Sample Efficient Generative Model for Molecular Dynamics Trajectories via Twisted Sequential Monte Carlo

Qijia Jiang
qjang@ucdavis.edu
UC Davis

Abstract

We study conditional generation of molecular dynamics trajectories, moving beyond unconditional Boltzmann equilibrium sampling from $\propto e^{-U}$. The motivation is inference-time path conditioning: given an initial frame together with constraints such as terminal states, intermediate frames, masks, or general trajectory-level rewards, the goal is to sample a full trajectory with the correct dynamical law. We formulate this problem as path-space inference under a Markov reference process and develop a hierarchy of twisted Sequential Monte Carlo methods built on learned score-based proposals. The resulting algorithms separate learning the reference dynamics from imposing new rewards at inference time, require only sample access to equilibrium frames and reference trajectories, and remain asymptotically exact as the number of particles $K \to \infty$ with controlled variance.

1. Introduction

Many applications of molecular dynamics trajectory generation require more than unconditional forward simulation (e.g., condition on experimentally measured intermediate states or study transition path between protein conformations). Typically the initial frame $x_0 \in \mathbb{R}^d$ is known together with molecular context such as an amino-acid sequence (Jing et al., 2024) or molecular graph (Iyengar et al., 2025), and one would like to generate trajectories consistent with additional information specified at inference time: terminal states, intermediate frames, masked coordinates, or more general trajectory-level rewards. Such tasks require modeling a distribution on the *whole paths* $x_{1:T} \in \mathbb{R}^{T\times d}$ rather than only adjacent-frame transitions $p(x_t, x_{t+1})$. For example, conditioning on the terminal state x_T requires sampling transition paths with the correct dynamical probability, which cannot be recovered from Boltzmann sampling alone.

We study this problem with twisted SMC, which propagates partial trajectories one frame at a time while allowing future information to influence present samples through particle weights and twists. Our setup has 2 distinct time axes: diffusion time s[1] and trajectory time t. Diffusion models operate along s, whereas SMC advances along $t \in [T]$ and marginalizes over the resulting sequence of latent variables. The target path measure we are interested in generating samples from can be written as

$$\begin{aligned} Q(x_{1:T}|r, x_0) &\propto p(x_1|x_0)p(x_2|x_1)\dots p(x_T|x_{T-1}) \\ &\times r(x_1, \dots, x_T) =: \frac{1}{Z(x_0)} Q^{\text{ref}}(x_{1:T}|x_0) \times r(x_{1:T}) \end{aligned} \tag{1}$$

where r is a positive bounded reward encoding conditioning information on the entire trajectory, and Q^{ref} is a Markov reference path measure, as a typical autoregressive MD trajectory is. The non-Markovian dependence enters through $r(x_{1:T})$, which is precisely what makes *joint* path-space modeling necessary.

We assume time-homogeneous reference dynamics, so the transition kernel $p(x_{t+1}|x_t)$ is the same across $t \in [T]$. Under the unconditioned reference process, the marginal law of each frame is the same Boltzmann distribution $\propto e^{-U}$.

Accepted to FoGen 2026: Foundations of Deep Generative Models: Understanding Memorization, Generalization, and Reasoning, an ICML 2026 workshop (non-archival).

[1] s indexes the noise level in the OU process, with $s = S$ corresponding to noise and $s = 0$ to clean data.

Throughout, we assume sample access to both equilibrium frames and reference trajectories Q^{ref} – such data are used to learn score models, but not analytical access to the corresponding densities or transition kernels. In particular, we do not require a closed-form expression for $p(x_{t+1}|x_t)$, which is important in molecular settings where one may wish to model relatively large time increments Δt for which small-step Gaussian approximations are inaccurate.

Sections 4-6 develop a hierarchy of conditional path-generation algorithms for (1). The motivating use case is: given trajectory data collected from classical expensive simulations across many systems, together with static equilibrium-like structures (assumed to be local minima of energy landscape therefore capture the high probability modes), we want to generate plausible trajectories for a new system under arbitrary inference-time constraints, *without retraining* the underlying generative model, or access to labeled reward pairs $x_{1:T}, r(x_{1:T})$. This serves as an efficient and adaptable alternative to traditional MD simulators. We specialize our discussion here to MD simulations, but such problem setups are also ubiquitous in video generation and our methodology carries over.

2. Related Work

Trajectory generation for molecular systems has been studied from several complementary perspectives. Boltzmann generators and related methods (Noé et al., 2019) target equilibrium distributions over configurations rather than dynamical path measures. Timewarp (Klein et al., 2023) learns a system-specific normalizing-flow transfer operator for a fixed lag time Δt; because it is autoregressive, it supports forward simulation but not trajectory-level inference-time conditioning, and it does not naturally transfer across systems with different equilibrium distributions. MDGen (Jing et al., 2024) instead models full trajectory distributions and therefore supports tasks such as interpolation, upsampling, and inpainting. Our setting is closer in spirit to (Jing et al., 2024) in that we target path measures rather than only one-step transitions, but compared to joint diffusion over the full trajectory space $\mathcal{P}(\mathbb{R}^{d\times T})$, we leverage twisted SMC that only learns diffusion model for $p^{\text{ref}}(x_{t+1}|x_t)$ and becomes more sample-efficient. Compared to (Schreiner et al., 2023), our approach is *transferrable* across systems with different Boltzmann equilibrium distributions, and is guaranteed to generate from the target (1). See also (Bigi et al., 2025; Jing et al., 2025) for recent developments and reviews in this area.

More broadly, our work is related to recent uses of SMC for score-based diffusion models in $\mathbb{R}^d$ (Phillips et al., 2024; Wu et al., 2025; Chen, Junhua and Richter, Lorenz and Berner, Julius and Blessing, Denis and Neumann, Gerhard and Anandkumar, Anima, 2024), as well as to SMC-based conditional generation and fine-tuning methods (Uehara et al., 2025; Wu et al., 2023). These works typically focus on static targets or conditioning problems for a single frame. In this work, the target is a trajectory law with both local Markov structure through $p(x_{t+1}|x_t)$ and global conditioning through a trajectory-level reward $r(x_{1:T})$. The former we have samples from, but no analytical expression; the latter no samples but assumed easy to evaluate. Our decomposition separates these two roles cleanly: learning is used for the reference dynamics, while SMC weights and twists handle new trajectory-level rewards online without retraining.

Controlled-drift approaches vs. SMC. An alternative paradigm is to learn a controlled drift that directly generates from the target path measure, thereby avoiding importance weights. This includes variational formulations of Doob's h-transform (Domingo-Enrich et al., 2025; Havens et al., 2025) and path-integral stochastic control methods (Levine, 2018). These approaches parametrize a Markov control $u_\theta(x_t, t)$ and optimize $D_{\text{KL}}(Q^{u_\theta}\|Q^*)$ or related objectives. They are most natural when the optimal control is itself Markov, as for endpoint rewards $r(x_T)$ or factored statewise potentials $\prod_t G_t(x_t)$. For a general trajectory reward $r(x_{1:T})$, however, the optimal control depends on the full history $x_{1:t}$, so a Markov parametrization $u_\theta(x_t, t)$ is fundamentally misspecified. Our SMC formulation imposes no such restriction: particles carry the full partial trajectory, the twist $\hat{\psi}_t(x_{1:t})$ may depend on the entire history, and asymptotic exactness is preserved even with an imperfect twist (Proposition 1). Appendix D studies a hybrid approach that combines controlled proposals with SMC correction, retaining the correctness guarantees of SMC while benefiting from improved proposals when the control is accurate.

3. SMC Ingredients

We impose conditioning at inference time through SMC, without retraining the underlying score models for each new reward. The key advantage of SMC is look-ahead: partial trajectories $x_{1:t}$ can be informed by future constraints on $x_{t+1:T}$, so promising prefixes receive more weight before the final time step. This avoids the high-variance alternative of sampling full trajectories from Q^{ref} and applying the reward only at the end, which becomes statistically unstable as

the horizon T grows.

Formally, SMC introduces a sequence of intermediate targets, determined by twists ψ_t, together with proposals $q_t(x_t \mid x_{0:t-1})$ for $t \in [T]$. The algorithm produces weighted particles approximating each intermediate law and an unbiased estimator of its normalizing constant. The unnormalized targets are

$$\pi_t(x_{0:t}) = p^{\text{ref}}(x_{0:t})\psi_t(x_{0:t}) \tag{2}$$

and we write $\mathbb{Q}_t(x_{0:t}) := \pi_t(x_{0:t})/\int \pi_t(x_{0:t})\, dx_{0:t}$ for the corresponding normalized law. The terminal choice $\psi_T(x_{0:T}) = r(x_{0:T})$ guarantees that the final normalized target at T coincides with the desired target path measure. The optimal twists in (2) can be shown to be

$$\psi_t^*(x_{0:t}) = \int p^{\text{ref}}(x_{t+1:T}|x_{0:t})r(x_{0:T})\, dx_{t+1:T} \tag{3}$$

which require marginalizing over future states. Under (3), the normalized law satisfies $\mathbb{Q}_t(x_{0:t}) = Q(x_{0:t} \mid r)$ for every t, so the particle system matches the true target marginals at all stages and no additional correction is needed at the terminal step.

In Sections 4 and 5, we approximate (3) by replacing the future-state integral with point predictions or posterior means, following the general strategy of (Wu et al., 2023). The important distinction in our setting is that reweighting here (surprisingly) doesn't have to be done over each diffusion step s, which will lead to a large number of weight calculation (ST in total for path-space proposal). We show in this work that even without analytical form of $p^{\text{ref}}(x_{t+1}|x_t)$ available the re-weighting can be done at the level of each t (therefore in total T weight calculation), and can account for future states appropriately.

We rely on a variant of SMC called *nested SMC* that renders the weight $w_t(x_{0:t-1}, x_t)$ after each diffusion proposal step $q_t(x_t|x_{0:t-1})$ computable while maintaining exactness guarantees. This introduces an inner importance sampler in each step. See (Uehara et al., 2025, Section 3.3) and (Naesseth et al., 2019, Algorithm 5) for general nested-SMC background, and (Wu et al., 2025) for a recent diffusion-based instance over $\mathcal{P}(\mathbb{R}^d)$.

In this work we use the one-step optimal proposal (up to normalizing constant as a function of $x_{0:t-1}$)

$$\begin{aligned} q_t(x_t|x_{0:t-1}) &\propto p_\theta^{\text{ref}}(x_t|x_{0:t-1})\hat{\psi}_t(x_{0:t-1}, x_t) \\ &\propto \frac{\pi_t(x_{0:t})}{\pi_{t-1}(x_{0:t-1})} \end{aligned} \tag{4}$$

for a given approximate twist $\hat{\psi}_t$ (Zhao et al., 2024, Proposition 3.3). Lemma 4 in Appendix A.1 makes the associated optimality statement precise. We discuss tractable objectives to learn the optimal twists (3) in Section 6. Sections 4-6 below instantiate the SMC scaffold above in 3 increasingly capable ways.

4. Take 1: look-ahead with cheap twists

In this section, we begin by training a one-step $t \to t+1$ conditional diffusion model targeting $p^{\text{ref}}(x_{t+1}|x_t)$. Given x_t, we add noise $x_{t+1}^s = e^{-s/2} \cdot x_{t+1}^0 + \sqrt{1-e^{-s}} \cdot z$ to the next frame following an OU process [2]:

$$dx_{t+1}^s = -\frac{1}{2}x_{t+1}^s ds + dW_s, \quad x_{t+1}^0 \sim p^{\text{ref}}(x_{t+1}|x_t)$$

then solving for the least squares objective as (Song et al., 2021):

$$\min_\theta \int_0^S \mathbb{E}_{x_t, x_{t+1}^s, x_{t+1}^0}\left[\|o_s^\theta(x_{t+1}^s, x_t) - \nabla_1 \log p_{s|0}(x_{t+1}^s|x_{t+1}^0)\|^2\right] ds$$

where

$$\nabla_1 \log p_{s|0}(x_{t+1}^s|x_{t+1}^0) = \frac{x_{t+1}^0 e^{-s/2} - x_{t+1}^s}{1-e^{-s}},$$

[2] Each trajectory frame x_t has a corresponding noised version x_t^s. Score models $o_s^\theta(x_{t+1}^s, x_t)$ are functions of the diffusion time s, evaluated at a noised frame, and conditioned on a clean frame x_t.

Algorithm 1 Nested SMC for Conditional Trajectory Generation

Require: First frame x_0, reward $r(\cdot)$, score o_s^θ, # particles K, inner samples M, horizon T

1: Set $x_0^k \leftarrow x_0$ for all $k \in [K]$; $\hat{Z}_T \leftarrow 1$
2: $\hat{\psi}_0(x_0) = 1$
3: **for** $t = 1, \ldots, T$ **do**
4: **for** $k = 1, \ldots, K$ **do** ▷ Inner IS sampler
5: **for** $j = 1, \ldots, M$ **do**
6: Sample $x_t^{k,j} \sim p_\theta^{\text{ref}}(\cdot \mid x_{0:t-1}^k)$ via SDE (5)
7: Compute Tweedie rollout $\hat{x}_{t+1:T}^{k,j}$ from $x_t^{k,j}$ using the scores o_s^θ :

$$\hat{x}_{t+1}(x_{t+1}^s; x_t) = e^{s/2}(x_{t+1}^s + (1 - e^{-s}) o_s^\theta(x_{t+1}^s, x_t)) \tag{6}$$

8: Compute the approximate twist:

$$v_t^{k,j} = \hat{\psi}_t(x_{0:t-1}^k, x_t^{k,j}) = r(x_{0:t-1}^k,\, x_t^{k,j},\, \hat{x}_{t+1:T}^{k,j}) \quad (\text{use } r(x_{0:T}^{k,j}) \text{ exactly if } t = T)$$

9: where $\hat{x}_{t+2:T}^{k,j}$ are generated recursively using (6)
10: **end for**
11: Resample a single particle: $x_t^k \sim \sum_{j=1}^M \frac{v_t^{k,j}}{\sum_l v_t^{k,l}} \delta_{x_t^{k,j}}$
12: Compute SMC weight $w_t^k \leftarrow \frac{1}{M} \sum_{j=1}^M v_t^{k,j} \,/\, \hat{\psi}_{t-1}(x_{0:t-1}^k)$
13: **end for**
14: Resample K trajectories $x_{0:t}^k$ from $\{[x_{0:t-1}^k, x_t^k]\}_k$ with weights $\frac{w_t^k}{\sum_l w_t^l}$
15: $\hat{Z}_T \leftarrow \hat{Z}_T \cdot \frac{1}{K} \sum_{k=1}^K w_t^k$
16: **end for**
17: **return** Final trajectories $\{x_{0:T}^k\}_{k=1}^K$; normalizing constant estimator $\hat{Z}_T$

and the marginal of this noising path is given by $p_s(x_{t+1}^s|x_t) = \int p_{s|0}(x_{t+1}^s|x_{t+1}^0) p(x_{t+1}^0|x_t)\, dx_{t+1}^0$ due to conditional independence. As long as samples from adjacent trajectory pairs $(x_{t+1}^0, x_t) \sim p(x_{t+1}^0, x_t)$ are available, this objective can be optimized and at sampling time with a trained score o_s^θ [3], we plug in the condition variable x_t from the previous step, and run the SDE

$$dZ_s = \frac{1}{2} Z_s ds + o_{S-s}^\theta(Z_s, x_t) ds + dW_s, \;\; Z_0 \sim \mathcal{N}(0, I) \tag{5}$$

to draw $Z_S \sim p^{\text{ref}}(x_{t+1}|x_t)$ approximately. In (5) the optimal $o_s^\theta(x_{t+1}^s, x_t) \approx \nabla_1 \log p_s(x_{t+1}^s|x_t)$. The whole process can be repeated sequentially to generate a sequence of length T, denoted as $x_{1:T}$.

Now, to wrap the auto-regressive generative model around the SMC framework, we propose the following Algorithm 1.

In Line 8 of Algorithm 1, we use the "posterior mean" output by the diffusion model for approximating the twists over the future states. The cheap twist $\hat{\psi}_t$ here is inspired by (Wu et al., 2023), where the states $x_{t+1:T}$ are approximated by *point estimators* using the conditional mean predictor as (z's are the Gaussian noise):

$$\hat{x}_{t+1}(x_t, z_{t+1}), \hat{x}_{t+2}(\hat{x}_{t+1}, z_{t+2}), \ldots$$

at a given time t as stand-in for $p^{\text{ref}}(x_{t+1:T}|x_{1:t})$ to avoid the integration. After Line 11 importance sampling, the x_t^k's are approximately distributed as $q_t^*(x_t|x_{0:t-1}^k) \propto p_\theta^{\text{ref}}(x_t|x_{0:t-1}^k)\hat{\psi}_t(x_{0:t-1}^k, x_t)$. The reason we use an IS sampler as subroutine is because (1) we cannot evaluate $p_\theta^{\text{ref}}(x_t|x_{0:t-1})$ from the diffusion model easily at an arbitrary point, which prevents an inner loop MCMC targeting q_t^*; (2) a Gaussian approximation as a proposal à la (Wu et al., 2023) is not fitting since $p_\theta^{\text{ref}}(x_t|x_{0:t-1})$ may not be well-approximated by a Gaussian (i.e., we are not considering a one-step diffusion from $s+1$ to s).

In Line 12, $\frac{1}{M} \sum_j v_t^{k,j}$ is an unbiased estimator of the normalizing constant of $q_t^*(x_t|x_{0:t-1}^k)$ as a function of $x_{0:t-1}^k$.

[3] The scores $o_s^\theta(x_{t+1}^s, x_t; \mathcal{G})$ implicitly depend on $\mathcal{G}$, the molecular graph / amino-acid sequence as well, but we leave this parameterization implicit for notation simplicity.

The incremental weights w_t^k come from

$$\frac{\pi_t(x_{0:t})}{\pi_{t-1}(x_{0:t-1})q_t^*(x_t|x_{0:t-1})}.$$

The proposed K trajectories are resampled according to the normalized weights at each $t \in [T]$ in Line 14 and the more promising ones are given higher weights early on based on partial sequences.

Lemma 1 (Correctness of the nested SMC targeting)**.** *Nested SMC of Algorithm 1 is an auxiliary SMC sampler on an extended state space, and the marginal over $x_{0:T}$ targets the posterior* (1)*.*

Proposition 1 (Unbiased estimator with finite M)**.** *Even with $M < \infty$, being an exact approximation, the normalizing constant estimator $\hat{Z}_T$ is unbiased for any $K \geq 1$. As the number of particles $K \to \infty$, the empirical discrete measure converges setwise to the desired posterior in* (1)*.*

The proofs of these results are given in Section A.2. At a high level, the twist $\hat{\psi}_t$ adjusts for the future reward, which as $t \to T$ converge to the true likelihood, where the base proposal traces out $Q^{\text{ref}}(x_{0:T})$ gradually, therefore the final distribution of the augmented chain has the desired posterior $Q(x_{0:T}|r)$. We'd like to note that unlike MDGen (Jing et al., 2024), this algorithm does not require labeled training data in the form of $(x_{0:T}, r(x_{0:T}))$ and can work with any $r(\cdot)$ at inference time given a pre-trained diffusion model without additional training. As written, no differentiability of the reward is required as well.

5. Take 2: look-ahead with cheap twists and pre-training for the marginal

The approach in Section 4, while theoretically sound, requires vast amount of paired adjacent frames (x_t, x_{t+1}) to learn the generative model for $p(x_{t+1}|x_t)$. As motivated in (Iyengar et al., 2025), static marginal structure data (from e.g., Protein Data Bank) is often more abundant than MD trajectory data so we aim to leverage structure pretraining, followed by temporal interpolation fine-tuning, for learning and simulating the time-series trajectory in this section.

Given samples from the marginal $p(x_t)$ with the stationary Boltzmann distribution, we aim to combine 2 pre-trained diffusion models to follow a prescribed marginal curve. The motivation is that this additional static data (and the corresponding diffusion model) can be composed with the one from MD to build better proposals in SMC. To do this, we modify the denoising process in (5) (which traces $p_s(x_{t+1}^s|x_t)$) so for a fixed t, the model follows from $s = S, \ldots, 0$ the new path

$$\bar{p}_{t+1}^s(x_{t+1}^s|x_t) \propto p_0(x_{t+1}^s)\frac{p_s(x_{t+1}^s|x_t)}{p_s(x_{t+1}^s)} \to p^{\text{ref}}(x_{t+1}|x_t) \quad \text{as } s \to 0 \tag{7}$$

while keeping the sequence of intermediate targets as $\pi_{t+1}(x_{1:t+1}) = p^{\text{ref}}(x_{1:t+1}) \cdot \psi_{t+1}(x_{1:t+1})$ as before with the outer weights w_t^k. Note that the extra ratio in (7) is ≈ 1 if the adjacent frames are almost independent, in which case the proposal $x_t^{k,j} \sim p_\theta^{\text{ref}}(\cdot|x_{0:t-1}^k)$ from Line 6 of Algorithm 1 is expected to greatly benefit from these additional marginal data.

