.25	$(0.05, 5.05]$	500,000	5,000	150,000

Table 2: Comparison of the two model-training datasets at matched MD budgets. The base-model dataset pools configurations from five equilibrium trajectories after discarding the first 50 ns of each trajectory and subsampling the remaining production segments every 50 ps. The restrained-MD-model dataset is constructed from a 250 ns high-temperature probe trajectory and matched restrained simulations. After discarding the equilibration portion of each restrained trajectory, 30 target configurations are retained per probe. Consequently, the two approaches use the same total MD time but different numbers of retained training configurations.

We use 20 runs with training seeds 42–61. Base training uses the same equilibrium pool across seeds. For synthetic refinement, the probes and posterior-matched pairs are independently resampled for every training seed at each MD-budget level. For restrained MD we uniformly select probes without replacement. At the maximum budget all runs use all 5000 probes.

F.4 MODEL, OPTIMIZATION, AND SAMPLING

The denoiser treats the 22 atoms as tokens. A two-layer GELU (Hendrycks & Gimpel, 2023) MLP maps each 3-vector to width 128; learned atom-slot and atomic-number embeddings and a sinusoidal time embedding followed by a two-layer GELU MLP are added. Three pre-norm Transformer (Vaswani et al., 2017) encoder layers use eight attention heads, feed-forward width 512, GELU activations, and zero dropout. A linear head predicts one three-dimensional noise vector per atom.

All models use batch size 4096, Adam with constant learning rate 10^{-3}, and an EMA with decay 0.99. The base model is trained from scratch for 50,000 updates with diffusion time t uniform on $[10^{-4}, 1 - 10^{-4}]$. Synthetic refinement initializes both the live and EMA models from the base-20k EMA checkpoint, resets the optimizer and update counter, and performs 30,000 bridge updates with diffusion time t uniform on $[10^{-4}, t^*]$; no equilibrium examples are mixed into these updates. The restrained-MD model is initialized randomly and trained from scratch for 50,000 lower-time bridge updates. The training-budget curve saves and evaluates checkpoints every 2000 updates.

Each evaluation uses 100,000 generated solute configurations in two batches of 50,000. Euler–Maruyama reverse sampling uses $\frac{1}{300}$ time step for base model and $\frac{t^*}{100}$ time step for synthetic fine-tune model. Base sampling runs the 50k model from standard Gaussian noise over $1 \to 0$. Synthetic one-shot sampling runs the base-20k model over $1 \to t^*$, shifts to the centered coordinate system, and runs the fine-tuned model over $t^* \to 0$. Restrained-MD evaluation initializes 100,000 Gaussian states at t^* and runs one $t^* \to 0$ reverse pass; this is Gibbs iteration 0. Each of the next 50 iterations applies the analytic forward update $x_{t^*} = \alpha_{t^*} x_0 + \sigma_{t^*} \epsilon$ followed by a 100-point reverse pass. Reported restrained-MD metrics are the arithmetic average of the ten per-iteration metrics at iterations 41–50. PyTorch deterministic algorithms are enabled, cuDNN benchmarking is disabled.

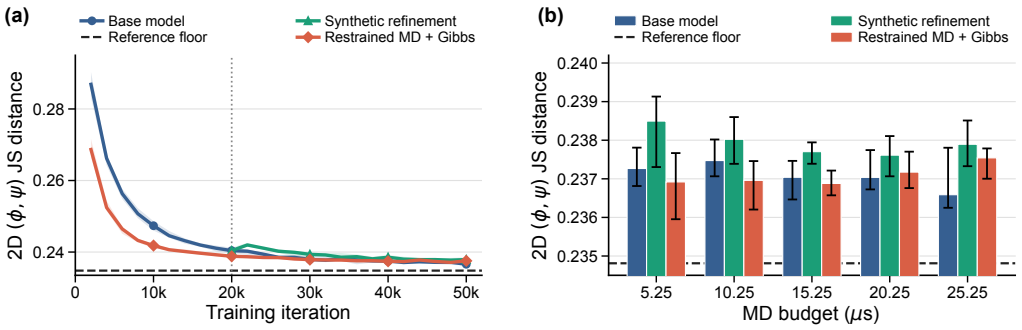

Figure 6: Two-dimensional Ramachandran JS distance versus (a) training updates at a fixed 25.25 μs MD budget and (b) matched MD budget at 50k updates. Curves and bars show medians over 20 runs, and bands and error bars show interquartile ranges. Dashed lines indicate the expected JS distance between a 100,000-sample multinomial draw and the 25.0 μs reference histogram.