Based on the formulation above one can leverage (Skreta et al., 2025; Du et al., 2023) for composing pre-trained diffusion models at inference time without additional training. With scores

$$o_s^\theta(x_{t+1}^s, x_t) \approx \nabla_1 \log p_s(x_{t+1}^s|x_t),$$
$$j_s^\theta(x_{t+1}^s) \approx \nabla \log p_s(x_{t+1}^s)$$

on hand (but not an explicit model of the log-likelihood), we can construct the proposal $x_t^{k,j} \sim p_\theta^{\text{ref}}(\cdot|x_{0:t-1}^k), j \in [M]$ [4] as given in Lemma 2 (proof in Appendix A.3). The pseudo-code that replaces the **inner IS sampler** from Algorithm 1 is given in Algorithm 2.

Lemma 2 (Feynman-Kac Reweighting for Composition)**.** *Consider the composition target from* (7)*. Decompose it as $\bar{p}_{t+1}^s(x|x_t) \propto p_0(x) \cdot h_s(x)$ where $h_s(x) := p_s(x|x_t)/p_s(x)$ is the "guidance" and $p_0(x)$ the static base measure. Run the reverse SDE from time $\tau = 0$ (noise level $s = S$) to $\tau = S$ (noise level $s = 0$):*

$$dX_\tau = \left[o_{S-\tau}^\theta(X_\tau, x_t) - j_{S-\tau}^\theta(X_\tau) + \tfrac{1}{2}j_0^\theta(X_\tau) + \tfrac{1}{2}X_\tau\right]d\tau + dW_\tau, \quad X_0 \sim \bar{p}_{t+1}^S(x_{t+1}^S|x_t), \tag{8}$$

[4] We keep this proposal choice for the inner IS sampler since it leads to simple weights and clean variance analysis. We discuss alternative (controlled inner proposals) in Appendix D.

Algorithm 2 One-Step Proposal $x^k_{0:t-1} \to x^k_t$ via FK Composition SDE + Nested SMC

Require: Prefix $x^k_{0:t-1}$ (conditioning frame x^k_{t-1}); scores j^θ_s, o^θ_s; twist $\hat{\psi}_t$; inner samples M

1: **Stage 1: Generate approximate $p^{\mathrm{ref}}(\cdot \mid x_{t-1})$ samples via composition SDE**
2: **for** $j = 1, \dots, M$ **do**
3: Sample $X^j_0 \sim \mathcal{N}(0, I)$
4: Run composition reverse SDE (8) from $\tau = 0$ to $\tau = S$ with drift $b_\tau(x) = o^\theta_{S-\tau}(x, x^k_{t-1}) - j^\theta_{S-\tau}(x) + \frac{1}{2} j^\theta_0(x) + \frac{1}{2} x$
5: Accumulate FKC log-weight along the path (Lemma 2):

$$\log w^j = \int_0^S \left[\tfrac{d}{2} + \langle j^\theta_{S-\tau},\ j^\theta_{S-\tau} - o^\theta_{S-\tau} \rangle + \tfrac{1}{2} \langle j^\theta_0,\ X^j_\tau + o^\theta_{S-\tau} - j^\theta_{S-\tau} \rangle \right] d\tau$$

6: **end for**
7: Resample M samples $\{\tilde{x}^j_t\}_{j=1}^M$ from $\{X^j_S\}_{j=1}^M$ with self-normalized weights $\bar{w}^j = w^j / \sum_l w^l$
8: **Stage 2: Twist-based inner IS (as in Algorithm 1)**
9: **for** $j = 1, \dots, M$ **do**
10: Compute twist weight: $v^{k,j}_t \leftarrow \hat{\psi}_t(x^k_{0:t-1}, \tilde{x}^j_t)$
11: **end for**
12: Select a single sample $x^k_t \sim \sum_j \frac{v^{k,j}_t}{\sum_l v^{k,l}_t} \delta_{\tilde{x}^j_t}$
13: Outer SMC weight (for this particular k): $w^k_t \leftarrow \frac{1}{M} \sum_{j=1}^M v^{k,j}_t \ / \ \hat{\psi}_{t-1}(x^k_{0:t-1})$

while accumulating the deterministic Feynman-Kac log-weight along the trajectory

$$d \log w_\tau = \left[\tfrac{d}{2} + \langle j^\theta_{S-\tau},\ j^\theta_{S-\tau} - o^\theta_{S-\tau} \rangle + \tfrac{1}{2} \langle j^\theta_0,\ X_\tau + o^\theta_{S-\tau} - j^\theta_{S-\tau} \rangle \right] d\tau \,. \tag{9}$$

Resampling from the final weighted particles $(X^m_S, w^m_S)_{m=1}^M$ with self-normalized weights correctly targets the proposal $p^{\mathrm{ref}}(x_{t+1}|x_t)$ for any fixed given x_t.

Note that Algorithm 2 requires only pointwise score evaluations along the sample path and has a two-stage structure. *Stage 1* runs the composition SDE (8), whose drift combines the h function scores $o^\theta_s - j^\theta_s$ with a half-strength data score $\frac{1}{2} j^\theta_0$. The FKC weight (9) consists of two terms: $\langle j_s, j_s - o_s \rangle$ corrects for the guidance ratio $h_s = p_s(\cdot|x_t)/p_s$, while $\frac{1}{2} \langle j^\theta_0, x + o_s - j_s \rangle$ incorporates the static base measure p_0. Crucially, adding $\frac{1}{2} j^\theta_0$ to the drift (rather than the full j^θ_0) cancels the Laplacian $\nabla \cdot j^\theta_0$ that would otherwise appear in the weight, so that the integrand involves only inner products of the scores. Resampling with self-normalized weights yields approximate samples $\tilde{x}^j_t \sim p^{\mathrm{ref}}(\cdot|x_{t-1})$. *Stage 2* then proceeds exactly as Algorithm 1: the twist $\hat{\psi}_t$ is evaluated on each resampled proposal, inner resampling selects x^k_t proportionally to $v^{k,j}_t$, and the outer SMC weights $\{w^k_t\}_k$ are used to resample the K partial trajectories $\{x^k_{0:t}\}_k$ at step t. This decomposition makes explicit that Take 2 uses the composition SDE as a *sampler* for p^{ref}, followed by the same twist-based nested SMC as Algorithm 1.

An alternative *single-stage* formulation combines the FKC and twist weights into a single IS: resample proportionally to $w^j v^{k,j}_t$ and set the outer weight to $\frac{1}{M} \sum_j w^j v^{k,j}_t / \hat{\psi}_{t-1}$. This preserves the unbiasedness of $\hat{Z}_T$ (Proposition 1), since it is another invocation of the nested SMC framework. However, when FKC weights have high variance, the product $w^j v^{k,j}_t$ is dominated by the FKC term, effectively nullifying the twist's guidance.

The *two-stage* version in Algorithm 2 introduces $O(1/M)$ bias from the intermediate self-normalized FKC resampling in Line 7 (since it produces samples from the empirical measure $\hat{p}^{\mathrm{ref}}_M = \sum_j \bar{w}^j \delta_{X^j_S}$ rather than from p^{ref} itself). Self-normalized importance sampling is consistent but biased for finite M. Consequently, $\hat{Z}_T$ under Algorithm 2 is biased with $|\mathbb{E}[\hat{Z}_T] - Z_T| = O(1/M)$, but remains consistent: $\hat{Z}_T \xrightarrow{P} Z_T$ as $K \to \infty$, $M \to \infty$. The CLT continues to hold (with a modified asymptotic variance from Algorithm 1; see Appendix B). In practice, the $O(1/M)$ bias is negligible compared to the variance introduced by the single-stage alternative (c.f., Section 7.1 for a detailed comparison).

6. Take 3: learn the optimal twists for look-ahead using data

The variance of the importance weights and the quality of the samples depend on the twist used. Motivated by the fact that the variance from the noisy rollout twist as used in Algorithm 1 can be high, below we propose to set up least squares objective in 2 ways (based on recursion properties) to solve for the optimal twist function. The new learned twists can be plugged into the same SMC framework earlier for following the *optimal sequence of intermediate targets* with good proposals. There are connections of methods on learning ψ^* to reinforcement learning-based fine-tuning (Uehara et al., 2025).

For learning the optimal twists, we build on the recursion from (Zhao et al., 2024, Equation 12,13), and (Uehara et al., 2025, Algorithm 2, 3). Collecting K sampled trajectories $x_{0:T}^{(k)} \sim Q^{\text{ref}}$ from the unconditioned reference,

1. Solve MC objective:
$$\{\psi_t^*\}_t \leftarrow \arg\min_{\psi} \sum_{t=0}^{T} \sum_{k=1}^{K} \left(r(x_{0:T}^{(k)}) - \psi_t(x_{0:t}^{(k)})\right)^2 \tag{10}$$
2. Or repeatedly solve the TD objective:
$$\{\psi_t^j\}_t \leftarrow \arg\min_{\psi} \sum_{t=0}^{T-1} \sum_{k=1}^{K} \left(\psi_t(x_{0:t}^{(k)}) - \psi_{t+1}^{j-1}(x_{0:t+1}^{(k)})\right)^2 \tag{11}$$

The validity of these objectives is established in Lemma 5 and is based on recasting conditional expectation as least squares problem and (3). Twist learning is also used in (Phillips et al., 2024) in the context of sampling.

Without samples from target marginals $Q(x_{1:t}|r)$, contrastive learning (i.e., marginal-matching) based technique from (Zhao et al., 2024, Section 4.1) is not implementable without additional importance weighting. With samples from Q^{ref} only, we can adapt to give a variant as follows, which becomes amenable to iterative stochastic gradient updates for optimization. Proofs can be found in Appendix A.4.

Lemma 3 (Gradient Based KL Twist Learning)**.** *Given objective*

$$\arg\min_{\theta} \sum_{t=1}^{T} KL(\sigma(x_{0:t}) \| \mathbb{Q}_t^{\theta}(x_{0:t})) = \arg\min_{\theta} \sum_{t=1}^{T} KL\left(\frac{1}{Z_t^{\psi^*}} p^{ref}(x_{0:t}) \psi_t^*(x_{0:t}) \| \frac{1}{Z_t^{\psi^\theta}} p^{ref}(x_{0:t}) \psi_t^{\theta}(x_{0:t})\right),$$

for each t, we have that the stochastic gradient can be estimated as

$$\begin{aligned} &\mathbb{E}_{x_{0:t} \sim \mathbb{Q}_t^{\theta}(x_{0:t})}[\nabla_\theta \log \psi_t^{\theta}(x_{0:t})] \times \mathbb{E}_{x_{0:T} \sim p^{ref}}[r(x_{0:T})] \\ &\quad - \mathbb{E}_{x_{0:T} \sim p^{ref}}[\nabla_\theta \log \psi_t^{\theta}(x_{0:t}) \times r(x_{0:T})] \end{aligned}$$

up to constants independent of θ. This is estimable using samples from p^{ref}, and current $\mathbb{Q}_t^{\theta}$ alone.

We presented 3 methods in this paper, in increasing computational complexity. Both Take 1 and Take 3 are unbiased, while Take 2 with two-stage formulation pays a small bias. Take 3 involves an amortized twist-learning step, while Take 2 pays a one-time cost of learning the marginal score. We give a quantitative analysis on the asymptotic variance of estimators below to make precise the statistical benefit one can expect in return. The proof of this result, along with additional discussion can be found in Appendix B.

Theorem (Informal: Asymptotic variance of nested SMC – precise version given in Section B.1)**.** *With multinomial resampling, $\sqrt{K}(\hat{Z}_T^K / Z_T - 1) \xrightarrow{d} \mathcal{N}(0, \sigma^2)$ with asymptotic variance*

$$\sigma^2 = \sum_{t=1}^{T} \left[\underbrace{R_t}_{\text{resampling}} + \underbrace{\frac{1}{M} D_t^{\hat{\psi}}}_{\text{inner IS noise}} \right], \tag{12}$$

where for $h_t(x_{0:t-1}) := \mathbb{Q}_t(x_{0:t-1}) / \mathbb{Q}_{t-1}(x_{0:t-1})$,

$$R_t := \chi^2\left(\mathbb{Q}_t(x_{0:t-1}) \,\middle\|\, \mathbb{Q}_{t-1}(x_{0:t-1})\right) = \text{Var}_{\mathbb{Q}_{t-1}}(h_t), \tag{13}$$

Table 1. 6D coupled double-well. Learned-proposal GT: $\log Z = -4.858$. True-dynamics GT: $\log Z = -5.161$. The bias columns are on the Z scale relative to the learned-proposal target.

M	Single-stage $\log \hat{Z}$	Two-stage $\log \hat{Z}$	Single-stage Z-bias	Two-stage Z-bias
5	-5.266 ± 1.740	-5.497 ± 0.248	$+0.010$	-0.004
10	-5.325 ± 1.198	-5.179 ± 0.203	$+0.002$	-0.002
20	-4.668 ± 1.090	-4.945 ± 0.158	$+0.008$	-0.001

and

$$D_t^{\hat{\psi}} := \mathbb{E}_{\mathbb{Q}_{t-1}}\big[h_t(x_{0:t-1})^2 \cdot \chi^2\big(q_t^*(\mathrm{d}x_t|x_{0:t-1}) \,\big\|\, p^{\mathrm{ref}}(\mathrm{d}x_t|x_{0:t-1})\big)\big]. \tag{14}$$

The decomposition (12) reveals the two-level structure of the nested SMC:

(a) *Resampling error* R_t measures the mismatch between successive twisted filtering distributions at the $x_{1:t-1}$-marginal. Under the optimal twist sequence $\hat{\psi}_t \propto \psi_t^*$, these marginals coincide for every t, so $R_t = 0$. This term is independent of the number of inner samples M.

(b) *Inner IS noise* $D_t^{\hat{\psi}}/M$ measures the difficulty of sampling from q_t^* using p^{ref} as the IS proposal, attenuated by the number of inner samples M. The factor $\chi^2(q_t^* \| p^{\mathrm{ref}})$ is large when the twist $\hat{\psi}_t$ significantly reshapes the prior transition, requiring many inner samples to approximate the optimal proposal well.

7. Experiments

We focus here on 3 experiments that most directly isolate the paper's central claims: the two-stage Take 2 decomposition in a high-dimensional synthetic system (Sec 7.1), rare-event conditioning on alanine dipeptide (Sec 7.2), and a path-dependent benchmark in which the optimal twist is not Markov in x_t alone (Sec 7.3).[5] Below *Bootstrap* is the untwisted SMC baseline: it propagates particles with the learned reference proposal with no learned twist or future look-ahead; particles are corrected only by the available reward. Appendix C contains additional experiments and ablations.

7.1. Single-stage vs. two-stage Take 2

This experiment isolates the effect of the two-stage decomposition in Take 2. We use a 6D coupled double-well with $T = 10$, $K = 100$, $M \in \{5, 10, 20\}$, with $r(x_{1:T}) = \exp\{-0.35\|x_T - \mathbf{1}\|_2^2\}$, so the task is to sample trajectories ending near the opposite well. All scores are learned, the inner proposal is the learned composition SDE, and we compare against the learned-proposal ground truth $\int p_\theta^{\mathrm{ref}}(x_{1:T}) \times r(x_{1:T}) dx_{1:T}$ while also reporting the true-dynamics reference in the caption.

The result in Table 1 provides the clearest evidence for the two-stage decomposition. Across all three M values, the two-stage estimator nearly eliminates the learned-target Z-bias while sharply reducing variability. This provides support that separating the noisy Stage-1 FKC correction from the outer twist weight, as we do in Algorithm 2, yields better performance.

7.2. Rare-event: Alanine dipeptide

We train a system-specific proposal model on alanine dipeptide backbone torsion trajectories and evaluate both forward simulation and a midpoint + endpoint conditioning task from the C7eq basin to the α_R basin. The final reported task contains 14 held-out transition windows from the test split. We compare against both the empirical test-window target and the learned-proposal target.

The learned proposal is already a reasonable forward model, with Ramachandran JSD 0.444 and TICA free-energy-surface JSD 0.360. On the conditioning task, naive diffusion without any conditioning essentially fails, while Bootstrap and Take 3 (10) both recover nontrivial conditioned transitions (c.f. Table 2).

[5] A reproducible repository can be found at the following link: https://anonymous.4open.science/r/SMC_for_MD-03AC/README.md

Table 2. Midpoint + endpoint conditioning on alanine dipeptide. Empirical test-window GT: $\log Z = -4.331$. Learned-proposal GT: $\log Z = -6.272$.

Method	$\log \hat{Z}$	Joint succ.	Midpoint succ.	Ramachandran JSD	TICA JSD
Naive diffusion	–	0.004	0.061	0.821	0.730
Bootstrap	-6.490 ± 0.490	0.705	0.784	0.709	0.528
Take 3	-5.613 ± 0.647	0.726	0.858	0.667	0.502

Table 3. Non-Markovian route-history benchmark. The reward depends on route history rather than on x_t alone. Take 3 (KL objective from Lemma 3) is the only method that is simultaneously close on $\log Z$, route split, and visitation JSD.

Method	$\log \hat{Z}$	Mean ESS	Upper / Lower	Visitation JSD
Exact-proposal GT: $\log Z = -3.582$*, upper/lower/undecided* = $0.749/0.114/0.137$.				
Bootstrap	-4.230 ± 0.934	62.27 ± 0.02	$0.545/0.223$	0.566 ± 0.044
Terminal-only IS	-4.472 ± 1.107	2.07 ± 0.85	$0.541/0.051$	0.579 ± 0.032
Take 3 (KL)	-3.760 ± 0.207	59.51 ± 0.44	$0.670/0.170$	0.356 ± 0.028

Relative to the empirical test-window target, Take 3 reduces the $\log \hat{Z}$ error from 2.159 to 1.282, improves joint success from 0.705 to 0.726, increases midpoint success from 0.784 to 0.858, and modestly improves both geometry metrics. The improvement is therefore on the full conditioned transition rather than on terminal occupancy alone: endpoint-only success remains slightly higher for Bootstrap (0.891 versus 0.852). Relative to the learned-proposal target, Bootstrap is closer, underscoring that proposal quality remains important.

7.3. Path-dependent benchmark

We experiment with the 2D coupled double-well, using $T = 36$ and a reward that depends on *which gate is hit first* on the way to the right well. This makes the target non-Markovian in x_t alone, although it becomes Markov after augmenting the state with a route-progress flag. We therefore learn Take 3 twists on an augmented state and report with the exact-drift backend (i.e., p^{ref} simulated with exact drift) so that the comparison isolates path dependence rather than proposal mismatch. Appendix C.3 gives the full setup, the twist-objective ablation, and the variance-decomposition plot, confirming that Take 3 trades the bootstrap-style terminal variance spike for moderate earlier-step contributions.

This benchmark separates the methods clearly (c.f. Figure 1 and Table 3). Terminal-only importance sampling collapses, with ESS only 2.07 and severe under-coverage of the lower route. Bootstrap is much more stable, but it still misses the target route split and occupancy pattern. Take 3 (KL) is the strongest overall method: it is closest to the target on $\log \hat{Z}$, has the best visitation JSD, and tracks the upper-versus-lower passage rates substantially more faithfully than the alternatives.

8. Conclusion

In this paper, we study conditional molecular-dynamics trajectory generation as an inference-time path-space problem, motivated by the fact that many molecular questions (e.g., predicting drug residence time) depend on kinetics and transition pathways rather than on static equilibrium structures alone. Our main contribution is a twisted-SMC framework that cleanly separates learning a reference dynamical model from imposing new trajectory-level rewards at test time. This yields a hierarchy of methods spanning cheap look-ahead heuristics, composition-SDE proposals, and learned twists, together with asymptotic correctness guarantees and an asymptotic variance decomposition that clarifies when each component matters. Empirically, the results show that history-aware twisting can materially improve conditional path generation and rare-event calibration, while the two-stage decomposition is important for making composition-based proposals reliable in practice.

Acknowledgements

We thank Saeed Saremi for several helpful discussions related to the topic.

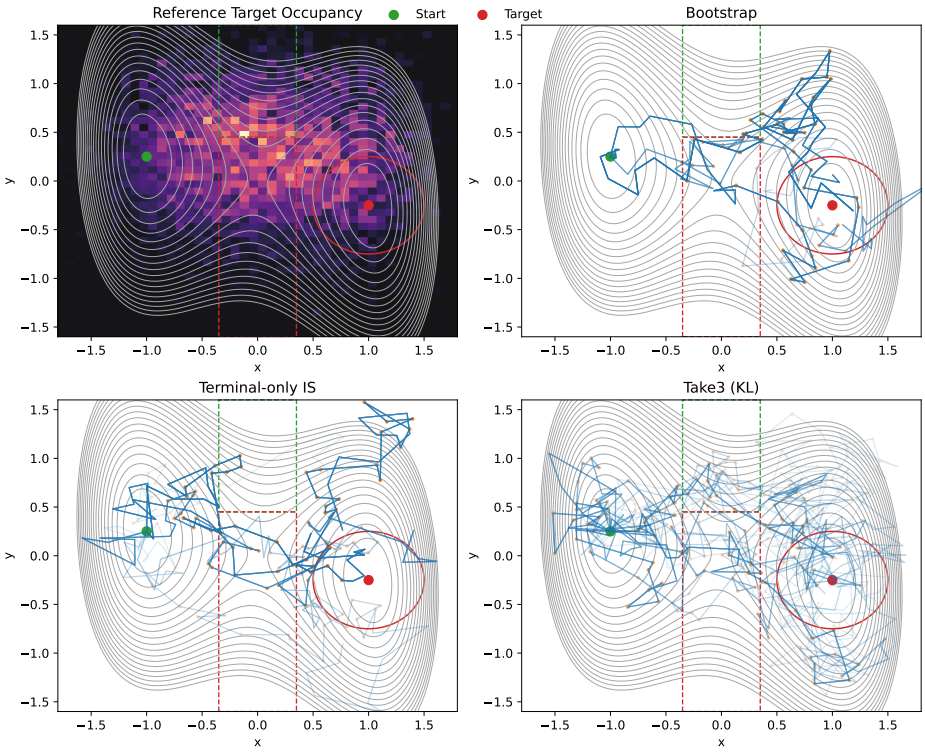

Figure 1. Occupancy plot. Top-left: reward-weighted reference occupancy over intermediate trajectory frames. The dashed boxes mark the upper and lower gates, and the red circle marks the endpoint region. Terminal-only IS visibly under-covers the target distribution, while Take 3 (KL) tracks both the route geometry and the final-state concentration more faithfully than Bootstrap.

References

Bigi, F., Chong, S., Kristiadi, A., and Ceriotti, M. FlashMD: long-stride, universal prediction of molecular dynamics. *arXiv preprint arXiv:2505.19350*, 2025.

Chen, Junhua and Richter, Lorenz and Berner, Julius and Blessing, Denis and Neumann, Gerhard and Anandkumar, Anima. Sequential controlled langevin diffusions. *arXiv preprint arXiv:2412.07081*, 2024.

Chopin, N. Central limit theorem for sequential monte carlo methods and its application to bayesian inference. *Annals of Statistics*, 32(6):2385–2411, December 2004. ISSN 0090-5364. doi: 10.1214/009053604000000698.

Chopin, N. and Papaspiliopoulos, O. *An Introduction to Sequential Monte Carlo*. Springer Series in Statistics. Springer, 2020.

Domingo-Enrich, C., Drozdzal, M., Karrer, B., and Chen, R. T. Adjoint Matching: Fine-tuning Flow and Diffusion Generative Models with Memoryless Stochastic Optimal Control. In *The Thirteenth International Conference on Learning Representations*, 2025.

Du, Y., Durkan, C., Strudel, R., Tenenbaum, J. B., Dieleman, S., Fergus, R., Sohl-Dickstein, J., Doucet, A., and Grathwohl, W. S. Reduce, reuse, recycle: Compositional generation with energy-based diffusion models and MCMC. In *International conference on machine learning*, pp. 8489–8510. PMLR, 2023.

Havens, A., Miller, B. K., Yan, B., Domingo-Enrich, C., Sriram, A., Wood, B., Levine, D., Hu, B., Amos, B., Karrer, B., et al. Adjoint sampling: Highly scalable diffusion samplers via adjoint matching. *arXiv preprint arXiv:2504.11713*, 2025.

Iyengar, A., Han, J., Sun, P., Jiang, M., Xie, J., and Ermon, S. Align Your Structures: Generating Trajectories with Structure Pretraining for Molecular Dynamics. In *ICML 2025 Generative AI and Biology (GenBio) Workshop*, 2025.

Jing, B., Stärk, H., Jaakkola, T., and Berger, B. Generative modeling of molecular dynamics trajectories. *Advances in Neural Information Processing Systems*, 37:40534–40564, 2024.

Jing, B., Berger, B., and Jaakkola, T. AI-based Methods for Simulating, Sampling, and Predicting Protein Ensembles. *arXiv preprint arXiv:2509.17224*, 2025.

Klein, L., Foong, A., Fjelde, T., Mlodozeniec, B., Brockschmidt, M., Nowozin, S., Noé, F., and Tomioka, R. Timewarp: Transferable acceleration of molecular dynamics by learning time-coarsened dynamics. *Advances in Neural Information Processing Systems*, 36:52863–52883, 2023.

Levine, S. Reinforcement learning and control as probabilistic inference: Tutorial and review. *arXiv preprint arXiv:1805.00909*, 2018.

Naesseth, C., Lindsten, F., and Schon, T. Nested sequential monte carlo methods. In *International Conference on Machine Learning*, pp. 1292–1301. PMLR, 2015.

Naesseth, C. A., Lindsten, F., Schön, T. B., et al. Elements of sequential monte carlo. *Foundations and Trends® in Machine Learning*, 12(3):307–392, 2019.

Noé, F., Olsson, S., Köhler, J., and Wu, H. Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. *Science*, 365(6457):eaaw1147, 2019.

Phillips, A., Dau, H.-D., Hutchinson, M. J., De Bortoli, V., Deligiannidis, G., and Doucet, A. Particle Denoising Diffusion Sampler. In *International Conference on Machine Learning*, pp. 40688–40724. PMLR, 2024.

Schreiner, M., Winther, O., and Olsson, S. Implicit transfer operator learning: Multiple time-resolution models for molecular dynamics. *Advances in Neural Information Processing Systems*, 36:36449–36462, 2023.

Skreta, M., Akhound-Sadegh, T., Ohanesian, V., Bondesan, R., Aspuru-Guzik, A., Doucet, A., Brekelmans, R., Tong, A., and Neklyudov, K. Feynman-Kac Correctors in Diffusion: Annealing, Guidance, and Product of Experts. In *Forty-second International Conference on Machine Learning*, 2025.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-Based Generative Modeling through Stochastic Differential Equations. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=PxTIG12RRHS.

Uehara, M., Zhao, Y., Wang, C., Li, X., Regev, A., Levine, S., and Biancalani, T. Inference-time alignment in diffusion models with reward-guided generation: Tutorial and review. *arXiv preprint arXiv:2501.09685*, 2025.

Wu, L., Trippe, B., Naesseth, C., Blei, D., and Cunningham, J. P. Practical and asymptotically exact conditional sampling in diffusion models. *Advances in Neural Information Processing Systems*, 36:31372–31403, 2023.

Wu, L., Han, Y., Naesseth, C. A., and Cunningham, J. P. Reverse diffusion sequential Monte Carlo samplers. *arXiv preprint arXiv:2508.05926*, 2025.

Zhao, S., Brekelmans, R., Makhzani, A., and Grosse, R. Probabilistic inference in language models via twisted sequential Monte Carlo. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 60704–60748, 2024.

The appendix is organized as follows:

- In Appendix A, we provide omitted proof details from Section 3 – Section 6.
- In Appendix B, we conduct a thorough variance analysis, comparing the statistical performance on estimators from Take 1-3.
- In Appendix C, we provide additional experimental results and ablations.
- In Appendix D, we discuss an extension using controlled inner proposal and draw connection to other controlled-based approaches in the literature.

A. Proof Details

A.1. Omitted Proofs from Section 3

Lemma 4 (Optimality of the locally optimal proposal)**.** *Consider the SMC framework with unnormalized intermediate targets $\pi_t(x_{1:t}) = p^{\mathrm{ref}}(x_{1:t} \mid x_0)\hat{\psi}_t(x_{1:t})$ for twist functions $\hat{\psi}_t$ satisfying $\hat{\psi}_T = r$, and denote $Z_t^{\hat{\psi}} := \int p^{\mathrm{ref}}(x_{1:t} \mid x_0)\hat{\psi}_t(x_{1:t})\, dx_{1:t}$.*

(i) ***(Zero conditional variance.)*** *For any twist $\hat{\psi}_t$, the one-step optimal proposal*

$$q_t^*(x_t \mid x_{1:t-1}) = \frac{p_\theta^{\mathrm{ref}}(x_t \mid x_{1:t-1})\hat{\psi}_t(x_{1:t-1}, x_t)}{\int p_\theta^{\mathrm{ref}}(x_t' \mid x_{1:t-1})\hat{\psi}_t(x_{1:t-1}, x_t')\, dx_t'}$$

yields the incremental importance weight

$$w_t(x_{1:t-1}) = \frac{\int p_\theta^{\mathrm{ref}}(x_t \mid x_{1:t-1})\hat{\psi}_t(x_{1:t-1}, x_t)\, dx_t}{\hat{\psi}_{t-1}(x_{1:t-1})}$$

which is independent of the proposed sample x_t, achieving $\mathrm{Var}(w_t \mid x_{1:t-1}) = 0$. No other proposal can attain lower conditional variance.

(ii) ***(Constant weights under optimal twist.)*** *When $\hat{\psi}_t = \psi_t^*$ (the optimal twist from (3)), the incremental weights are constant across particles: $w_1 \equiv Z(x_0)$ and $w_t \equiv 1$ for $t = 2, \ldots, T$. Consequently, the normalizing constant estimator satisfies $\hat{Z}_T = Z(x_0)$ almost surely and has zero variance.*

Proof. **Part (i).** The standard SMC incremental weight at step t is

$$w_t(x_{1:t}) = \frac{\pi_t(x_{1:t})}{\pi_{t-1}(x_{1:t-1})\, q_t(x_t \mid x_{1:t-1})}.$$

Substituting $\pi_t = p^{\mathrm{ref}}(x_{1:t} \mid x_0)\hat{\psi}_t(x_{1:t})$, $\pi_{t-1} = p^{\mathrm{ref}}(x_{1:t-1} \mid x_0)\hat{\psi}_{t-1}(x_{1:t-1})$, and the optimal proposal q_t^*:

$$\begin{aligned} w_t &= \frac{p^{\mathrm{ref}}(x_t \mid x_{1:t-1})\hat{\psi}_t(x_{1:t})}{\hat{\psi}_{t-1}(x_{1:t-1})} \cdot \frac{\int p_\theta^{\mathrm{ref}}(x_t' \mid x_{1:t-1})\hat{\psi}_t(x_{1:t-1}, x_t')\, dx_t'}{p_\theta^{\mathrm{ref}}(x_t \mid x_{1:t-1})\hat{\psi}_t(x_{1:t})} \\ &= \frac{\int p_\theta^{\mathrm{ref}}(x_t \mid x_{1:t-1})\hat{\psi}_t(x_{1:t-1}, x_t)\, dx_t}{\hat{\psi}_{t-1}(x_{1:t-1})} \end{aligned}$$

where in the 1st equality we used the Markov property. The resulting expression depends only on $x_{1:t-1}$, so the weight is deterministic given the history, giving $\mathrm{Var}(w_t \mid x_{1:t-1}) = 0$. Since the conditional variance is non-negative for any proposal, the optimal proposal achieves the global minimum.

Part (ii). When $\hat{\psi}_t = \psi_t^*$, the numerator in part (i) becomes

$$\int p^{\mathrm{ref}}(x_t \mid x_{1:t-1})\psi_t^*(x_{1:t-1}, x_t)\, dx_t = \int p^{\mathrm{ref}}(x_t \mid x_{1:t-1}) \left[\int p^{\mathrm{ref}}(x_{t+1:T} \mid x_{1:t}) r(x_{1:T})\, dx_{t+1:T} \right] dx_t$$

$$= \int p^{\mathrm{ref}}(x_{t:T} \mid x_{1:t-1})\, r(x_{1:T})\, dx_{t:T} \;=\; \psi^*_{t-1}(x_{1:t-1})$$

where we used the chain rule and the definition of ψ^*_{t-1} from (3). Therefore the weights $w_t = \psi^*_{t-1}(x_{1:t-1})/\psi^*_{t-1}(x_{1:t-1}) = 1$ for $t \geq 2$.

For the normalizing constants:

$$\begin{aligned} Z_t^{\psi^*} &= \int p^{\mathrm{ref}}(x_{1:t} \mid x_0)\psi^*_t(x_{1:t})\, dx_{1:t} \\ &= \int p^{\mathrm{ref}}(x_{1:t} \mid x_0) \left[\int p^{\mathrm{ref}}(x_{t+1:T} \mid x_{1:t}) r(x_{1:T})\, dx_{t+1:T}\right] dx_{1:t} \\ &= \int p^{\mathrm{ref}}(x_{1:T} \mid x_0)\, r(x_{1:T})\, dx_{1:T} = Z(x_0) \end{aligned}$$

for every t. Hence $Z_t^{\psi^*}/Z_{t-1}^{\psi^*} = 1$, and $\hat{Z}_T = \prod_{t=1}^T \frac{1}{K}\sum_{k=1}^K w_t^k = Z(x_0)$ almost surely. □

Remark. For a general approximate twist $\hat{\psi}_t$ that does not satisfy the backward recursion $\hat{\psi}_{t-1}(x_{1:t-1}) \propto \int p^{\mathrm{ref}}(x_t \mid x_{1:t-1})\hat{\psi}_t(x_{1:t-1}, x_t)\, dx_t$, the weight from part (i) will still depend on $x_{1:t-1}$ (though not on x_t). In practice, the Tweedie-based twist used in Algorithm 1 only approximates the optimal twist (3), so the weights will not be exactly constant, but the optimal proposal still provides variance reduction over any alternative proposal.

A.2. Omitted Proofs from Section 4

Lemma 1 and Proposition 1 are the main results establishing the correctness of our nested SMC framework.

Proof of Lemma 1. We can show that the proposed algorithm is nothing but a standard SMC algorithm for the sequence of *un-normalized* extended target distribution defined recursively as:

$$\bar{\Pi}_t(x_{0:t}, u_{1:t}) := \frac{\tau_t(u_t)}{\hat{\psi}_{t-1}(x_{0:t-1})}\, \phi_t^M(u_t|x_{0:t-1})\, \gamma_t^M(x_t|u_t) \cdot \bar{\Pi}_{t-1}(x_{0:t-1}, u_{1:t-1})$$

where given any arbitrary $x^k_{0:t-1}$, we define

$$\bar{\Pi}_0(x_0) = \delta(x_0) = \pi_0(x_0)$$

$$\tau_t(u_t) = \frac{1}{M}\sum_{j=1}^M \hat{\psi}_t(x^k_{0:t-1}, u_t^{k,j}) \to \text{normalizing constant estimator } Z_{q_t}(x^k_{0:t-1}) \text{ for } \bar{q}^*_t(x_t|x^k_{0:t-1})$$

and the proposal distributions for (u_t, x_t) given $x^k_{0:t-1}$:

$$\phi_t^M(u_t|x_{0:t-1}) = \prod_{j=1}^M p^{\mathrm{ref}}_\theta(u_t^{k,j}|x^k_{0:t-1}) \to \text{normalized proposal for } u_t^{k,j}$$

$$\gamma_t^M(x_t|u_t) = \sum_{j=1}^M \frac{\hat{\psi}_t(x^k_{0:t-1}, u_t^{k,j})}{\sum_l \hat{\psi}_t(x^k_{0:t-1}, u_t^{k,l})}\delta(x_t^k = u_t^{k,j}) \to \text{normalized proposal for } x_t^k$$

From this, each of the intermediate weights can be calculated as

$$\bar{w}_t(x_{0:t-1}, x_t) = \frac{\bar{\Pi}_t(x_{0:t}, u_{1:t})}{\bar{\Pi}_{t-1}(x_{0:t-1}, u_{1:t-1})\, \phi_t^M(u_t|x_{0:t-1})\, \gamma_t^M(x_t|u_t)} = \frac{\tau_t(u_t)}{\hat{\psi}_{t-1}(x_{0:t-1})} \tag{15}$$

which agrees with our definition of w_t^k from Algorithm 1 used for resampling. Above we used the $u_t^{k,j}$'s to denote the M auxiliary samples used in proposing x_t^k from $x^k_{0:t-1}$ for each $k \in [K]$.

Now it remains to check that marginalizing out the auxiliary variables $\int \bar{\Pi}_t(x_{0:t}, u_{1:t})\, du_{1:t} = \pi_t(x_{0:t}) = p_\theta^{\text{ref}}(x_{0:t})\hat{\psi}_t(x_{0:t})$ for all $t \in [T]$, since then it implies the target at time T is the desired $\int \bar{\Pi}_T(x_{0:T}, u_{1:T})\, du_{1:T} = p_\theta^{\text{ref}}(x_{0:T})r(x_{0:T})$ as $\hat{\psi}_T(x_{0:T}) = r(x_{0:T})$. We proceed by induction.

Base case ($t = 0$): $\bar{\Pi}_0(x_0) = \delta(x_0) = \pi_0(x_0)$.

Inductive step: Assume $\int \bar{\Pi}_{t-1}(x_{0:t-1}, u_{1:t-1})\, du_{1:t-1} = \pi_{t-1}(x_{0:t-1})$. Since $u_t = (u_t^{1:M})$ is independent of $u_{1:t-1}$ given $x_{0:t-1}$, we can factor:

$$\int \bar{\Pi}_t(x_{0:t}, u_{1:t})\, du_{1:t} = \left(\int \frac{\tau_t(u_t)}{\hat{\psi}_{t-1}(x_{0:t-1})}\, \phi_t^M(u_t|x_{0:t-1})\, \gamma_t^M(x_t|u_t)\, du_t \right) \cdot \pi_{t-1}(x_{0:t-1})\,.$$

For the inner integral, substituting the definitions of τ_t, ϕ_t^M, γ_t^M and applying the delta functions from γ_t^M:

$$\begin{aligned}
&\int \tau_t(u_t)\phi_t^M(u_t|x_{0:t-1})\gamma_t^M(x_t|u_t)\, du_t^{1:M} \\
&= \sum_{l=1}^M \int \frac{\frac{1}{M}\sum_j \hat{\psi}_t(x_{0:t-1}, u_t^j) \cdot \hat{\psi}_t(x_{0:t-1}, x_t)}{\sum_m \hat{\psi}_t(x_{0:t-1}, u_t^m)}\Bigg|_{u_t^l = x_t} p_\theta^{\text{ref}}(x_t|x_{0:t-1}) \prod_{j\neq l} p_\theta^{\text{ref}}(u_t^j|x_{0:t-1})\, du_t^{j\neq l}\,.
\end{aligned}$$

The key observation is that with $u_t^l = x_t$ pinned, the numerator $\frac{1}{M}\sum_j \hat{\psi}_t(\cdot, u_t^j)$ is exactly $\frac{1}{M}$ times the denominator $\sum_m \hat{\psi}_t(\cdot, u_t^m)$.

Continuing, we have

$$\begin{aligned}
&= \sum_{l=1}^M \int \frac{1}{M}\hat{\psi}_t(x_{0:t-1}, x_t) p_\theta^{\text{ref}}(x_t|x_{0:t-1}) \prod_{j\neq l} p_\theta^{\text{ref}}(u_t^j|x_{0:t-1})\, du_t^{j\neq l} \\
&= \sum_{l=1}^M \frac{1}{M}\hat{\psi}_t(x_{0:t-1}, x_t) p_\theta^{\text{ref}}(x_t|x_{0:t-1}) \\
&= \hat{\psi}_t(x_{0:t-1}, x_t)\, p_\theta^{\text{ref}}(x_t|x_{0:t-1})\,.
\end{aligned}$$

Including the $1/\hat{\psi}_{t-1}(x_{0:t-1})$ prefactor from the definition of $\bar{\Pi}_t$, the full inner integral becomes

$$\frac{\hat{\psi}_t(x_{0:t})\, p_\theta^{\text{ref}}(x_t|x_{0:t-1})}{\hat{\psi}_{t-1}(x_{0:t-1})} = \frac{p^{\text{ref}}(x_{0:t})\hat{\psi}_t(x_{0:t})}{p^{\text{ref}}(x_{0:t-1})\hat{\psi}_{t-1}(x_{0:t-1})} = \frac{\pi_t(x_{0:t})}{\pi_{t-1}(x_{0:t-1})}$$

where we used the Markov factorization $p^{\text{ref}}(x_{0:t}) = p^{\text{ref}}(x_{0:t-1})\, p^{\text{ref}}(x_t|x_{0:t-1})$ and the definition $\pi_t(x_{0:t}) = p^{\text{ref}}(x_{0:t})\hat{\psi}_t(x_{0:t})$. Combined with the inductive hypothesis:

$$\int \bar{\Pi}_t(x_{0:t}, u_{1:t})\, du_{1:t} = \frac{\pi_t(x_{0:t})}{\pi_{t-1}(x_{0:t-1})} \cdot \pi_{t-1}(x_{0:t-1}) = \pi_t(x_{0:t})$$

which finishes the induction.