F.5 Metrics and additional results

For every generated set, (ϕ, ψ) is accumulated on the same 300×300 grid as the reference. With $m = (p + q)/2$, the reported global metric is

$$d_{\mathrm{JS}}(p, q) = \left[\tfrac{1}{2}\mathrm{KL}_2(p\|m) + \tfrac{1}{2}\mathrm{KL}_2(q\|m)\right]^{1/2}, \tag{15}$$

where zero-count terms are omitted. The rare-state relative error is $p_{\mathrm{gen}}(R_5)/p_{\mathrm{ref}}(R_5) - 1$. Figure 3 reports medians across the 20 runs; bands and error bars span the first and third quartiles. For restrained MD, the last-ten-iteration average described above is computed within each run before the across-run median and quartiles are taken.

Because the generated histogram contains only 100,000 samples spread over 90,000 bins, even an exact generator has nonzero empirical JS distance. We estimate this finite-sample floor by 5000 independent multinomial draws of 100,000 samples from the 25.0 μs reference histogram; its mean is 0.234812. Figure 6 shows that all three methods approach this floor during training and remain close to it throughout the MD-budget sweep, supporting the conclusion that the improvement in R_5 does not come at the expense of global fidelity.

G Chignolin: setup and additional results

G.1 Reference simulation and collective variables

We simulate CLN025 (YYDPETGTWY) at 340 K with OpenMM using the AMBER14 ff14SB protein force field (Maier et al., 2015) and GBN2 implicit-solvent model (Nguyen et al., 2013) using the protein.ff14SB.xml and gbn2.xml definitions supplied with OpenMM. The system uses no nonbonded cutoff, constrains bonds to hydrogen, removes center-of-mass motion, and sets the solute and solvent dielectric constants to 1 and 78.5. A Langevin-middle integrator uses a 2 fs step and friction $1\ \mathrm{ps}^{-1}$. Velocities are initialized at 340 K with seed 2026061800. The trajectory is propagated for 22.36 μs and stored every 10 ps, yielding 2,236,000 frames.

Only the 93 heavy atoms are retained. Coordinates are converted to Å and each frame is Kabsch-aligned, using the ten Cα atoms, to a fixed folded reference structure used consistently across all experiments. The two evaluation coordinates are

$$\mathrm{NO1} = d(\mathrm{Asp3\,N}, \mathrm{Gly7\,O}), \qquad \mathrm{NO2} = d(\mathrm{Asp3\,N}, \mathrm{Thr8\,O}). \tag{16}$$

The folded region shown in Figure 4(a) is $4.5 < \mathrm{NO1} < 8.0$ Å and $\mathrm{NO2} < 3.6$ Å. The misfolded region used for every quantitative result is

$$\mathcal{R}_{\mathrm{mis}} = \{\mathrm{NO1} < 3.7\,\text{Å},\ 5.0 < \mathrm{NO2} < 8.0\,\text{Å}\}. \tag{17}$$

G.2 Data budgets and fixed reference

The data-scaling experiment uses only the first 22 μs. We discard the first 2 μs as warm-up and define a single reference distribution from the remaining 2,000,000 frames in $(2, 22]$ μs. This reference contains 7494 misfolded frames, hence $p_{\mathrm{ref}}(\mathcal{R}_{\mathrm{mis}}) = 0.003747$. All generated samples, including models trained at smaller data budgets, are compared with this same reference.

The four training windows are prefixes of the post-warm-up interval (Table 3). For each window we recompute the per-atom Cartesian center and one global scale, the square root of the mean coordinate variance. No frame after the stated endpoint contributes to its normalization, autoencoder, latent statistics, probes, or diffusion training.

MD budget (μs)	Source interval (μs)	Frames
7	$(2, 7]$	500,000
12	$(2, 12]$	1,000,000
17	$(2, 17]$	1,500,000
22	$(2, 22]$	2,000,000

Table 3: Chignolin training windows. The 2–22 μs reference is fixed across all four data budgets.

G.3 Autoencoder and latent representation

We train one independent autoencoder for each data budget and share that representation across all ten diffusion seeds at that budget. The encoder and decoder each have width 512 and four residual blocks. A block applies layer normalization followed by two linear layers with a SiLU activation and adds the result to the skip connection with a $1/\sqrt{2}$ factor. The bottleneck has dimension 128.