The claim at $t = T$ then gives $\int \bar{\Pi}_T(x_{0:T}, u_{1:T})\, du_{1:T} = \pi_T(x_{0:T}) = p_\theta^{\text{ref}}(x_{0:T})\, r(x_{0:T})$ as desired. □

Consequently, $\bar{\Pi}_t$ is normalized by the same constant Z_t as π_t, and $\bar{\Pi}_t(x_{0:t}, u_{1:t})$ admits $\pi_t(x_{0:t})$ as the x-marginal. Our algorithm is a SMC sampler for the (unnormalized) target sequence $\{\bar{\Pi}_t\}_t$ with proposal distribution $\phi_t^M(u_t|x_{0:t-1})\gamma_t^M(x_t|u_t)$, and the weights for each step $t \to t+1$ are confirmed by (15). This proof follows the general framework set up in (Naesseth et al., 2015, Appendix A.1).

Proof of Proposition 1. **Unbiasedness.** We show $\mathbb{E}[\hat{Z}_T] = Z_T$ by adapting the classical SMC unbiasedness argument (Naesseth et al., 2019, Section 4.A) to the extended state space from Lemma 1.

Write $y_t^k = (x_t^k, u_t^k)$ for the extended particle at step t, where $u_t^k = (u_t^{k,1}, \dots, u_t^{k,M})$ are the M inner IS samples. Define $S_t := \sum_{k=1}^K w_t^k$, with $w_0^k = 1$ and $S_0 = K$. Abbreviate the extended normalized proposal

$$\mathcal{Q}_t^k := \phi_t^M(u_t^k \mid x_{1:t-1}^{a_{t-1}^k})\, \gamma_t^M(x_t^k \mid u_t^k)\,,$$

where a_{t-1}^k is the index of the particle at step $t-1$ resampled as parent of particle k at step t.

By Lemma 1, Algorithm 1 is a standard SMC on the extended state space. The joint law of all extended particles $y_{1:T}^{1:K}$ and multinomial resampling indices $a_{1:T-1}^{1:K}$ is

$$Q\big(y_{1:T}^{1:K},\, a_{1:T-1}^{1:K}\big) \;=\; \prod_{k=1}^K \mathcal{Q}_1^k \;\cdot\; \prod_{t=2}^T \prod_{k=1}^K \frac{w_{t-1}^{a_{t-1}^k}}{S_{t-1}}\, \mathcal{Q}_t^k\,. \tag{16}$$

At $t{=}1$ all particles share the fixed initial frame x_0, so there is no resampling and each draws $y_1^k \sim \mathcal{Q}_1(\cdot)$ independently.

For each terminal index $i \in [K]$, define the ancestral lineage $b_T^i = i$ and $b_t^i = a_t^{b_{t+1}^i}$ for $t = T{-}1, \dots, 1$, tracing back the unique ancestral path of particle i. In other words, the reconstructed path $x_{1:T}^i = x_{1:T}^{b_{1:T}^i} = (x_1^{b_1^i}, x_2^{b_2^i}, \dots, x_T^i)$ traces through the ancestry chain for the final particle $x_{1:T}^i$.

Substituting $\hat{Z}_T = K^{-T} \prod_{t=1}^T S_t$ and expanding (16):

$$\frac{\hat{Z}_T}{Z_T}\, Q \;=\; \frac{1}{K^T\, Z_T} \prod_{t=1}^T S_t \;\cdot\; \prod_{k=1}^K \mathcal{Q}_1^k \;\cdot\; \prod_{t=2}^T \prod_{k=1}^K \frac{w_{t-1}^{a_{t-1}^k}}{S_{t-1}}\, \mathcal{Q}_t^k\,. \tag{17}$$

The resampling denominators at step t ($t \geq 2$) contribute S_{t-1}^{-K}, while $\hat{Z}_T$ contributes S_{t-1} at index $t{-}1$. These partially cancel:

$$\prod_{t=1}^T S_t \cdot \prod_{t=2}^T S_{t-1}^{-K} = S_T \cdot \prod_{s=1}^{T-1} S_s^{1-K}.$$

Now fix a terminal index i and write $S_T = \sum_{i=1}^K w_T^{b_T^i} = \sum_{i=1}^K w_T^i$. At each step $s = 1, \dots, T{-}1$, separate the lineage particle b_{s+1}^i (whose ancestor is $b_s^i = a_s^{b_{s+1}^i}$) from the remaining $K{-}1$ non-lineage particles:

$$\frac{\prod_{k=1}^K w_s^{a_s^k}}{S_s^{K-1}} = w_s^{b_s^i} \cdot \prod_{k \neq b_{s+1}^i} \frac{w_s^{a_s^k}}{S_s}\,.$$

The first factor above is the unnormalized ancestor weight on the lineage; the remaining factors are the normalized resampling probabilities $\bar{w}_s^{a_s^k} := w_s^{a_s^k}/S_s$ for non-lineage particles. Similarly, separate the proposals in (17) as: $\prod_k \mathcal{Q}_t^k = \mathcal{Q}_t^{b_t^i} \cdot \prod_{k \neq b_t^i} \mathcal{Q}_t^k$ for a fixed i.

Collecting across all steps, we have

$$\begin{aligned}
\frac{\hat{Z}_T}{Z_T}\, Q &= \frac{1}{K^T\, Z_T} \left(\sum_{i=1}^K w_T^{b_T^i} \right) \prod_{s=1}^{T-1} \left(w_s^{b_s^i} \cdot \prod_{k \neq b_{s+1}^i} \bar{w}_s^{a_s^k} \right) \prod_{t=1}^T \left(\mathcal{Q}_t^{b_t^i} \cdot \prod_{k \neq b_t^i} \mathcal{Q}_t^k \right) \\
&= \frac{1}{K^T\, Z_T} \sum_{i=1}^K \Bigg[\underbrace{\prod_{t=1}^T w_t^{b_t^i}\, \mathcal{Q}_t^{b_t^i}}_{\text{lineage of } i} \;\cdot\; \underbrace{\prod_{j \neq b_1^i} \mathcal{Q}_1^j \;\cdot\; \prod_{t=2}^T \prod_{k \neq b_t^i} \bar{w}_{t-1}^{a_{t-1}^k}\, \mathcal{Q}_t^k}_{\text{non-lineage}} \Bigg].
\end{aligned} \tag{18}$$

By the recursive definition $\bar{\Pi}_t(y_{1:t}) = \frac{\tau_t(u_t)}{\hat{\psi}_{t-1}(x_{1:t-1})}\, \phi_t^M(u_t \mid x_{1:t-1})\, \gamma_t^M(x_t \mid u_t) \cdot \bar{\Pi}_{t-1}(y_{1:t-1})$ from Lemma 1, each factor satisfies

$$w_t^{b_t^i}\, \mathcal{Q}_t^{b_t^i} = \frac{\tau_t}{\hat{\psi}_{t-1}}\, \mathcal{Q}_t^{b_t^i} = \bar{\Pi}_t(y_{1:t}^{b_{1:t}^i}) / \bar{\Pi}_{t-1}(y_{1:t-1}^{b_{1:t-1}^i}),$$

and the product telescopes:

$$\prod_{t=1}^{T} w_t^{b_t^i}\, \mathcal{Q}_t^{b_t^i} = \frac{\bar{\Pi}_T(y_{1:T}^{b_{1:T}^i})}{\bar{\Pi}_0(x_0)} = \bar{\Pi}_T(y_{1:T}^{b_{1:T}^i})\,,$$

where $\bar{\Pi}_0(x_0) = \pi_0(x_0) = 1$ since x_0 is the fixed deterministic initial state.

Take expectations of (18) (integrate over $y_{1:T}^{1:K}$ and sum over $a_{1:T-1}^{1:K}$). By symmetry all K terms in $\sum_i$ contribute equally. So picking $i = 1$ and multiply by K:

$$\mathbb{E}_Q\left[\frac{\hat{Z}_T}{Z_T}\right] = \frac{1}{K^{T-1}Z_T} \sum_{a_{1:T-1}^{1:K}} \int \bar{\Pi}_T(y_{1:T}^{b_{1:T}^1}) \cdot \prod_{j \neq b_1^1} \mathcal{Q}_1^j \cdot \prod_{t=2}^{T} \prod_{k \neq b_t^1} \bar{w}_{t-1}^{a_{t-1}^k}\, \mathcal{Q}_t^k \; dy_{1:T}^{1:K}\,.$$

Each non-lineage particle $k \neq b_t^1$ at step $t \geq 2$ marginalizes to 1:

$$\sum_{a_{t-1}^k=1}^{K} \bar{w}_{t-1}^{a_{t-1}^k} \int \mathcal{Q}_t^k \, dy_t^k = \underbrace{\textstyle\sum_{a_{t-1}^k} \frac{w_{t-1}^{a_{t-1}^k}}{S_{t-1}}}_{=1} \cdot \underbrace{\int \mathcal{Q}_t^k \, dy_t^k}_{=1} = 1\,,$$

and similarly $\int \mathcal{Q}_1^j \, dy_1^j = 1$ for $j \neq b_1^1$.

After marginalizing all non-lineage particles, the ancestor chain $b_{1:T-1}^1$ ranges over K^{T-1} values (each $b_t^1 \in [K]$ for $t = 1, \ldots, T-1$), and by relabeling each gives the same integral:

$$\mathbb{E}\left[\frac{\hat{Z}_T}{Z_T}\right] = \frac{K^{T-1}}{K^{T-1}Z_T} \int \bar{\Pi}_T(y_{1:T})\, dy_{1:T} = \frac{1}{Z_T} \int \bar{\Pi}_T(x_{1:T}, u_{1:T})\, dx_{1:T}\, du_{1:T}\,.$$

By the marginal property $\int \bar{\Pi}_T(x_{1:T}, u_{1:T})\, du_{1:T} = \pi_T(x_{1:T})$ from Lemma 1:

$$\mathbb{E}\left[\frac{\hat{Z}_T}{Z_T}\right] = \frac{1}{Z_T} \int \pi_T(x_{1:T})\, dx_{1:T} = \frac{Z_T}{Z_T} = 1\,,$$

concluding $\mathbb{E}[\hat{Z}_T] = Z_T$ is unbiased for any $M > 1$.

Consistency (setwise convergence as $K \to \infty$ to the target (1). This is a direct consequence of (Chopin & Papaspiliopoulos, 2020, Proposition 11.4), by viewing Algorithm 1 as a standard SMC sampler on the extended state space $(x_{0:t}, u_{1:t})$ with unnormalized target sequence $\{\bar{\Pi}_t\}_{t=0}^T$ and bounded incremental weights w_t. □

A.3. Omitted Proofs from Section 5

Proof of Lemma 2. We write $\bar{p}_{t+1}^s(x|x_t) = p_0(x) \cdot h_s(x)/\mathcal{Z}_s$ with p_0 a static base measure and the guidance $h_s = p_s(\cdot|x_t)/p_s$ a ratio of two OU diffusion marginals. Since $p_s(\cdot|x_t)$ and p_s both satisfy the OU Fokker-Planck equation, we handle h_s with the product-of-marginals FKC framework (Skreta et al., 2025, Prop. D.5). The static factor p_0 does *not* satisfy Fokker-Planck; we incorporate it via a reward-drift mechanism analogous to (Skreta et al., 2025, Prop. 3.4), adding $\frac{1}{2}\nabla \log p_0$ to the SDE drift so that the Laplacian correction $\nabla \cdot j_0^\theta$ cancels exactly.

The h-function has score $\nabla \log h_s = o_s^\theta - j_s^\theta$. By (Skreta et al., 2025, Eq. 242), the product SDE for $h_s = (q_s^1)^{+1}(q_s^2)^{-1}$ ($q_s^1 = p_s(\cdot|x_t)$, $q_s^2 = p_s$) has drift $\frac{1}{2}x + o_s^\theta - j_s^\theta$. Adding the half-strength reward drift $\frac{1}{2}j_0^\theta$ yields (8).

By (Skreta et al., 2025, Eq. 243) with $\beta_1 = +1$, $\beta_2 = -1$ ($\sum_i \beta_i = 0$) and scores o_s^θ, j_s^θ, the weights evolve as: $d \log w_s = g_s^h ds$ where

$$g_s^h = (0-1)\big\langle \nabla, \underbrace{-\tfrac{1}{2}x}_{f_{\mathrm{OU}}} \big\rangle + \tfrac{1}{2}\|o_s^\theta - j_s^\theta\|^2 - \tfrac{1}{2}\big[\|o_s^\theta\|^2 - \|j_s^\theta\|^2\big] = \tfrac{d}{2} + \big\langle j_s^\theta,\, j_s^\theta - o_s^\theta \big\rangle.$$

This weighted SDE traces the curve $h_s(x_s)$.

Write the unnormalized target $\tilde{p}_s = p_0 \cdot h_s$ and consider the drift $v = \frac{1}{2}x + o_s - j_s + \alpha\, j_0^\theta$. The FK weight rate g_s is determined by the FK-PDE $\partial_s \tilde{p}_s = -\langle \nabla, \tilde{p}_s\, v\rangle + \frac{1}{2}\Delta \tilde{p}_s + g_s\, \tilde{p}_s$, which means

$$g_s = (\partial_s \tilde{p}_s + \langle \nabla, \tilde{p}_s\, v\rangle - \tfrac{1}{2}\Delta \tilde{p}_s)/\tilde{p}_s\,.$$

Since p_0 is static ($\partial_s p_0 = 0$), we have $\partial_s \tilde{p}_s = p_0\, \partial_s h_s$, and the product rules give

$$\begin{aligned}\langle \nabla, \tilde{p}_s\, v\rangle &= p_0 \langle \nabla, h_s\, v\rangle + h_s \langle \nabla p_0,\, v\rangle,\\ \tfrac{1}{2}\Delta \tilde{p}_s &= \tfrac{1}{2} p_0\, \Delta h_s + \langle \nabla p_0,\, \nabla h_s\rangle + \tfrac{1}{2} h_s\, \Delta p_0\,.\end{aligned}$$

Dividing through by $\tilde{p}_s = p_0 h_s$ and using $\nabla p_0/p_0 = j_0^\theta$, $\frac{\Delta p_0}{p_0} = \nabla \cdot j_0^\theta + \|j_0^\theta\|^2$:

$$g_s = \underbrace{\frac{\partial_s h_s + \langle \nabla, h_s\, v\rangle - \frac{1}{2}\Delta h_s}{h_s}}_{g_s^{h,\alpha}\text{: depends on drift choice } \alpha} + \underbrace{\langle j_0^\theta,\, v\rangle - \langle j_0^\theta,\, \nabla \log h_s\rangle - \tfrac{1}{2}\nabla \cdot j_0^\theta - \tfrac{1}{2}\|j_0^\theta\|^2}_{\text{residual from static } p_0}\,.$$

The first underbraced term $g_s^{h,\alpha}$ equals g_s^h when $\alpha = 0$; for general α, the extra αj_0^θ in the drift contributes $\alpha\, \nabla \cdot j_0^\theta + \alpha\langle j_0^\theta,\, \nabla \log h_s\rangle$ to $g_s^{h,\alpha}$ via $\langle \nabla, h_s \cdot \alpha j_0^\theta\rangle/h_s = \alpha\, \nabla \cdot j_0^\theta + \alpha\langle j_0^\theta,\, \nabla \log h_s\rangle$. Absorbing this into the second term and substituting $v - \nabla \log h_s = \frac{1}{2}x + \alpha\, j_0^\theta$, the total p_0-dependent contribution becomes:

$$\begin{aligned} g_s^{p_0}(\alpha) &= \langle j_0^\theta,\, \tfrac{1}{2}x + \alpha\, j_0^\theta\rangle + \alpha\langle j_0^\theta,\, o_s - j_s\rangle + \alpha\, \nabla \cdot j_0^\theta - \tfrac{1}{2}\nabla \cdot j_0^\theta - \tfrac{1}{2}\|j_0^\theta\|^2\\ &= \tfrac{1}{2}\langle j_0^\theta,\, x\rangle + \big(\alpha - \tfrac{1}{2}\big)\big(\nabla \cdot j_0^\theta + \|j_0^\theta\|^2\big) + \alpha\langle j_0^\theta,\, o_s - j_s\rangle\,.\end{aligned}$$

Setting $\alpha = \frac{1}{2}$ eliminates the Laplacian $\nabla \cdot j_0^\theta$ and the squared norm $\|j_0^\theta\|^2$:

$$g_s^{p_0}\big(\tfrac{1}{2}\big) = \tfrac{1}{2}\langle j_0^\theta,\, x + o_s - j_s\rangle\,.$$

The total weight rate $g_s = g_s^h + g_s^{p_0} = \frac{d}{2} + \langle j_s,\, j_s - o_s\rangle + \frac{1}{2}\langle j_0^\theta,\, x + o_s - j_s\rangle$ gives (9). The FK-PDE framework (Skreta et al., 2025, Prop. A.1) targets the unnormalized density $\tilde{p}_s = p_0 h_s$; since $\mathcal{Z}_s = \int p_0 h_s\, dx$ is unknown, we use self-normalized importance weights $\bar{w}^m = w^m / \sum_l w^l$ (where $w^m = \exp(\int_0^S g_s\, ds)$ along the m-th particle path), which estimate expectations under $\bar{p}_s = \tilde{p}_s/\mathcal{Z}_s$ without requiring $\mathcal{Z}_s$. □

Remark (Weight Requirement). The reason for the additional weights for sampling from composition density $\propto q_1 q_2$ comes from the fact that for 2 general diffusion process on q_1, q_2,

$$\begin{aligned} o_1^s(x^s) + o_2^s(x^s) &= \nabla \log \int q_1(x^0)\mathcal{K}(x^s|x^0)dx^0 + \nabla \log \int q_2(x^0)\mathcal{K}(x^s|x^0)dx^0\\ &\neq \nabla \log \int q_1(x^0)q_2(x^0)\mathcal{K}(x^s|x^0)dx^0 =: o_{1,2}^s(x^s)\end{aligned}$$

where the second line is the *correct* score to use for sampling from the product distribution $\propto q_1 q_2$. Even though at $s = 0$ the scores $o_{1,2}^0(x^0) = o_1^0(x^0) + o_2^0(x^0) \approx \nabla \log q_1(x) + \nabla \log q_2(x) = \nabla \log(q_1(x)q_2(x))$, this is not true at arbitrary $s > 0$ during the generation process (with noising kernel $\mathcal{K}$). Therefore naively running with $o_1^s(x^s) + o_2^s(x^s)$ backward without additional correction will not sample from the desired product density.

A.4. Omitted Proofs from Section 6

The objectives proposed in (10) and (11) are based on the following lemma.

Lemma 5 (Optimality of twist learning objectives)**.** *Both objectives* (10) *and* (11) *have* ψ_t^* *as a fixed point and characterize optimality as follows.*

(i) (Monte Carlo regression). *For objective* (10)*, the minimizer over unconstrained* ψ_t *is* $\psi_t^*(x_{1:t}) = \mathbb{E}_{Q^{ref}}[r(x_{1:T}) \mid x_{1:t}]$*, the conditional expectation of the reward under* Q^{ref}*, which equals* (3) *by definition. This follows from the standard* L^2 *projection characterization:* $\arg\min_{\psi_t} \mathbb{E}[(r(x_{1:T}) - \psi_t(x_{1:t}))^2] = \mathbb{E}[r(x_{1:T})|x_{1:t}]$.

(ii) (Bootstrapped/TD regression). *For objective (11), the fixed point satisfies the Bellman recursion* $\psi_t^*(x_{1:t}) = \mathbb{E}_{Q^{\text{ref}}}[\psi_{t+1}^*(x_{1:t+1}) \mid x_{1:t}]$ *with terminal condition* $\psi_T^* = r$*, consistent with (3) by the tower property. Starting from* $\psi_t^0 = r$*, the iterates converge to* ψ_t^* *as* $j \to \infty$*.*

Proof. We prove each claim in turn.

(i) Monte Carlo regression. Write the population risk as

$$L(\psi_t) = \mathbb{E}_{x_{1:T}\sim Q^{\text{ref}}}\big[(r(x_{1:T}) - \psi_t(x_{1:t}))^2\big].$$

By the law of total expectation, conditioning on $x_{1:t}$:

$$L(\psi_t) = \mathbb{E}_{x_{1:t}}\Big[\mathbb{E}_{x_{t+1:T}\sim Q^{\text{ref}}(\cdot|x_{1:t})}\big[(r(x_{1:T}) - \psi_t(x_{1:t}))^2\big]\Big].$$

For each fixed $x_{1:t}$, the inner expectation is minimized (as a function of $\psi_t(x_{1:t}) \in \mathbb{R}$) by the conditional mean:

$$\psi_t^*(x_{1:t}) = \mathbb{E}_{Q^{\text{ref}}}[r(x_{1:T}) \mid x_{1:t}] = \int p^{\text{ref}}(x_{t+1:T}|x_{1:t})\, r(x_{1:T})\, dx_{t+1:T},$$

which is exactly (3). This is the standard characterization of L^2-projections: the conditional expectation is the unique minimizer of $\mathbb{E}[(Y - g(X))^2]$ over measurable functions $g(X)$.

(ii) Bootstrapped/TD regression. At convergence ($\psi_t^j \to \psi_t^*$ as $j \to \infty$), the fixed-point equation for each t reads:

$$\psi_t^*(x_{1:t}) = \arg\min_c \mathbb{E}_{x_{t+1}\sim Q^{\text{ref}}(\cdot|x_{1:t})}\big[(c - \psi_{t+1}^*(x_{1:t+1}))^2\big] = \mathbb{E}_{Q^{\text{ref}}}[\psi_{t+1}^*(x_{1:t+1}) \mid x_{1:t}],$$

again by the L^2-projection characterization. This is precisely the Bellman recursion $\psi_t^*(x_{1:t}) = \mathbb{E}_{Q^{\text{ref}}}[\psi_{t+1}^*(x_{1:t+1}) \mid x_{1:t}]$ with terminal condition $\psi_T^* = r$. By the tower property of conditional expectation, iterating the recursion from T backwards:

$$\psi_t^*(x_{1:t}) = \mathbb{E}_{Q^{\text{ref}}}[\psi_T^*(x_{1:T}) \mid x_{1:t}] = \mathbb{E}_{Q^{\text{ref}}}[r(x_{1:T}) \mid x_{1:t}],$$

recovering (3). Convergence of the iterates $\psi_t^j \to \psi_t^*$ follows because the map $\psi_{t+1} \mapsto \mathbb{E}[\psi_{t+1}|x_{1:t}]$ is a contraction in $L^2(Q^{\text{ref}})$ (by Jensen's inequality / conditional expectation being an orthogonal projection), and the iteration proceeds backwards from the fixed terminal condition $\psi_T^0 = \psi_T^* = r$. □

From this proof and Lemma 4, we also see that the twist $\hat{\psi}_t = \psi_t^*$ is the unique choice (up to constants) for which the SMC importance weights w_t^k are constant across all k for every t, i.e., the normalizing constant estimator $\hat{Z}_T$ has zero variance.

Lemma 3 below gives an alternative way of training for the twists, based on marginal distribution matching.

Proof of Lemma 3. Write $\sigma(x_{1:t}) = \frac{1}{Z_t^{\psi^*}} p^{\text{ref}}(x_{1:t})\psi_t^*(x_{1:t})$ and $\mathbb{Q}_t^\theta(x_{1:t}) = \frac{1}{Z_t^{\psi^\theta}} p^{\text{ref}}(x_{1:t})\psi_t^\theta(x_{1:t})$. The KL divergence of interest is

$$\begin{aligned}
\mathrm{KL}(\sigma\|\mathbb{Q}_t^\theta) &= \mathbb{E}_{x_{1:t}\sim\sigma}\left[\log\frac{\sigma(x_{1:t})}{\mathbb{Q}_t^\theta(x_{1:t})}\right] \\
&= \mathbb{E}_{x_{1:t}\sim\sigma}\left[\log\frac{Z_t^{\psi^\theta}\ \psi_t^*(x_{1:t})}{Z_t^{\psi^*}\ \psi_t^\theta(x_{1:t})}\right] \\
&= \log Z_t^{\psi^\theta} - \mathbb{E}_\sigma[\log\psi_t^\theta(x_{1:t})] + \text{const}(\theta),
\end{aligned}$$

where $\text{const}(\theta)$ collects terms independent of θ, since $p^{\text{ref}}(x_{1:t})$ cancels in the ratio.

Using $\sigma(x_{1:t}) = \frac{1}{Z_t^{\psi^*}} p^{\text{ref}}(x_{1:t})\psi_t^*(x_{1:t})$ and the definition of ψ_t^* from (3):

$$\mathbb{E}_{x_{1:t}\sim\sigma}\big[h(x_{1:t})\big] = \frac{1}{Z_t^{\psi^*}}\mathbb{E}_{x_{1:t}\sim p^{\text{ref}}}\big[\psi_t^*(x_{1:t})\, h(x_{1:t})\big]$$

$$= \frac{1}{Z_t^{\psi^*}} \mathbb{E}_{x_{1:t}\sim p^{\text{ref}}}\left[\mathbb{E}_{x_{t+1:T}\sim p^{\text{ref}}(\cdot|x_{1:t})}\left[r(x_{1:T})\right] h(x_{1:t})\right]$$
$$= \frac{1}{Z_t^{\psi^*}} \mathbb{E}_{x_{1:T}\sim p^{\text{ref}}}\left[r(x_{1:T})\, h(x_{1:t})\right],$$

where the last equality uses the tower property: $\mathbb{E}_{x_{1:t}}[\mathbb{E}_{x_{t+1:T}|x_{1:t}}[r]\cdot h(x_{1:t})] = \mathbb{E}_{x_{1:T}}[r\cdot h(x_{1:t})]$. Applying this with $h(x_{1:t}) = \log Z_t^{\psi^\theta} - \log(\psi_t^\theta(x_{1:t}))$ yields the objective (up to the global constant $1/Z_t^{\psi^*}$ which is θ-independent and can be dropped from the $\arg\min$):

$$\mathcal{L}_t(\theta) = \mathbb{E}_{x_{1:T}\sim p^{\text{ref}}}[r(x_{1:T})] \times \log Z_t^{\psi^\theta} - \mathbb{E}_{x_{1:T}\sim p^{\text{ref}}}\left[r(x_{1:T})\, \log \psi_t^\theta(x_{1:t})\right].$$

We compute the gradient of each term separately. The results follow from summing over $t \in [T]$.

First term: $\nabla_\theta \log Z_t^{\psi^\theta}$. Since $Z_t^{\psi^\theta} = \int p^{\text{ref}}(x_{1:t})\psi_t^\theta(x_{1:t})\, dx_{1:t}$:

$$\begin{aligned}\nabla_\theta \log Z_t^{\psi^\theta} &= \frac{1}{Z_t^{\psi^\theta}} \int p^{\text{ref}}(x_{1:t})\, \nabla_\theta \psi_t^\theta(x_{1:t})\, dx_{1:t} \\ &= \frac{1}{Z_t^{\psi^\theta}} \int p^{\text{ref}}(x_{1:t})\, \psi_t^\theta(x_{1:t})\, \nabla_\theta \log \psi_t^\theta(x_{1:t})\, dx_{1:t} \\ &= \mathbb{E}_{x_{1:t}\sim \mathbb{Q}_t^\theta}\left[\nabla_\theta \log \psi_t^\theta(x_{1:t})\right],\end{aligned}$$

where we recognized the expectation under $\mathbb{Q}_t^\theta(x_{1:t}) = \frac{1}{Z_t^{\psi^\theta}} p^{\text{ref}}(x_{1:t})\psi_t^\theta(x_{1:t})$.

Second term: $\nabla_\theta \mathbb{E}_{p^{\text{ref}}}[r(x_{1:T}) \log \psi_t^\theta(x_{1:t})]$. Since p^{ref} does not depend on θ, the gradient passes inside the expectation:

$$\mathbb{E}_{x_{1:T}\sim p^{\text{ref}}}\left[r(x_{1:T})\, \nabla_\theta \log \psi_t^\theta(x_{1:t})\right].$$

Combining, the stochastic gradient (up to the constant $1/Z_t^{\psi^*}$ which does not affect the $\arg\min$) is

$$\nabla_\theta \mathcal{L}_t(\theta) = \mathbb{E}_{x_{1:t}\sim \mathbb{Q}_t^\theta}\left[\nabla_\theta \log \psi_t^\theta(x_{1:t})\right] \times \mathbb{E}_{x_{1:T}\sim p^{\text{ref}}}[r(x_{1:T})] - \mathbb{E}_{x_{1:T}\sim p^{\text{ref}}}\left[r(x_{1:T})\, \nabla_\theta \log \psi_t^\theta(x_{1:t})\right].$$

Both expectations are estimable from samples: the first requires running the current twisted SMC to sample $x_{1:t} \sim \mathbb{Q}_t^\theta$, and the second uses trajectories $x_{1:T} \sim p^{\text{ref}}$ directly. □

B. Variance Analysis

In this section, we analyze the variance of the normalizing constant estimator $\hat{Z}_T$ for all three algorithms in a unified extended-state SMC framework. The exact CLT follows from the standard SMC theory of (Chopin, 2004) after applying the nested-SMC construction of (Naesseth et al., 2015). We then isolate a local one-step decomposition into prefix mismatch and inner-IS noise, which is the quantity used in the algorithmic comparisons below.

B.1. Variance decomposition for Nested SMC

Our terminal target is the normalized path law $\mathbb{Q}_T(x_{1:T}) = p^{\text{ref}}(x_{1:T})\, r(x_{1:T})/Z_T$. Given a twist sequence $\hat{\psi}_0, \ldots, \hat{\psi}_T$ with $\hat{\psi}_T = r$, denote the unnormalized twisted intermediate target and its normalized law by

$$\pi_t^{\hat{\psi}}(x_{1:t}) := p^{\text{ref}}(x_{1:t})\, \hat{\psi}_t(x_{1:t}), \qquad \mathbb{Q}_t(x_{1:t}) := \frac{\pi_t^{\hat{\psi}}(x_{1:t})}{Z_t^{\hat{\psi},\text{glob}}},$$

where $Z_t^{\hat{\psi},\text{glob}} := \int \pi_t^{\hat{\psi}}(x_{1:t})\, dx_{1:t}$. At each outer loop, nested SMC introduces inner samples $u_t = (u_t^1, \ldots, u_t^M)$ and selected state x_t. We write $y_t = (x_t, u_t)$ and $y_{1:t} = (x_{1:t}, u_{1:t})$. Given a prefix $x_{1:t-1}$, the extended proposal is

$$\mathcal{Q}_t^M(dy_t|x_{1:t-1}) := \phi_t^M(du_t|x_{1:t-1})\, \gamma_t^M(dx_t|x_{1:t-1}, u_t),$$

where

$$\phi_t^M(du_t|x_{1:t-1}) := \prod_{j=1}^M p^{\mathrm{ref}}(du_t^j|x_{1:t-1})$$

and

$$\gamma_t^M(dx_t|x_{1:t-1}, u_t) := \sum_{j=1}^M \frac{\hat{\psi}_t(x_{1:t-1}, u_t^j)}{\sum_{\ell=1}^M \hat{\psi}_t(x_{1:t-1}, u_t^\ell)} \delta_{u_t^j}(dx_t).$$

The extended incremental weight is

$$G_t^M(y_{1:t}) := \frac{\tau_t(u_t; x_{1:t-1})}{\hat{\psi}_{t-1}(x_{1:t-1})}, \qquad \tau_t(u_t; x_{1:t-1}) := \frac{1}{M}\sum_{j=1}^M \hat{\psi}_t(x_{1:t-1}, u_t^j). \tag{19}$$

Crucially, G_t^M depends on the prefix and the inner samples, but not on the selected x_t after u_t is fixed. By Lemma 1, this is a standard SMC sampler on the extended state space:

- $\Pi_t^M(dy_{1:t})$ denotes the normalized extended target at time t;
- its x-marginal is $\mathbb{Q}_t(dx_{1:t})$;
- the prediction law before the step-t correction is

$$\widetilde{\Pi}_t^M(dy_{1:t}) := \Pi_{t-1}^M(dy_{1:t-1})\mathcal{Q}_t^M(dy_t|x_{1:t-1}).$$

We normalize the extended weight by its prediction mean,

$$\bar{G}_t^M(y_{1:t}) := \frac{G_t^M(y_{1:t})}{\widetilde{\Pi}_t^M(G_t^M)}. \tag{20}$$

Finally define the normalized future factor

$$\Lambda_{t,T}^M(y_{1:t}) := \mathbb{E}\left[\prod_{\ell=t+1}^T \bar{G}_\ell^M(Y_{1:\ell}) \,\middle|\, Y_{1:t} = y_{1:t}\right], \qquad \Lambda_{T,T}^M \equiv 1, \tag{21}$$

where the expectation is taken w.r.t the extended proposal kernels at later times. This factor is the normalized expected future normalizing constant from a particle prefix. It is exactly the object that records how CLT-scale particle error at time t propagates through later SMC corrections.

The quantity (20) is defined as such since the true normalizing constant $Z_T = \prod_{t=1}^T \widetilde{\Pi}_t^M(G_t^M)$, therefore $\hat{Z}_T/Z_T = \prod_{t=1}^T \frac{1}{K}\sum_{k=1}^K \bar{G}_t^M(y_{1:t}^k)$, which is the quantity we want to analyze.

Theorem 1 (Extended-state asymptotic variance of Nested SMC)**.** *Assume the moment conditions of the SMC CLT hold for the extended-state nested SMC sampler of Lemma 1. Let Π_t^M be the normalized extended target, $\widetilde{\Pi}_t^M(dy_{1:t}) = \Pi_{t-1}^M(dy_{1:t-1})\mathcal{Q}_t^M(dy_t|y_{1:t-1})$ the prediction law before weighting, and G_t^M the extended incremental weight. Define $\bar{G}_t^M = G_t^M/\widetilde{\Pi}_t^M(G_t^M)$ and*

$$\Lambda_{t,T}^M(y_{1:t}) := \mathbb{E}\left[\prod_{\ell=t+1}^T \bar{G}_\ell^M(Y_{1:\ell}) \,\middle|\, Y_{1:t} = y_{1:t}\right], \qquad \Lambda_{T,T}^M \equiv 1.$$

Then with multinomial resampling,

$$\sqrt{K}\left(\frac{\hat{Z}_T^K}{Z_T} - 1\right) \xrightarrow{d} \mathcal{N}(0, \sigma_{Z,T}^{2,M}),$$

where

$$\sigma_{Z,T}^{2,M} = \sum_{t=1}^T \widetilde{\Pi}_t^M\left[\left(\bar{G}_t^M \Lambda_{t,T}^M - 1\right)^2\right]. \tag{22}$$

Proof. By Lemma 1 and the extended-target construction of (Naesseth et al., 2015), the nested sampler is an ordinary auxiliary SMC sampler on $y_{1:t}$. The CLT of (Chopin, 2004) therefore applies to this extended model. Using (Chopin, 2004)'s notation, for a test function f on the extended time-t space,

$$\widetilde{V}_t^M(f) = \widehat{V}_{t-1}^M(\mathcal{Q}_t^M f) + \Pi_{t-1}^M\left[\mathrm{Var}_{\mathcal{Q}_t^M}(f)\right],$$

followed by the correction and resampling recursions

$$V_t^M(f) = \widetilde{V}_t^M\left(\bar{G}_t^M(f - \Pi_t^M f)\right), \qquad \widehat{V}_t^M(f) = V_t^M(f) + \mathrm{Var}_{\Pi_t^M}(f).$$

Equivalently, applying the same Chopin–Feynman–Kac recursion to the relative normalizing-constant estimator gives the closed-form variance obtained by propagating the terminal constant function through the normalized future-mass operators. In our notation the time-t influence function is

$$\bar{G}_t^M(y_{1:t})\mathbb{E}\left[\prod_{\ell=t+1}^{T}\bar{G}_\ell^M(Y_{1:\ell})\,\middle|\,Y_{1:t}=y_{1:t}\right] = \bar{G}_t^M(y_{1:t})\Lambda_{t,T}^M(y_{1:t}).$$

This random variable has $\widetilde{\Pi}_t^M$-mean one – this follows from normalization:

$$\widetilde{\Pi}_t^M[\bar{G}_t^M\Lambda_{t,T}^M] = \Pi_t^M[\Lambda_{t,T}^M],$$

since

$$\Pi_t^M(dy_{1:t}) = \bar{G}_t^M(y_{1:t})\widetilde{\Pi}_t^M(dy_{1:t}).$$

Therefore by tower property,

$$\Pi_t^M[\Lambda_{t,T}^M] = \mathbb{E}\left[\prod_{\ell=t+1}^{T}\bar{G}_\ell^M(Y_{1:\ell})\right] = 1$$

as recursively,

$$\mathbb{E}\left[\bar{G}_{t+1}^M(Y_{1:t+1})\,\middle|\,Y_{1:t}\sim\Pi_t^M\right] = \widetilde{\Pi}_{t+1}^M[\bar{G}_{t+1}^M] = 1.$$

So the corresponding variance contribution at t is $\widetilde{\Pi}_t^M[(\bar{G}_t^M\Lambda_{t,T}^M - 1)^2]$. Summing over t gives (22). □

This exact variance is the right object for the terminal estimator. It contains the backward future factor $\Lambda_{t,T}^M$, which accounts for propagation of earlier particle error through later SMC steps. For the purpose of comparing proposal and twist mechanisms, however, it is useful to isolate the one-step contribution obtained when the prefix at time $t-1$ is treated as an exact draw from $\mathbb{Q}_{t-1}$. The following simplified result replacing $\Lambda_{t,T}^M$ by 1 exactly captures the local variance under the iid prefix assumption. Write

$$Z_t^{\hat{\psi}}(x_{1:t-1}) := \int p^{\mathrm{ref}}(x_t|x_{1:t-1})\hat{\psi}_t(x_{1:t-1},x_t)\,dx_t$$

for the local normalizing constant,

$$q_t^*(dx_t|x_{1:t-1}) := \frac{p^{\mathrm{ref}}(dx_t|x_{1:t-1})\hat{\psi}_t(x_{1:t-1},x_t)}{Z_t^{\hat{\psi}}(x_{1:t-1})}$$

for the locally optimal proposal, and

$$h_t(x_{1:t-1}) := \frac{\mathbb{Q}_t(x_{1:t-1})}{\mathbb{Q}_{t-1}(x_{1:t-1})},$$

where $\mathbb{Q}_t(x_{1:t-1})$ denotes the $x_{1:t-1}$-marginal of $\mathbb{Q}_t$.