With coordinates in their standardized units and distances in Å, let $\tilde{\mathbf{r}}_{na}$ and $\hat{\tilde{\mathbf{r}}}_{na}$ denote the input and reconstructed Cartesian coordinates of atom a in configuration n, and let $\mathbf{r}_{na}$ and $\hat{\mathbf{r}}_{na}$ be the corresponding coordinates after transforming back to Å. For a batch of B configurations, the four loss terms are

$$\mathcal{L}_{\mathrm{coord}} = \frac{1}{279B} \sum_{n=1}^{B} \sum_{a=1}^{93} \left\| \hat{\tilde{\mathbf{r}}}_{na} - \tilde{\mathbf{r}}_{na} \right\|_2^2,$$

$$\mathcal{L}_{\mathrm{bond}} = \frac{1}{98B} \sum_{n=1}^{B} \sum_{(a,b)\in\mathcal{E}_{\mathrm{bond}}} \left(\|\hat{\mathbf{r}}_{na} - \hat{\mathbf{r}}_{nb}\|_2 - \|\mathbf{r}_{na} - \mathbf{r}_{nb}\|_2 \right)^2,$$

$$\mathcal{L}_{\mathrm{C}\alpha} = \frac{1}{45B} \sum_{n=1}^{B} \sum_{(a,b)\in\mathcal{E}_{\mathrm{C}\alpha}} \left(\|\hat{\mathbf{r}}_{na} - \hat{\mathbf{r}}_{nb}\|_2 - \|\mathbf{r}_{na} - \mathbf{r}_{nb}\|_2 \right)^2,$$

and

$$\mathcal{L}_{\mathrm{NO}} = \frac{1}{2B} \sum_{n=1}^{B} \sum_{k=1}^{2} \left(\hat{d}_{nk}^{\mathrm{NO}} - d_{nk}^{\mathrm{NO}} \right)^2.$$

Here, $\mathcal{E}_{\mathrm{bond}}$ contains the 98 covalent heavy-atom bonds in the fixed molecular topology, while $\mathcal{E}_{\mathrm{C}\alpha}$ contains all $\binom{10}{2} = 45$ pairwise distances among the ten Cα atoms. The final term explicitly preserves NO1 and NO2, the two distances used to distinguish the folded and misfolded conformations. The complete objective is

$$\mathcal{L}_{\mathrm{AE}} = \mathcal{L}_{\mathrm{coord}} + 0.1\mathcal{L}_{\mathrm{bond}} + 0.1\mathcal{L}_{\mathrm{C}\alpha} + \mathcal{L}_{\mathrm{NO}}.$$

Thus, the coordinate term controls overall reconstruction, the bond term preserves local molecular geometry, the Cα term preserves the global backbone arrangement, and the NO term emphasizes reconstruction accuracy in the rare-state collective variables.

We use Adam, batch size 1024, gradient clipping at 1, and bfloat16 autocasting. The learning rate warms linearly to 2×10^{-4} over 2000 updates and follows a cosine schedule with 100,000-update

horizon and minimum multiplier 0.1. Training stops at the fixed 60,000-update endpoint; the common autoencoder seed is 2026090744. Across the four validation splits, the coordinate RMSD is 0.0406–0.0420 Å and the decoded-to-original misfolded-population ratio is 0.983–1.008.

After encoding every frame, each latent coordinate is standardized. Its mean and standard deviation are estimated from at most 200,000 evenly spaced frames in the corresponding training window. All posterior construction, diffusion training, and reverse sampling below occur in this standardized 128-dimensional latent space.

G.4 Probe allocation and synthetic posterior pairs

For each MD budget, the $(\mathrm{NO1}, \mathrm{NO2})$ plane of that training window is divided into a 50×50 grid. If cell b contains n_b frames, 10,000 distinct probe frames are allocated in proportion to $n_b^{0.5}$; integer remainders are assigned by largest fractional part. Sampling within cells is uniform without replacement.