Proposition 2 (Local iid-prefix variance decomposition)**.** *Assume the terms below are finite. For each t,*

$$\widetilde{\Pi}_t^M\left[(\bar{G}_t^M - 1)^2\right] = R_t + \frac{1}{M}D_t^{\hat{\psi}}, \tag{23}$$

where

$$R_t := \chi^2\left(\mathbb{Q}_t(x_{1:t-1})\,\middle\|\,\mathbb{Q}_{t-1}(x_{1:t-1})\right) = \mathrm{Var}_{\mathbb{Q}_{t-1}}(h_t)$$

and

$$D_t^{\hat{\psi}} := \mathbb{E}_{\mathbb{Q}_{t-1}}\left[h_t(x_{1:t-1})^2\chi^2\left(q_t^*(\cdot|x_{1:t-1})\,\middle\|\,p^{\mathrm{ref}}(\cdot|x_{1:t-1})\right)\right].$$

Proof. Under $\widetilde{\Pi}_t^M$, the prefix marginal is $x_{1:t-1} \sim \mathbb{Q}_{t-1}$ and the inner samples are iid from $p^{\mathrm{ref}}(\cdot|x_{1:t-1})$. Conditioning on $x_{1:t-1}$,

$$\begin{aligned}\mathbb{E}[G_t^M \mid x_{1:t-1}] &= \frac{1}{\hat{\psi}_{t-1}(x_{1:t-1})}\,\mathbb{E}\left[\frac{1}{M}\sum_{j=1}^{M}\hat{\psi}_t(x_{1:t-1},u_t^j)\;\middle|\; x_{1:t-1}\right]\\ &= \frac{1}{\hat{\psi}_{t-1}(x_{1:t-1})}\int p^{\mathrm{ref}}(x_t|x_{1:t-1})\,\hat{\psi}_t(x_{1:t-1},x_t)\,dx_t\\ &= \frac{Z_t^{\hat{\psi}}(x_{1:t-1})}{\hat{\psi}_{t-1}(x_{1:t-1})},\end{aligned} \tag{24}$$

and the selected x_t does not affect this weight once u_t is fixed. Since $\mathbb{Q}_t(x_{1:t}) \propto p^{\mathrm{ref}}(x_{1:t})\hat{\psi}_t(x_{1:t})$, its $x_{1:t-1}$-marginal is

$$\begin{aligned}\mathbb{Q}_t(x_{1:t-1}) &= \int \mathbb{Q}_t(x_{1:t})\,dx_t = \frac{p^{\mathrm{ref}}(x_{1:t-1})}{Z_t^{\hat{\psi},\mathrm{glob}}}\int p^{\mathrm{ref}}(x_t|x_{1:t-1})\,\hat{\psi}_t(x_{1:t-1},x_t)\,dx_t\\ &= \frac{p^{\mathrm{ref}}(x_{1:t-1})\cdot Z_t^{\hat{\psi}}(x_{1:t-1})}{Z_t^{\hat{\psi},\mathrm{glob}}},\end{aligned}$$

so, with $c_t := Z_{t-1}^{\hat{\psi},\mathrm{glob}}/Z_t^{\hat{\psi},\mathrm{glob}}$,

$$\mathbb{E}[G_t^M \mid x_{1:t-1}] = \frac{1}{c_t}h_t(x_{1:t-1}). \tag{25}$$

Because $\mathbb{E}_{\mathbb{Q}_{t-1}}[h_t] = 1$ by definition, $\widetilde{\Pi}_t^M(G_t^M) = 1/c_t$ and hence $\bar{G}_t^M = c_t G_t^M$. Therefore

$$\mathbb{E}[\bar{G}_t^M|x_{1:t-1}] = h_t(x_{1:t-1}).$$

The law of total variance gives

$$\widetilde{\Pi}_t^M\left[(\bar{G}_t^M - 1)^2\right] = \mathrm{Var}_{\mathbb{Q}_{t-1}}(h_t) + \mathbb{E}_{\mathbb{Q}_{t-1}}\left[\mathrm{Var}(\bar{G}_t^M|x_{1:t-1})\right]. \tag{26}$$

The first term is R_t as claimed. For the second term, conditional on $x_{1:t-1}$, the only randomness in G_t^M is the sample mean τ_t. Hence with M inner samples,

$$\mathrm{Var}(\bar{G}_t^M \mid x_{1:t-1}) = \frac{c_t^2\,\mathrm{Var}_{p^{\mathrm{ref}}}(\hat{\psi}_t(x_{1:t-1},\cdot))}{M\,\hat{\psi}_{t-1}(x_{1:t-1})^2}. \tag{27}$$

Using the definition of q_t^*,

$$\begin{aligned}\mathrm{Var}_{p^{\mathrm{ref}}}\big(\hat{\psi}_t(x_{1:t-1},\cdot)\big) &= \int p^{\mathrm{ref}}(x_t|x_{1:t-1})\,\hat{\psi}_t(x_{1:t-1},x_t)^2\,dx_t - \left(Z_t^{\hat{\psi}}(x_{1:t-1})\right)^2\\ &= \left(Z_t^{\hat{\psi}}\right)^2\left[\int \frac{q_t^*(x_t|x_{1:t-1})^2}{p^{\mathrm{ref}}(x_t|x_{1:t-1})}\,dx_t - 1\right]\\ &= \left(Z_t^{\hat{\psi}}\right)^2\chi^2\big(q_t^*(\cdot|x_{1:t-1})\,\big\|\,p^{\mathrm{ref}}(\cdot|x_{1:t-1})\big),\end{aligned}$$

and (25), together with (24), imply $c_t Z_t^{\hat{\psi}}/\hat{\psi}_{t-1} = h_t$. Substituting into (27) yields

$$\mathrm{Var}(\bar{G}_t^M \mid x_{1:t-1}) = \frac{h_t(x_{1:t-1})^2}{M}\,\chi^2\big(q_t^*(\cdot|x_{1:t-1})\,\big\|\,p^{\mathrm{ref}}(\cdot|x_{1:t-1})\big).$$

Taking expectations over $\mathbb{Q}_{t-1}$ gives $D_t^{\hat{\psi}}/M$ and proves (23). □

We give a lemma below showing that in certain regime $\sum_t(R_t + D_t^{\hat{\psi}}/M)$ is a controlled approximation to the exact terminal CLT variance from Theorem 1. Without this condition, the same expression remains an exact local diagnostic of fresh prefix mismatch and inner-IS noise, but it should not be interpreted as the full terminal asymptotic variance.

Lemma 6 (Backward-stability approximation)**.** *Let*

$$S_{\mathrm{loc}}^{M} := \sum_{t=1}^{T} \left(R_t + \frac{1}{M} D_t^{\hat{\psi}} \right).$$

If for some $\eta \geq 0$*,*

$$\sum_{t=1}^{T} \widetilde{\Pi}_t^M \left[(\bar{G}_t^M)^2 (\Lambda_{t,T}^M - 1)^2 \right] \leq \eta^2 S_{\mathrm{loc}}^M, \tag{28}$$

then the exact terminal asymptotic variance satisfies

$$\left| \sigma_{Z,T}^{2,M} - S_{\mathrm{loc}}^M \right| \leq (2\eta + \eta^2) S_{\mathrm{loc}}^M. \tag{29}$$

Proof. Write $a_t = \bar{G}_t^M - 1$ and $b_t = \bar{G}_t^M (\Lambda_{t,T}^M - 1)$. Since $\bar{G}_t^M \Lambda_{t,T}^M - 1 = a_t + b_t$, Theorem 1 and Proposition 2 give

$$\sigma_{Z,T}^{2,M} - S_{\mathrm{loc}}^M = \sum_{t=1}^{T} \widetilde{\Pi}_t^M (2 a_t b_t + b_t^2).$$

By Cauchy–Schwarz across both t and the prediction laws,

$$\left| \sum_{t=1}^{T} \widetilde{\Pi}_t^M (a_t b_t) \right| \leq \left(\sum_{t=1}^{T} \widetilde{\Pi}_t^M (a_t^2) \right)^{1/2} \left(\sum_{t=1}^{T} \widetilde{\Pi}_t^M (b_t^2) \right)^{1/2} \leq \sqrt{S_{\mathrm{loc}}^M} \sqrt{\eta^2 S_{\mathrm{loc}}^M} = \eta S_{\mathrm{loc}}^M,$$

and the assumed bound (28) controls the b_t^2 sum by $\eta^2 S_{\mathrm{loc}}^M$. □

Condition (28) is a CLT-level stability condition, not a consequence of taking K large. It holds when the normalized future factor $\Lambda_{t,T}^M$ is close to one in the weighted $L^2(\widetilde{\Pi}_t^M)$ sense, meaning that earlier prefix errors have weak predictive power for later corrections.

B.2. Comparing the three Algorithms

By Proposition 2, the per-step variance $R_t + D_t^{\hat{\psi}}/M$ depends on two design choices: (i) the *twist* $\hat{\psi}_t$, which controls both R_t and $D_t^{\hat{\psi}}$; and (ii) the *inner proposal mechanism*, which affects $D_t^{\hat{\psi}}$ through the effective proposal distribution.

We compare the three algorithms by decomposing their variance differences into inner-proposal and twist-quality contributions. All statements below assume the nested SMC framework of Section B.1, with K outer particles and M inner samples per step.

B.2.1. Take 1 vs. Take 2: inner proposal comparison

Takes 1 and 2 share the same twist $\hat{\psi}_t$, hence the same resampling term R_t. Their variance difference lies entirely in the inner IS noise D_t/M.

Take 1 (Algorithm 1) draws M inner samples from the conditional SDE (5) with learned score $\hat{o}_s$, producing $u_t^j \sim q_1(\cdot|x_{t-1})$, and weights by $\hat{\psi}_t$. By Proposition 2, the inner IS noise is

$$D_t^{(1)} = \mathbb{E}_{\mathbb{Q}_{t-1}} \left[h_t^2 \, \chi^2(q_t^* \| q_1(\cdot|x_{t-1})) \right]. \tag{30}$$

When $\hat{o}_s = o_s$ exactly, $q_1 = p^{\mathrm{ref}}(\cdot|x_{t-1})$ and this recovers $D_t^{\hat{\psi}}$ from Proposition 2.

Take 2 (Algorithm 2) generates M' composition SDE trajectories, FKC-resamples them to M particles in Stage 1, then applies twist-based IS in Stage 2 (identically to Take 1 but with effective proposal q_2 in place of p^{ref}):

$$D_t^{(2)} = \mathbb{E}_{\mathbb{Q}_{t-1}} \left[h_t^2 \, \chi^2(q_t^* \| q_2(\cdot|x_{t-1})) \right], \tag{31}$$

where $q_2(\cdot|x_{t-1})$ denotes the effective proposal distribution from Stage 1's FKC-weighted self-normalized resampling.

Directly comparing $\chi^2(q_t^*\|q_1)$ and $\chi^2(q_t^*\|q_2)$ is difficult since $q_t^* \propto p^{\mathrm{ref}}\hat{\psi}_t$ depends on the twist. However, since both share the same twist $\hat{\psi}_t$, we can reduce the comparison to a more tractable proxy. If the normalized twist ratio satisfies $\hat{\psi}_t(x_{1:t})/Z_t^{\hat{\psi}}(x_{1:t-1}) \le C_\psi$ pointwise, then for either proposal $q_i \in \{q_1, q_2\}$,

$$1 + \chi^2\big(q_t^*(\cdot|x_{1:t-1}) \,\big\|\, q_i(\cdot|x_{t-1})\big) \le C_\psi^2 \left(1 + \chi^2\big(p^{\mathrm{ref}}(\cdot|x_{t-1}) \,\big\|\, q_i(\cdot|x_{t-1})\big)\right), \tag{32}$$

since $(q_t^*)^2/q_i = (\hat{\psi}_t/Z_t^{\hat{\psi}})^2\,(p^{\mathrm{ref}})^2/q_i \le C_\psi^2\,(p^{\mathrm{ref}})^2/q_i$. With the same constant C_ψ and the same h_t in both $D_t^{(1)}$ and $D_t^{(2)}$, the ordering of $\chi^2(p^{\mathrm{ref}}\|q_i)$ determines the ordering of $D_t^{(i)}$. We therefore bound $\chi^2(p^{\mathrm{ref}}\|q_i)$ for each algorithm below. Define the score-error fields

$$\varepsilon_c^s(z; x_{t-1}) := \hat{o}_s(z, x_{t-1}) - o_s(z, x_{t-1}), \qquad \varepsilon_m^s(z) := \hat{j}_s(z) - j_s(z), \qquad \varepsilon_{\mathrm{eq}}(z) := \hat{j}_0(z) - j_0(z).$$

Lemma 7 (Score-error bounds on inner proposal quality)**.** *Let $P^{(1)}$ and $P^{(2)}$ denote the path measures of the conditional reverse SDE (5) and composition SDE (8) respectively. Assume the score errors are small enough that the squared Girsanov likelihood ratio $L_t := (\mathrm{d}P^{(\ell)}/\mathrm{d}\hat{P}^{(\ell)}|_{\mathcal{F}_t})^2$ satisfies $L_t \approx 1$ along typical paths (the precise bound with pointwise-sup errors is given in the proof). Then, to leading order,*

(a) *The conditional SDE (5) with learned score $\hat{o}_s$ generates terminal marginal $q_1(\cdot|x_{t-1})$ satisfying*

$$\chi^2\big(p^{\mathrm{ref}}(\cdot|x_{t-1}) \,\big\|\, q_1(\cdot|x_{t-1})\big) \lesssim C_{\mathrm{init}}^{(1)} + \int_0^S \mathbb{E}_{p_s(\cdot|x_{t-1})}\big[\|\varepsilon_c^s(Z_s; x_{t-1})\|^2\big]\,\mathrm{d}s =: E_{\mathrm{cond}}, \tag{33}$$

where $C_{\mathrm{init}}^{(1)} := \chi^2\big(p_S(\cdot|x_{t-1}) \,\big\|\, \mathcal{N}(0, I)\big)$ is the initialization mismatch.

(b) *The composition SDE (8) with FKC reweighting (9) generates, after self-normalized resampling with M' particles, an effective proposal $q_2(\cdot|x_{t-1})$ satisfying*

$$\begin{aligned}\chi^2\big(p^{\mathrm{ref}}(\cdot|x_{t-1}) \,\big\|\, q_2(\cdot|x_{t-1})\big) \lesssim C_{\mathrm{init}}^{(2)} + \int_0^S \mathbb{E}_{\bar{p}^s(\cdot|x_{t-1})}\big[\|\varepsilon_c^s(Z_s; x_{t-1}) - \varepsilon_m^s(Z_s) + \tfrac{1}{2}\varepsilon_{\mathrm{eq}}(Z_s)\|^2\big]\,\mathrm{d}s \\ + O(1/M') =: E_{\mathrm{comp}},\end{aligned} \tag{34}$$

where $\bar{p}^s$ is the composition probability path (7), $C_{\mathrm{init}}^{(2)} := \chi^2(\bar{p}_S(\cdot|x_{t-1})\|\mathcal{N}(0, I))$, and the $O(1/M')$ term is the self-normalized resampling bias from Stage 1.

Proof. **Part (a).** Let P denote the path measure of the conditional reverse SDE with exact score o_s and Q the path measure with learned score $\hat{o}_s = o_s + \varepsilon_c^s$. The drift difference is $\delta b_\tau = \varepsilon_c^{S-\tau}(Z_\tau; x_{t-1})$. By Girsanov's theorem, the Radon–Nikodym derivative of the path measures is

$$\left.\frac{\mathrm{d}P}{\mathrm{d}Q}\right|_{\mathcal{F}_S} = \exp\left(\int_0^S \langle \varepsilon_c^{S-\tau}, \mathrm{d}W_\tau\rangle - \tfrac{1}{2}\int_0^S \|\varepsilon_c^{S-\tau}\|^2\,\mathrm{d}\tau\right).$$

By the data-processing inequality, $\chi^2(p^{\mathrm{ref}}(\cdot|x_{t-1})\|q_1(\cdot|x_{t-1})) \le \chi^2(P\|Q)$. Now to bound $\chi^2(P\|Q) = \mathbb{E}_Q[(\mathrm{d}P/\mathrm{d}Q)^2] - 1$, define the squared likelihood-ratio process $L_t := (\mathrm{d}P/\mathrm{d}Q|_{\mathcal{F}_t})^2$. By Itô's formula,

$$\mathrm{d}L_t = L_t\big(2\langle \varepsilon_c^{S-t},\, \mathrm{d}W_t\rangle + \|\varepsilon_c^{S-t}\|^2\,\mathrm{d}t\big),$$

so $m(t) := \mathbb{E}_Q[L_t]$ satisfies $m'(t) = \mathbb{E}_Q[L_t\,\|\varepsilon_c^{S-t}(Z_t; x_{t-1})\|^2]$, with $m(0) = 1$, as the first part has mean zero. Under a pointwise bound $\sup_x \|\varepsilon_c^s(x; x_{t-1})\|^2 \le \beta(s)$, this gives $m'(t) \le \beta(S-t)\,m(t)$, and Grönwall's inequality yields the bound $\chi^2(P\|Q) = m(S) - 1 \le \exp(\int_0^S \beta(s)\,\mathrm{d}s) - 1$. In the small-error regime where $L_t \approx 1$, we can instead approximate $m'(t) \approx \mathbb{E}_{p_s}[\|\varepsilon_c^s\|^2]$ and integrate to get:

$$\chi^2(P\|Q) = m(S) - 1 \approx \int_0^S \mathbb{E}_{p_s}[\|\varepsilon_c^s\|^2]\,\mathrm{d}s.$$

The initialization mismatch $C_{\text{init}}^{(1)}$ accounts for the gap between $p_S(\cdot|x_{t-1})$ and the Gaussian initialization $\mathcal{N}(0, I)$, and is absorbed via the chain rule $\chi^2(P\|Q) \le (1 + C_{\text{init}}^{(1)})(1 + \chi^2_{\text{drift}}) - 1$.

Part (b). The composition SDE (8) has drift $b_\tau = o_{S-\tau} - j_{S-\tau} + \frac{1}{2}j_0 + \frac{1}{2}x$. With learned scores, the drift becomes $\hat{b}_\tau = \hat{o}_{S-\tau} - \hat{j}_{S-\tau} + \frac{1}{2}\hat{j}_0 + \frac{1}{2}x$, giving drift error $\delta b_\tau = \varepsilon_c^{S-\tau} - \varepsilon_m^{S-\tau} + \frac{1}{2}\varepsilon_{\text{eq}}$. Under exact scores, the composition SDE with FKC weight (9) targets $p^{\text{ref}}(\cdot|x_{t-1})$ (Lemma 2). There are two error sources when using approximate scores: (i) the drift error changes the path law from P_{comp} (exact scores) to $\hat{P}_{\text{comp}}$ (learned scores); (ii) the FKC weight (9) is evaluated with approximate scores.

For (i), the same argument as Part (a), applied to the composition drift error, gives

$$\chi^2(P_{\text{comp}}\|\hat{P}_{\text{comp}}) \approx \int_0^S \mathbb{E}_{\bar{p}^s}\big[\|\varepsilon_c^s - \varepsilon_m^s + \tfrac{1}{2}\varepsilon_{\text{eq}}\|^2\big]\,\mathrm{d}s.$$

For (ii), the FKC log-weight (9) is a $\mathrm{d}\tau$ integral of inner products of the scores j_s, j_0, o_s (writing $s = S - \tau$). Under approximate scores the pathwise log-weight error is

$$\log\frac{\hat{G}}{G} = \int_0^S \big[\langle\hat{j}_s, \hat{j}_s - \hat{o}_s\rangle - \langle j_s, j_s - o_s\rangle + \tfrac{1}{2}\langle\hat{j}_0, X_\tau + \hat{o}_s - \hat{j}_s\rangle - \tfrac{1}{2}\langle j_0, X_\tau + o_s - j_s\rangle\big]\,\mathrm{d}\tau.$$

Expanding, the integrand consists of bilinear terms pairing score errors $(\varepsilon_m^s, \varepsilon_c^s, \varepsilon_{\text{eq}})$ with the true scores, plus quadratic terms in the errors themselves (e.g., $\langle\varepsilon_m^s, \varepsilon_m^s - \varepsilon_c^s\rangle$). Assuming the true scores remain bounded along the SDE path, the magnitude $|\log(\hat{G}/G)|$ is controlled by the same integrated score errors that enter (i).

Combining both sources via data processing on the reweighted terminal marginals yields (34), after accounting for the self-normalized resampling in Stage 1 that adds $O(1/M')$ bias. □

When does Take 2 improve over Take 1? Define the integrated score errors $E_c := \int_0^S \mathbb{E}[\|\varepsilon_c^s\|^2]\,\mathrm{d}s$, $E_m := \int_0^S \mathbb{E}[\|\varepsilon_m^s\|^2]\,\mathrm{d}s$, $E_{\text{eq}} := \int_0^S \mathbb{E}[\|\varepsilon_{\text{eq}}\|^2]\,\mathrm{d}s$, and $E_{c\text{-}m} := \int_0^S \mathbb{E}[\|\varepsilon_c^s - \varepsilon_m^s\|^2]\,\mathrm{d}s$. This comparison implicitly assumes that the conditional-score error averaged under the conditional path in (33) is comparable to the same error averaged under the composition path in (34).

Sufficient condition for Take 2 improvement: Assume $C_{\text{init}}^{(1)} = C_{\text{init}}^{(2)}$ (comparable initialization quality) and that the $O(1/M')$ resampling bias is negligible. Then $E_{\text{comp}} < E_{\text{cond}}$, i.e., Take 2's inner proposal is closer to p^{ref} than Take 1's, if and only if

$$\tfrac{1}{4}E_{\text{eq}} + E_{c\text{-}m} + \int_0^S \mathbb{E}[\langle\varepsilon_{\text{eq}},\, \varepsilon_c^s - \varepsilon_m^s\rangle]\,\mathrm{d}s < E_c. \tag{35}$$

In applications such as MD that we consider here, equilibrium samples from p_0 are often far more abundant than trajectory data, so both j_s and j_0 are trained on more data than o_s. This motivates the following: suppose $E_m, E_{\text{eq}} \ll E_c$ (marginal scores are much more accurate than the conditional score). Then the cross-term and equilibrium contributions are negligible, and the criterion simplifies to

$$E_{c\text{-}m} < E_c \tag{36}$$

Define $\rho := \mathrm{Cov}_{\text{int}}(\varepsilon_c, \varepsilon_m)/\sqrt{E_c E_m}$ (the integrated score-error correlation). Then (36) becomes

$$\rho > \frac{1}{2}\sqrt{\frac{E_m}{E_c}}\,. \tag{37}$$

When $E_m \ll E_c$, the right side is close to zero: even a small positive correlation between the conditional and marginal score errors suffices for Take 2 to improve over Take 1. This also agrees with our intuition that the benefit of take 2 over take 1 will be more substantial if the adjacent frames are close to independent (i.e., marginal score close to conditional score).

B.2.2. TAKE 1 VS. TAKE 3: TWIST QUALITY COMPARISON

Takes 1 and 3 both use the conditional SDE as their inner proposal (hence identical q_1), but employ different twists: Take 1 uses the Tweedie-based twist $\hat{\psi}^{(1)}$ and Take 3 uses a learned twist $\hat{\psi}^{(3)}$ trained via Lemma 5. By Proposition 2,

the local variance proxy satisfies

$$S_{\mathrm{loc}}^{M,(1)} - S_{\mathrm{loc}}^{M,(3)} = \sum_{t=1}^{T} \underbrace{\left[R_t^{(1)} - R_t^{(3)}\right]}_{\text{resampling difference}} + \frac{1}{M} \sum_{t=1}^{T} \underbrace{\left[D_t^{\hat{\psi}^{(1)}} - D_t^{\hat{\psi}^{(3)}}\right]}_{\text{inner IS difference}}. \tag{38}$$

A twist closer to the optimal ψ^* reduces R_t (by aligning successive twisted targets $\mathbb{Q}_t$ and $\mathbb{Q}_{t-1}$ at their $x_{1:t-1}$-marginals) but may increase $D_t^{\hat{\psi}}$ (since $q_t^* \propto p^{\mathrm{ref}}\hat{\psi}_t$ depends on the twist, and a stronger twist pulls q_t^* further from p^{ref}). We formalize these effects below, beginning with a general result showing that R_t is governed not by the pointwise error of $\hat{\psi}_t$ alone, but by the one-step Bellman consistency of the twist sequence.

Lemma 8 (Resampling variance from Bellman residual)**.** *For any positive twist sequence $\hat{\psi}_t$, define*

$$Z_t^{\hat{\psi}}(x_{1:t-1}) := \int p^{\mathrm{ref}}(x_t|x_{1:t-1})\hat{\psi}_t(x_{1:t})\, dx_t$$

and the log-Bellman residual

$$B_t^{\hat{\psi}}(x_{1:t-1}) := \log Z_t^{\hat{\psi}}(x_{1:t-1}) - \log \hat{\psi}_{t-1}(x_{1:t-1}). \tag{39}$$

Let $h_t(x_{1:t-1}) := \mathbb{Q}_t(x_{1:t-1})/\mathbb{Q}_{t-1}(x_{1:t-1})$ be the density ratio of the successive prefix marginals. Then

$$h_t(x_{1:t-1}) = \frac{\exp(B_t^{\hat{\psi}}(x_{1:t-1}))}{\mathbb{E}_{\mathbb{Q}_{t-1}}[\exp(B_t^{\hat{\psi}})]}, \tag{40}$$

and hence

$$R_t = \mathrm{Var}_{\mathbb{Q}_{t-1}}\left(\frac{\exp(B_t^{\hat{\psi}})}{\mathbb{E}_{\mathbb{Q}_{t-1}}[\exp(B_t^{\hat{\psi}})]}\right). \tag{41}$$

If $\|B_t^{\hat{\psi}}\|_\infty \le \delta \ll 1$, then

$$R_t = \mathrm{Var}_{\mathbb{Q}_{t-1}}\left(B_t^{\hat{\psi}}\right) + O(\delta^3). \tag{42}$$

Moreover, if $\hat{\psi}_t = \psi_t^* e^{\epsilon_t}$, where ψ^* is the optimal twist and $q_t^{*,\mathrm{true}} \propto p^{\mathrm{ref}}\psi_t^*$ is the optimal proposal, then

$$\begin{aligned} B_t^{\hat{\psi}} &= \log \mathbb{E}_{q_t^{*,\mathrm{true}}}\left[e^{\epsilon_t(x_{1:t})} \mid x_{1:t-1}\right] - \epsilon_{t-1}(x_{1:t-1}) \\ &= \bar{\epsilon}_t - \epsilon_{t-1} + O(\|\epsilon_t\|_\infty^2), \end{aligned} \tag{43}$$

where $\bar{\epsilon}_t := \mathbb{E}_{q_t^{*,\mathrm{true}}}[\epsilon_t \mid x_{1:t-1}]$.

Proof. From the proof of Proposition 2,

$$h_t(x_{1:t-1}) = c_t\, \frac{Z_t^{\hat{\psi}}(x_{1:t-1})}{\hat{\psi}_{t-1}(x_{1:t-1})} = c_t\, \exp(B_t^{\hat{\psi}}(x_{1:t-1})),$$

where $c_t = Z_{t-1}^{\hat{\psi},\mathrm{glob}}/Z_t^{\hat{\psi},\mathrm{glob}}$ is independent of $x_{1:t-1}$. Since h_t is a density ratio, $\mathbb{E}_{\mathbb{Q}_{t-1}}[h_t] = 1$, and therefore $c_t = 1/\mathbb{E}_{\mathbb{Q}_{t-1}}[\exp(B_t^{\hat{\psi}})]$. This proves (40). (41) now follows from $R_t = \mathrm{Var}_{\mathbb{Q}_{t-1}}(h_t)$.

For the linearization, write $B = B_t^{\hat{\psi}}$ and $\mu = \mathbb{E}_{\mathbb{Q}_{t-1}}[B]$. Uniform boundedness on B gives

$$\frac{e^B}{\mathbb{E}[e^B]} = 1 + (B - \mu) + O(\delta^2),$$

so its variance is $\mathrm{Var}_{\mathbb{Q}_{t-1}}(B) + O(\delta^3)$ as claimed in (42). Finally, if $\hat{\psi}_t = \psi_t^* e^{\epsilon_t}$, then since

$$Z_t^{\hat{\psi}}(x_{1:t-1}) = \int p^{\mathrm{ref}}(x_t|x_{1:t-1})\, \psi_t^*(x_{1:t})\, e^{\epsilon_t(x_{1:t})}\, dx_t.$$

and $q_t^{*,\mathrm{true}}(\cdot|x_{1:t-1}) = p^{\mathrm{ref}}(\cdot|x_{1:t-1})\psi_t^*(x_{1:t})/\psi_{t-1}^*(x_{1:t-1})$ by definition, we have

$$Z_t^{\hat{\psi}}(x_{1:t-1}) = \psi_{t-1}^*(x_{1:t-1})\mathbb{E}_{q_t^{*,\mathrm{true}}}\left[e^{\epsilon_t(x_{1:t})} \mid x_{1:t-1}\right],$$

and substituting this identity into (39) gives the first equality in (43). The second equality follows from $\log \mathbb{E}[e^{\epsilon_t}] = \mathbb{E}[\epsilon_t] + O(\|\epsilon_t\|_\infty^2)$ by a Taylor expansion. □

Comparing Take 1 and Take 3 twist errors. The linearized decomposition (43) shows that R_t depends on the *incremental* twist error $\bar{\epsilon}_t - \epsilon_{t-1}$, not on ϵ_t alone. We now analyze this quantity for each twist estimator.

Take 1 (Tweedie twist). At each step t, the twist is estimated by $\hat{\psi}_t^{(1)}(x_{1:t}) = \frac{1}{N}\sum_{n=1}^N r(\hat{x}_{t+1:T}^{(n)})$ from N independent Tweedie rollouts approximately sampling from $p^{\mathrm{ref}}(\cdot|x_{1:t})$, each generating a trajectory $\hat{x}_{t+1:T}^{(n)}$ by iteratively denoising from $x_{1:t}$. With

$$R_t^{(1)} \approx \mathrm{Var}_{\mathbb{Q}_{t-1}^{(1)}}\left(\bar{\epsilon}_t^{(1)} - \epsilon_{t-1}^{(1)}\right), \tag{44}$$

even if the rollout estimates are individually accurate, their errors at times t and $t-1$ need not cancel because Take 1 does not explicitly enforce the one-step Bellman relation $\hat{\psi}_{t-1} \approx \mathbb{E}_{p^{\mathrm{ref}}}[\hat{\psi}_t \mid x_{1:t-1}]$. In fact since it estimates $\hat{\psi}_t^{(1)}$ separately at each time, the variance will be approximately additive.

Take 3 (learned twist). Take 3 fits a single twist family across times. If the learned twist has multiplicative Bellman residual

$$\hat{\psi}_{t-1}^{(3)}(x_{1:t-1}) = Z_t^{\hat{\psi}^{(3)}}(x_{1:t-1})\,\exp\left(\zeta_{t-1}^{(3)}(x_{1:t-1})\right), \tag{45}$$

then $B_t^{\hat{\psi}^{(3)}} = -\zeta_{t-1}^{(3)}$ exactly, and

$$R_t^{(3)} \approx \mathrm{Var}_{\mathbb{Q}_{t-1}^{(3)}}\left(\zeta_{t-1}^{(3)}\right). \tag{46}$$

The TD objective directly targets this Bellman consistency relation; the MC and KL objectives instead fit the same intermediate twist family by regression or variational projection, and reduce R_t to the extent that they produce a temporally coherent twist sequence.

These together give a precise account of when Take 3 reduces resampling variance relative to Take 1. The linearized resampling variance for Take 1 is $R_t^{(1)} \approx \mathrm{Var}(\bar{\epsilon}_t^{(1)}) + \mathrm{Var}(\epsilon_{t-1}^{(1)})$ (no cross-step cancellation), while for Take 3 it is $R_t^{(3)} \approx \mathrm{Var}(\zeta_{t-1}^{(3)})$ (only the function-class residual contributes). Take 3 helps when its learned twist has lower *Bellman-residual* variance than the rollout twist, and not necessarily a uniformly smaller pointwise twist error.

B.3. Practical diagnostics

The R_t vs. D_t trade-off. With the optimal twist ($\epsilon = 0$), $R_t = 0$, but $D_t^{\psi^*}$ is not zero in general. Since $h_t = 1$ under the optimal twist, we have

$$D_t^{\psi^*} = \mathbb{E}_{\mathbb{Q}_{t-1}^*}\left[\chi^2(q_t^{*,\mathrm{true}}(\cdot|x_{1:t-1}) \,\|\, p^{\mathrm{ref}}(\cdot|x_{1:t-1}))\right], \tag{47}$$

which measures how far the optimal proposal $q_t^{*,\mathrm{true}} \propto p^{\mathrm{ref}}\psi_t^*$ deviates from the prior p^{ref}. Conversely, with the trivial twist ($\hat{\psi}_t = 1$ for $t < T$, $\hat{\psi}_T = r$), we have $q_t^* = p^{\mathrm{ref}}$ at intermediate steps and $D_t = 0$, but all resampling variance concentrates at the terminal step: $R_t = 0$ for $t < T$ while $R_T = \chi^2\big(\pi(x_{1:T-1}) \,\|\, p^{\mathrm{ref}}(x_{1:T-1})\big)$. This is the bootstrap particle filter regime, where the entire difficulty of the problem appears as a single large weight step at the end. This reveals a fundamental trade-off: stronger twists reduce R_t but increase D_t. Since $\sigma_t^2 = R_t + D_t/M$, the number of inner samples M controls the trade-off: for large M, D_t/M is negligible and the optimal twist ($R_t = 0$) dominates; for small M (especially $M = 1$), D_t/M can dominate.

Both variance components admit approximate diagnostics from the SMC output. The inner chi-squared $\chi^2(q_t^*\|p^{\mathrm{ref}}|x_{1:t-1})$ at step t for particle k is estimated from the inner ESS: the inner IS weights are $\hat{\psi}_t(x_{1:t-1}, u_t^j)/\sum_l \hat{\psi}_t(x_{1:t-1}, u_t^l)$, and $\mathbb{E}[\mathrm{ESS}_t/M] \to 1/(1+\chi^2(q_t^*\|p^{\mathrm{ref}}))$ as $M \to \infty$. Thus $\chi^2(q_t^*\|p^{\mathrm{ref}}) \approx M/\mathrm{ESS}_t - 1$, providing a computable proxy for the inner IS difficulty. The empirical variance of the normalized outer weights estimates the *total* per-step variance contribution $R_t + D_t/M$. Therefore R_t can be inferred after combining the outer-weight variance with an estimate of the inner component derived from the inner ESS proxy above. Together, these diagnostics identify the dominant error source at each step and inform the choice of the algorithms.

C. Additional Experiments

This appendix collects three kinds of supplementary empirical support: synthetic controls and the conditional-generation benchmark (Section C.1), two biophysical benchmarks (Section C.2), and details for the path-dependent benchmark together with some informative diagnostics. All experiments are done on a Macbook Pro with M1 chip and 8GB memory.

C.1. Synthetic controls and conditional generation

C.1.1. 1D DOUBLE-WELL

We use the one-dimensional double-well potential $U(x) = (x^2 - 1)^2$ with overdamped Langevin as reference dynamics, time step $\Delta t = 0.05$, horizon $T = 10$, and initial state $x_0 = -1$. The terminal reward is

$$r(x_{1:T}) = \exp(-3(x_T - 1)^2).$$

The learned-score backend uses a reverse-SDE proposal with $S_{\max} = 2.0$ and 25 Euler–Maruyama steps. The score networks are 3-layer MLPs with hidden width 64 trained for 3,000 epochs. The Take 3 twist is trained from 5,000 reference trajectories. We use $K = 50$ particles, $M = 5$ inner samples, 15 trials for Bootstrap/Take 1/Take 3, and 10 trials for Take 2.

Table 4. In this simple short-horizon experiment, Bootstrap is closest to the learned-proposal target, while exact-drift Take 3 (MC) is the closest to the true-dynamics target with *reduced* variance.

Backend	Method	$\log \hat{Z}$	Mean ESS
Learned-proposal GT: -1.974. *True-dynamics GT:* -2.681.			
Learned-score	Bootstrap	-1.986 ± 0.337	46.1 ± 0.3
	Take 1 (Tweedie)	-2.169 ± 0.440	34.2 ± 2.6
	Take 2 (two-stage)	-2.520 ± 0.363	31.4 ± 2.9
	Take 3 (MC)	-2.124 ± 0.308	42.9 ± 2.1
	Take 3 (TD)	-3.023 ± 1.018	27.5 ± 1.5
Exact-drift	Bootstrap	-3.033 ± 0.442	45.4 ± 0.1
	Take 1 (Tweedie)	-2.752 ± 0.541	33.5 ± 2.7
	Take 3 (MC)	-2.704 ± 0.202	42.9 ± 1.3
	Take 3 (TD)	-3.116 ± 0.703	27.0 ± 1.6

This is primarily a calibration and variance-control experiment. Bootstrap matches the learned-proposal target well already, so the main effect of twisting is somewhat modest. Unlike the harder benchmarks, Take 1 and Take 3 are close for the exact-drift variant.

C.1.2. 2D MÜLLER–BROWN BARRIER CROSSING

In this experiment, the reference process is overdamped Langevin on the Müller–Brown potential with $\Delta t = 0.01$ and $T = 20$, starting from minimum A. The reward favors passing near the BC saddle at time $T/2$ and ending near minimum B:

$$r(x_{1:T}) = \exp(-3\|x_{T/2} - x_{\text{saddle}}\|^2 - 5\|x_T - x_B\|^2).$$

The 2 local minima A, B of the Müller-Brown potential define a rare-event transition. We use $K = 50$, $M = 12$, and the learned reverse-SDE proposal with $S_{\max} = 2.0$ and 25 diffusion steps. The score and twist networks use hidden width 256 and are trained for 5,000 epochs; the Take 3 twist is fit from 20,000 reference trajectories. We report 8 trials for Bootstrap/Take 1/Take 3 and 5 trials for Take 2. The exact-drift backend uses $-\nabla U$ of the potential as proposal instead of o_s, while still learning the twists.

This is the hardest synthetic experiment and is best interpreted as a proposal-mismatch diagnostic. The learned-score backend is proposal-limited, so better twists alone do not recover the true rare-event normalizing constant.

We also fine tune this experiment to test the Appendix B.2.1 Take 2 ρ-criterion (37). We run an *asymmetric-score* version for this same Müller–Brown task. The Take 2 composition drift combines a conditional score $o_s(z'|z)$ with a

Table 5. The result remains proposal-limited on the learned backend: Bootstrap stays closest to the learned-proposal target, while Take 2 is not stable. Under exact drift, Take 3 (KL) and Take 3 (MC) are the closest rows to the true-dynamics target with *significantly reduced* variance.

Backend	Method	$\log \hat{Z}$	Mean ESS
Learned-proposal GT: -5.273. *True-dynamics GT:* -7.038.			
Learned-score	Bootstrap	-4.965 ± 0.413	45.9 ± 0.1
	Take 1 (Tweedie)	-44.831 ± 1.847	12.3 ± 0.7
	Take 3 (KL)	-5.892 ± 0.622	39.4 ± 1.7
	Take 3 (MC)	-6.579 ± 0.976	40.1 ± 1.3
	Take 3 (TD)	-13.091 ± 2.126	41.5 ± 0.8
	Take 2 (two-stage, FKC)	-44.793 ± 2.651	10.7 ± 1.2
	Take 2 (no FKC)	-46.714 ± 3.917	13.0 ± 0.7
Exact-drift	Bootstrap	-12.917 ± 4.050	45.9 ± 0.3
	Take 1 (Tweedie)	-58.560 ± 6.384	12.1 ± 0.7
	Take 3 (KL)	-7.319 ± 0.558	40.5 ± 1.5
	Take 3 (MC)	-7.505 ± 0.829	34.8 ± 2.8
	Take 3 (TD)	-12.342 ± 3.478	42.6 ± 0.4

marginal score $j_s(z')$. This diagnostic trains the marginal score deliberately more accurately than the conditional score: j_s uses more equilibrium samples, a larger network, and a longer training schedule, whereas o_s uses fewer adjacent transition pairs, a smaller network and a shorter schedule. This creates the regime $E_m < E_c$, which is the regime in which Take 2 could plausibly help. The remaining question is whether the two score errors cancel in the composition drift. On paired transition samples we estimate

$$E_m = \mathbb{E}\|\epsilon_m(z_{t+1})\|^2, \qquad E_c = \mathbb{E}\|\epsilon_c(z_t, z_{t+1})\|^2, \qquad \rho_{\text{proxy}} = \frac{\mathbb{E}\langle \epsilon_c, \epsilon_m \rangle}{\sqrt{E_c E_m}},$$

where ϵ_m is the marginal-score error against the exact Boltzmann score $-\nabla U(z_{t+1})$, and ϵ_c is the conditional-score error against the exact Euler transition score for $p^{\text{ref}}(z_{t+1}|z_t)$. This is only a single-step proxy for the integrated condition in (37), but it directly checks the cancellation mechanism that Take 2 relies on.

Table 6. Asymmetric score error diagnostic. Although the marginal score is more accurate than the conditional score, the measured correlation proxy still falls below the ρ-threshold (37), so the simplified Take 2 improvement criterion predicts no gain.

Marginal rel. MSE	Conditional rel. MSE	ρ_{proxy}	Threshold	Criterion satisfied?
0.452	0.859	0.497	0.544	No

Table 6 shows that we are operating in the regime where Take 2 is not expected to improve. Empirically it does not: Bootstrap remains close to the weak learned-proposal target (-10.652 ± 3.37 versus GT -4.209), while Take 2 remain far off (-37.7 ± 2.803). We regard this as a useful negative validation of the Take 2 criterion.

C.1.3. Upsampling and Inpainting

In this experiment, we use the 2D coupled double-well

$$U(x, y) = (x^2 - 1)^2 + y^2 + 0.5xy$$

with $\Delta t = 0.02$ and horizon $T = 20$. To avoid overloading notation, write the generated latent state as $z_t = (z_t^{(1)}, z_t^{(2)}) \in \mathbb{R}^2$ and write $y_t \in \mathbb{R}^2$ for a fixed reference frame from an exact-dynamics trajectory sampled from the right well. The conditional-generation tasks are defined by soft observation likelihoods comparing the generated state z_t with the observed reference values y_t:

- **Upsampling**: observe the full 2D frame every second time step, with likelihood $\exp(-8\|z_t - y_t\|^2)$ at observed times. The missing intermediate frames are inferred.

- **Inpainting**: observe only the first coordinate at every time, with likelihood $\exp(-15(z_t^{(1)} - y_t^{(1)})^2)$. The second coordinate is unobserved and must be inferred from the dynamics and the observed first coordinate.

We use $K = 50$, $M = 8$, and 5 trials. The score models use hidden width 128 and 3,000 epochs; the Take 3 twist suite uses hidden width 256, 5,000 epochs, and 20,000 reference trajectories.

Table 7. The exact-drift rows remain the clean comparison of twists because they remove proposal mismatch. The learned backend is retained here only as a diagnostic: the severe learned vs. true ground-truth gap in upsampling shows that proposal mismatch dominates in that setting. ESS is included as an SMC stability diagnostic, while MSE is the pooled posterior-mean reconstruction error.