We use the same linear VP schedule as Equation (14). Let $\tilde{z}_j$ be a standardized latent probe and $s_{t^*} = \sigma_{t^*}/\alpha_{t^*}$. We draw a perturbed restraint center $y_j = \tilde{z}_j + s_{t^*}\xi_j$, where $\xi_j \sim \mathcal{N}(0, I)$. For each y_j, we sample 30 clean endpoints with replacement from the empirical version of the restrained distribution in Equation (5), taking $z = y_j$ and $s = s_{t^*}$ and assigning equal base mass to every latent frame. By Lemma 1, these are exact samples from the corresponding empirical diffusion posterior. This produces 300,000 bridge pairs per data budget without additional MD. We evaluate $t^* = 0.30$ and 0.35, for which $(\alpha_{t^*}, \sigma_{t^*}, s_{t^*})$ equals $(0.62955, 0.77696, 1.23415)$ and $(0.53423, 0.84534, 1.58237)$. A pair set is shared across the ten diffusion seeds at fixed data budget and t^*.

G.5 Diffusion training and one-shot evaluation

The denoiser is a time-conditioned residual MLP with width 1024, eight FiLM (Perez et al., 2018) residual blocks, and a 256-dimensional Fourier time embedding (Vaswani et al., 2017). It predicts VP noise in the standardized latent space. We use seeds 42–51, batch size 2048, Adam, EMA decay 0.99, gradient clipping at 1, and bfloat16 autocasting. The learning-rate schedule warms to 2×10^{-4} over 2000 updates and then decays by the same cosine rule to 0.1 of its peak at 160,000 updates.

The base model is trained for 160,000 updates with diffusion time $t \sim \mathrm{Uniform}(10^{-4}, 1 - 10^{-4})$. Synthetic refinement copies the complete live model, EMA, optimizer, and random state at the 80,000-update checkpoint and performs 80,000 additional bridge updates with diffusion time $t \sim \mathrm{Uniform}(10^{-4}, t^*)$. The optimizer and global learning-rate clock are not reset, and no equilibrium examples are mixed into these lower-time updates. The $t^* = 0.30$ runs reuse the audited autoencoders and base trajectories from the $t^* = 0.35$ experiment; exact Base@80k and Base@160k state hashes are checked before reuse.

Every evaluation generates 100,000 configurations with the EMA parameters. Euler–Maruyama sampling uses one common 300-transition grid. Base sampling uses Base@160k from 1 to 0. Synthetic one-shot sampling uses Base@80k from 1 to t^* and the refined model from t^* to 0. The same evaluation random stream is used for Base and both synthetic branches within each seed, yielding paired comparisons. Training curves evaluate every 10,000 updates; the smaller data budgets retain only the final 160,000-update evaluation.

G.6 Metrics and bridge-time sensitivity

The fixed 2–22 μs reference defines 100 interior bins along each NO coordinate, with endpoints at its 10^{-4} and 0.9999 quantiles. Explicit underflow and overflow bins give a 102×102 probability table. We compare the generated and reference tables using the base-2 JS distance in Equation (15). The expected finite-sample JS distance is 0.09822, estimated from 200 independent 100,000-sample resamples of the fixed 2–22 μs reference ensemble. Rare-state accuracy is the generated fraction $p_{\mathrm{gen}}(\mathcal{R}_{\mathrm{mis}})$. Figures and tables report the median over ten independently trained diffusion models; bands and error bars span the first and third quartiles.

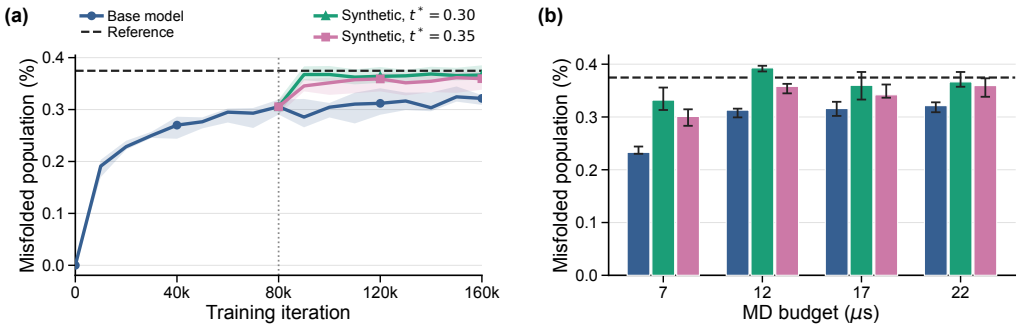

Figure 7: Chignolin bridge-time comparison. Generated misfolded population versus (a) training iteration at a fixed 22 μs MD budget and (b) MD budget at 160k total updates. Curves and bars show medians over ten seeds; bands and error bars show interquartile ranges. The dashed line is the fixed reference population, 0.3747%, and the dotted line in (a) marks the 80k branch point.

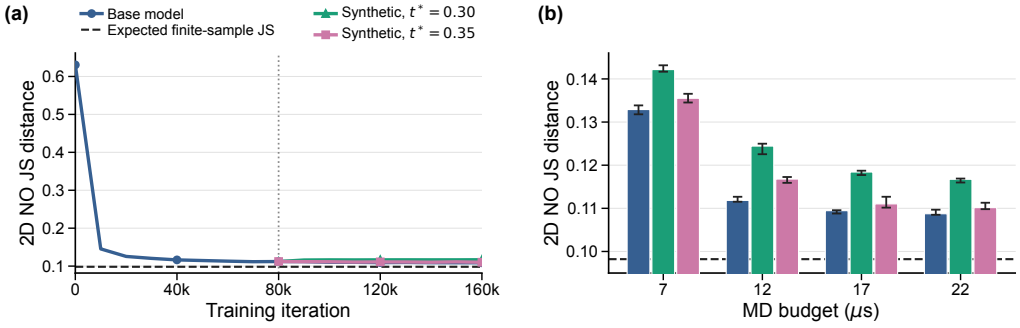

Figure 8: Chignolin global fidelity. Two-dimensional NO JS distance versus (a) training iteration at a fixed 22 μs MD budget and (b) MD budget at 160k total updates, for both $t^* = 0.30$ and 0.35. Curves and bars show medians over ten seeds; bands and error bars show interquartile ranges. Dashed lines indicate the expected finite-sample JS distance, 0.09822, estimated from 200 independent 100,000-sample resamples of the fixed 2–22 μs reference ensemble. The dotted line in (a) marks the 80k branch point.

MD budget (μs)	Method	$p_{\mathrm{gen}}(\mathcal{R}_{\mathrm{mis}})$ (%)	|relative error| (%)	2D NO JS distance	
		Median [Q1, Q3]		Median	[Q1, Q3]
7	Base	0.2330 [0.2302, 0.2440]	37.8	0.13290	[0.13179, 0.13386]
	Synthetic, $t^* = .30$	0.3320 [0.3132, 0.3558]	11.4	0.14217	[0.14167, 0.14316]
	Synthetic, $t^* = .35$	0.3010 [0.2833, 0.3145]	19.7	0.13552	[0.13453, 0.13655]
12	Base	0.3130 [0.2993, 0.3158]	16.5	0.11184	[0.11153, 0.11268]
	Synthetic, $t^* = .30$	0.3930 [0.3862, 0.3970]	4.9	0.12443	[0.12254, 0.12498]
	Synthetic, $t^* = .35$	0.3580 [0.3448, 0.3628]	4.5	0.11679	[0.11591, 0.11726]
17	Base	0.3160 [0.3020, 0.3287]	15.7	0.10946	[0.10880, 0.10956]
	Synthetic, $t^* = .30$	0.3600 [0.3330, 0.3852]	3.9	0.11842	[0.11772, 0.11872]
	Synthetic, $t^* = .35$	0.3420 [0.3365, 0.3615]	8.7	0.11103	[0.11016, 0.11268]
22	Base	0.3215 [0.3090, 0.3277]	14.2	0.10875	[0.10850, 0.10968]
	Synthetic, $t^* = .30$	0.3665 [0.3572, 0.3852]	2.2	0.11673	[0.11599, 0.11690]
	Synthetic, $t^* = .35$	0.3595 [0.3382, 0.3732]	4.1	0.11016	[0.10981, 0.11129]

Table 4: Final Chignolin metrics at 160k total updates. Relative error is computed from each across-seed median and the fixed reference population 0.3747%.

Both bridge times improve the median misfolded population over the base model at every data budget, but their quantitative tradeoffs differ. $t^* = 0.30$ is usually closer in rare-state population, while $t^* = 0.35$ usually incurs a smaller increase in 2D NO JS distance. At 22 μs, for example, the absolute relative population error is 2.2% at $t^* = 0.30$ and 4.1% at $t^* = 0.35$, whereas the median JS distances are 0.11673 and 0.11016. Both nearby choices consistently improve the rare-state estimate, so performance is not sharply sensitive over the tested range; however, we do not yet have a general strategy for choosing the optimal t^*.