Task	Backend	Method	$\log \hat{Z}$	Mean ESS	Pooled MSE
Upsampling. Learned-proposal GT: -39.023. *True-dynamics GT:* -16.636.					
Upsampling	Exact-drift	Bootstrap	-16.901 ± 1.497	35.1 ± 0.4	0.045
		Take 3 (KL)	-16.242 ± 0.689	33.1 ± 0.4	0.043
		Take 3 (MC)	-15.342 ± 0.495	40.0 ± 0.3	0.041
		Take 3 (TD)	-15.791 ± 0.816	26.8 ± 1.3	0.054
	Learned-score	Bootstrap	-21.685 ± 1.214	31.0 ± 0.2	0.047
		Take 3 (KL)	-22.386 ± 0.809	23.6 ± 1.1	0.054
		Take 3 (MC)	-21.412 ± 0.412	36.9 ± 0.4	0.029
		Take 3 (TD)	-23.327 ± 2.251	21.1 ± 1.3	0.049
Inpainting. Learned-proposal GT: -14.739. *True-dynamics GT:* -12.775.					
Inpainting	Exact-drift	Bootstrap	-12.956 ± 0.413	35.1 ± 0.2	1.522
		Take 3 (KL)	-12.948 ± 0.259	44.3 ± 0.2	1.599
		Take 3 (MC)	-13.235 ± 0.486	43.1 ± 0.6	1.408
		Take 3 (TD)	-12.814 ± 1.420	40.7 ± 1.3	1.184
	Learned-score	Bootstrap	-16.580 ± 0.151	29.6 ± 0.3	2.780
		Take 1 (Tweedie)	-69.981 ± 4.063	6.6 ± 0.4	4.977
		Take 2 (two-stage)	-73.558 ± 4.083	6.9 ± 0.2	5.273
		Take 3 (KL)	-16.870 ± 0.325	41.2 ± 0.4	2.865
		Take 3 (MC)	-17.727 ± 0.451	41.4 ± 0.6	2.360
		Take 3 (TD)	-17.000 ± 0.659	38.7 ± 0.8	2.720

This experiment supports two distinct conclusions. First, on the exact-drift backend, Take 1 fails severely in this dense-conditioning regime, while Take 3 is well calibrated and also improves reconstruction quality. The best Take 3 objective depends on the metric: for exact-drift upsampling, MC gives the lowest pooled MSE (0.041 versus 0.043 for KL and 0.045 for Bootstrap), while for exact-drift inpainting TD gives the lowest pooled MSE (1.184 versus 1.408 for MC, and 1.522 for Bootstrap). Second, the learned backend is strongly proposal-limited, especially for upsampling: the learned-proposal target is -39.0 while the true-dynamics target is only -16.6. The learned-backend rows are useful mainly as a diagnostic showing that once the proposal is badly misspecified, neither good particle survival nor additional twist training is enough to recover the true conditional target.

C.2. Two biophysical experiments

We include details for the alanine dipeptide experiment (mentioned in Section 7.2) and consider a tetrapeptide transition-path-sampling benchmark inspired by MDGen (Jing et al., 2024) in this section.

Alanine dipeptide. We train a system-specific reference model on backbone torsion trajectories and evaluate both forward simulation and a midpoint+endpoint conditioning task from the C7eq basin to the α_R basin. This remains a within-system generalization test rather than a cross-system transfer benchmark. The main conclusions are:

- The learned proposal gives a reasonable forward model, with Ramachandran JSD 0.444 and TICA free-energy-surface JSD 0.360.
- On the primary midpoint+endpoint task, which has 14 held-out transition windows in the test split, naive diffusion essentially fails, with joint success rate 0.004, whereas Bootstrap and Take 3 reach 0.705 and 0.726 respectively. Relative to the empirical test-window target, Take 3 reduces the $\log \hat{Z}$ error from 2.159 to 1.282, improves

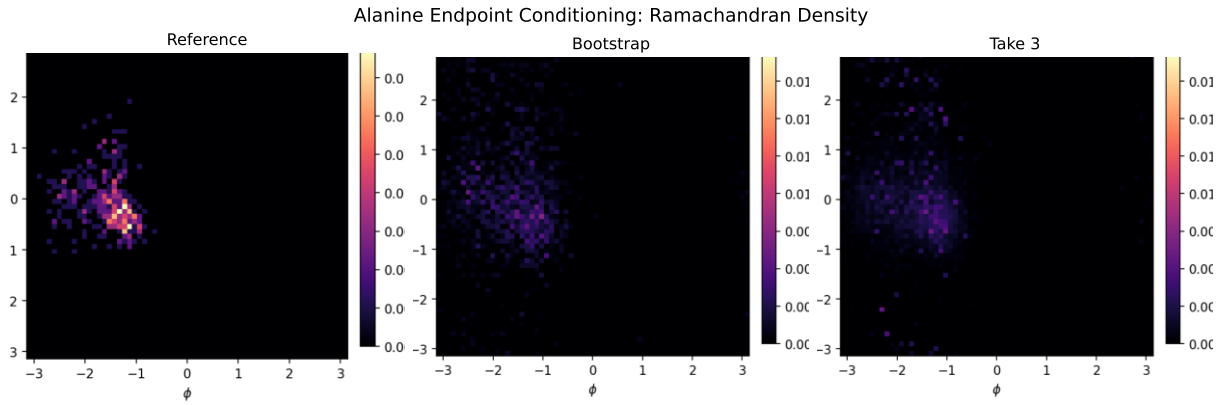

Figure 2. Alanine midpoint+endpoint conditioning in Ramachandran space. The reference transition windows occupy a compact region, while Bootstrap and Take 3 both broaden that distribution under the learned proposal. Take 3 places somewhat more mass in the high-density reference region than Bootstrap, consistent with the lower Ramachandran JSD reported in the table summary.

midpoint success from 0.784 to 0.858, and lowers both Ramachandran JSD ($0.709 \rightarrow 0.667$) and TICA JSD ($0.528 \rightarrow 0.502$). Endpoint-only success is slightly higher for Bootstrap (0.891 versus 0.852), so the gain is on the full conditioned transition rather than on terminal occupancy alone.

- A stricter midpoint+endpoint variant, with a 30-step horizon and tighter midpoint/terminal basin radii, leaves only 6 matching transition windows in the held-out split. Although this empirical reference is noisier, the qualitative trend is consistent: Take 3 again lowers the empirical $\log \hat{Z}$ error (from 0.717 to 0.628), improves success rate ($0.569 \rightarrow 0.667$), and improves both TICA JSD ($0.482 \rightarrow 0.441$) and Ramachandran JSD ($0.689 \rightarrow 0.677$).

We therefore regard alanine as a realistic molecular benchmark that shows a clear rare-event benefit for Take 3 in a moderate-rarity regime, while also making clear that the underlying proposal still somewhat limits perfect calibration. Figure 2 provides a qualitative view of the same endpoint-conditioning task in Ramachandran space.

Tetrapeptide transfer benchmark. We use the same tetrapeptide data as MDGen (Jing et al., 2024) for a cross-system transfer test on the reference proposal. Each trajectory is represented by padded backbone/side-chain torsion features $(\cos\theta, \sin\theta)$, together with a one-hot encoding of the four-residue sequence. For each held-out tetrapeptide, we train a single sequence-conditioned conditional score model on the other 49 peptides. The learned proposal therefore sees the peptide identity and a short local history of recent frames, but it is not trained on transition windows from the held-out peptide. The held-out trajectory is used only to define and evaluate the task: we fit a 10-state TICA/MSM discretization, choose a medium-difficulty start/end state pair, and score generated paths by a soft endpoint reward toward the selected end state. This is a reward-only transition-path-sampling problem.

The reported Take 3 trains the SMC twist as follows: after the transferable proposal is trained, we generate *unconditional* rollout windows on the held-out sequence and train the twist from their endpoint rewards. This uses the reward oracle and the learned proposal, but not real held-out transition windows for twist training. We compare it with naive diffusion, which estimates Z by plain proposal rollouts, and Bootstrap, which uses the same proposal with reward reweighting/resampling but no learned look-ahead twist. We report errors against two references as earlier: the *learned-proposal target*, estimated from many unconditional proposal rollouts, and the *empirical target*, estimated from held-out reference windows starting in the chosen start state.

Our result shows that:

- The trained Take 3 twist tracks the learned-proposal target more closely than both naive rollout Monte Carlo and Bootstrap, with mean absolute error 0.044 versus 0.071 for naive diffusion and 0.068 for Bootstrap. It also substantially reduces run-to-run variability, with reported $\log \hat{Z}$ standard deviation 0.104 versus 0.227 for naive diffusion and 0.198 for Bootstrap.

- Relative to the empirical reference-window target, Take 3 also has the smallest mean absolute error among the

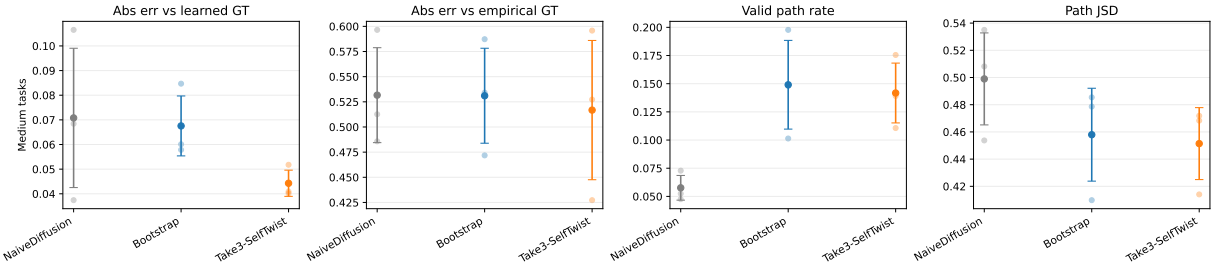

Figure 3. Transfer experiment over 15 held-out tetrapeptide task instances on a 50-peptide dataset. From left to right: mean absolute $\log Z$ error relative to the learned-proposal target, mean absolute $\log Z$ error relative to the empirical reference-window target, valid-path rate, and state-visitation JSD. Naive diffusion serves as a plain rollout Monte Carlo baseline, while Bootstrap and Take 3 additionally provide SMC-based normalizing-constant estimates. Take 3 improves learned-target calibration and reduces estimator variability, with small gains on the empirical-target and visitation metrics. Error bars show standard errors across held-out task instances.

methods, at 0.517 versus 0.532 for naive diffusion and 0.531 for Bootstrap.

- Path-quality metrics are mixed but slightly favor Take 3 on visitation fit. Naive diffusion is not competitive (valid-path rate 0.058, visitation JSD 0.499). Bootstrap and Take 3 have similar valid-path rates (0.149 versus 0.142), while Take 3 attains a slightly lower visitation JSD (0.451 versus 0.458).

Taken together, this experiment indicates that transfer is feasible in principle, while also making clear that cross-system proposal mismatch remains a major bottleneck. Figure 3 summarizes the same point: relative to naive diffusion and Bootstrap, Take 3 improves learned-target calibration and reduces estimator variability, with small but directionally consistent gains on the empirical-target and visitation metrics.

C.3. Detailed results for the path-dependent benchmark

This subsection gives the full setup and additional plots for the path-dependent benchmark in Section 7.3. To probe the setting where the optimal twist is not Markov in x_t alone, we use a synthetic benchmark on the 2D coupled double-well

$$U(x, y) = (x^2 - 1)^2 + y^2 + 0.5xy\,,$$

with time step $\Delta t = 0.02$, horizon $T = 36$, and initial state $z_0 = (-1, 0.25)$. Write the position as $z_t = (x_t, y_t)$. The target is the right well near $z_R = (1, -0.25)$, but the reward also depends on *which gate is hit first*. Concretely, we maintain a route-progress flag $m_t \in \{0, +1, -1\}$ that records whether the trajectory has first entered the upper gate

$$\{|x_t| \le 0.35,\ y_t \ge 0.45\}$$

or the lower gate

$$\{|x_t| \le 0.35,\ y_t \le -0.45\}.$$

The incremental log reward is

$$\log G_t(z_t, m_t) = 0.04\,\mathbf{1}\{m_t = +1\} - 0.04\,\mathbf{1}\{m_t = -1\}$$

for $t < T$, with terminal term

$$\log G_T(z_T, m_T) = 0.04\,\mathbf{1}\{m_T = +1\} - 0.04\,\mathbf{1}\{m_T = -1\} - 10\|z_T - z_R\|^2.$$

This reward is non-Markovian in z_t alone but becomes Markov after augmenting the state with m_t. We therefore learn Take 3 twists on the augmented state (z_t, m_t) and report only the exact-drift backend, so the comparison isolates the effect of path dependence rather than proposal mismatch. We use $K = 64$, $M = 12$, eight trials, reverse-diffusion hyperparameters $(S_{\max}, n_{\text{diff}}) = (1.5, 20)$, score networks with hidden width 192 trained for 3000 epochs, and twist networks with hidden width 256 trained for 3500 epochs on 12,000 reference trajectories.

Table 8. Non-Markovian route-history benchmark on the 2D coupled double-well. The reward depends on which gate is hit first, so the task is not Markov in x_t alone. Take 3 (KL) is the only method that is simultaneously close on $\log Z$, route split, and occupancy JSD.

Method	$\log \hat{Z}$	Mean ESS	Upper / Lower	Visitation JSD
Exact-proposal GT: $\log Z = -3.582$*, upper/lower/undecided* = $0.749/0.114/0.137$.				
Bootstrap	-4.230 ± 0.934	62.27 ± 0.02	$0.545/0.223$	0.566 ± 0.044
Terminal-only IS	-4.472 ± 1.107	2.07 ± 0.85	$0.541/0.051$	0.579 ± 0.032
Take 1 (Tweedie)	-69.076 ± 12.425	19.66 ± 2.56	$0.736/0.125$	0.648 ± 0.006
Take 3 (KL)	-3.760 ± 0.207	59.51 ± 0.44	$0.670/0.170$	0.356 ± 0.028
Take 3 (MC)	-9.585 ± 0.620	55.02 ± 1.18	$0.670/0.090$	0.424 ± 0.018
Take 3 (TD)	-10.648 ± 1.335	54.99 ± 1.16	$0.543/0.096$	0.475 ± 0.024

The resulting target is genuinely multi-route rather than concentrated on a single passage: under the exact proposal ground truth, the weighted route split is 0.749 upper-first, 0.114 lower-first, and 0.137 undecided (reach the horizon without hitting either gate), with joint endpoint-and-upper success probability 71%. Table 8 and Figure 1 show three consistent patterns:

- Terminal-only importance sampling collapses: its mean ESS is only 2.07, its $\log \hat{Z}$ error is nearly one, and it severely under-represents the lower route.
- Bootstrap is much more stable than terminal-only IS, but it still misses the target occupancy pattern and route split, with visitation JSD 0.566 and upper/lower rates $0.545/0.223$ versus the target $0.749/0.114$.
- Take 3 (KL) is the best overall method on this benchmark. It is closest to the exact proposal ground truth on $\log Z$ (-3.760 ± 0.207 versus GT -3.582), has the best visitation JSD (0.356), and recovers a route split ($0.670/0.170$) closer to the target than Bootstrap or terminal-only IS.

The MC and TD twist objectives are worse than KL in this example. These results support the claim: once the reward depends on the *history* rather than on x_t alone, terminal-only reweighting becomes unreliable, and the history-aware Take 3 twist is the most robust option among the methods.

Figure 4 shows a direct theory-validation diagnostic for this same benchmark using the practical Appendix B.3 estimators of R_t and D_t/M. To keep the comparison aligned with the decomposition, both methods are evaluated under the same exact-p^{ref} nested wrapper with the same inner budget $M = 12$, changing only the twist. The split-panel figure makes the mechanism easier to read: the top row shows the *non-terminal* contributions, while the bottom panel isolates the *terminal* step. Under the trivial twist used in Bootstrap, essentially all difficulty is deferred to the end: the non-terminal sums are negligible ($\sum_{t<T} R_t \approx 0.01$, $\sum_{t<T} D_t/M \approx 0.00$), but the terminal total is 17.51. Take 3 (KL) instead spreads moderate variance across earlier steps ($\sum_{t<T} R_t \approx 3.24$, $\sum_{t<T} D_t/M \approx 0.58$) while collapsing the terminal spike to 0.72. This is the qualitative behavior predicted by the theory: a stronger twist aligns the intermediate targets, prevents the bootstrap-style terminal blow-up, and yields a much smaller total variance budget overall.

D. Extension: controlled diffusion proposals

Takes 1-3 all use the prior $p^{\mathrm{ref}}(x_{t+1}|x_t)$ as the inner proposal kernel (Take 2's composition SDE targets the same transition through a different sampler). The twist $\hat{\psi}_t$ enters only through the SMC weights: it changes which particles survive, but not where the next proposals are generated. A natural extension is to incorporate reward information directly into the proposal by tilting the diffusion drift toward high-twist regions. This appendix formalizes this idea.

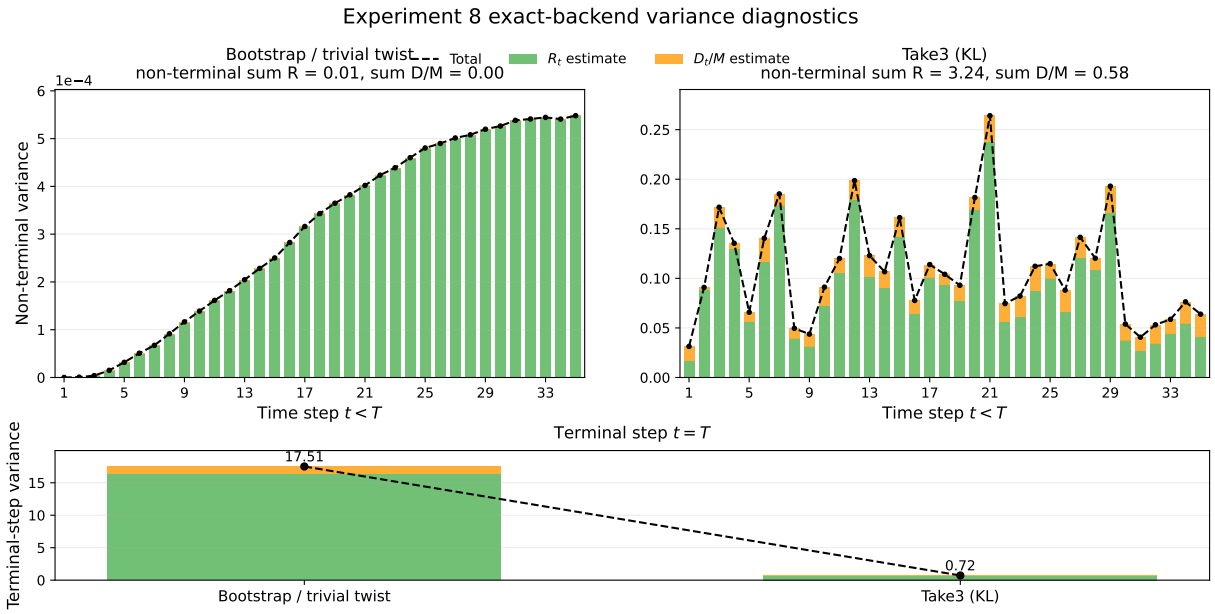

Figure 4. Practical variance decomposition under the exact-proposal backend. The top row shows the non-terminal contributions for the trivial twist and for Take 3 (KL), each on its own scale; the bottom panel compares the terminal-step totals directly. The trivial twist has almost no pre-terminal variance but a very large terminal spike, whereas Take 3 (KL) trades that for moderate earlier-step variance and a much smaller terminal contribution.

D.1. Controlled proposals and corrected SMC weights

Let $q_t^u(x_t|x_{1:t-1})$ be any proposal kernel, possibly history-dependent, obtained by applying a control u to the reference dynamics. The SMC target is unchanged: only the proposal changes. Therefore the incremental weight becomes

$$w_t^k = \frac{p^{\mathrm{ref}}(x_t^k|x_{t-1}^k)\,\hat{\psi}_t(x_{1:t}^k)}{q_t^u(x_t^k|x_{1:t-1}^k)\,\hat{\psi}_{t-1}(x_{1:t-1}^k)}. \tag{48}$$

When $q_t^u = p^{\mathrm{ref}}$, this reduces to the usual twisted-SMC weight from Takes 1 and 3. Thus controlled proposals do not change the target measure; they only change the variance of the importance weights.

For diffusion models with an Euler–Maruyama transition, a convenient controlled proposal is

$$x_t = x_{t-1} + \big(b(x_{t-1}) + \sigma^2 u_{t-1}(x_{1:t-1})\big)\Delta t + \sigma\sqrt{\Delta t}\,\varepsilon_t, \qquad \varepsilon_t \sim \mathcal{N}(0, I). \tag{49}$$

Here the control may depend on the full particle history $x_{1:t-1}$, as long as it is adapted and does not look into the future. The one-step density ratio in (48) is available in closed form:

$$\log \frac{p^{\mathrm{ref}}(x_t|x_{t-1})}{q_t^u(x_t|x_{1:t-1})} = \frac{\sigma^2\Delta t}{2}\|u_{t-1}\|^2 - u_{t-1}^\top\big(x_t - x_{t-1} - b(x_{t-1})\Delta t\big). \tag{50}$$

This is the discrete Gaussian form of the Girsanov correction. It is exact for the transition (49); if applied to a coarse learned transition that is not well approximated by a single Gaussian Euler step, it should instead be viewed as an approximation.

D.2. The locally optimal proposal

We next explain why the Doob transform is the relevant "locally optimal" proposal. Fix any positive twist sequence $\hat{\psi}_0, \ldots, \hat{\psi}_T$ with $\hat{\psi}_T = r$. The corresponding unnormalized intermediate target is

$$\pi_t^{\hat{\psi}}(x_{1:t}) := p^{\mathrm{ref}}(x_{1:t})\,\hat{\psi}_t(x_{1:t}).$$

Its normalized version is $\mathbb{Q}_t^{\hat{\psi}} = \pi_t^{\hat{\psi}} / Z_t^{\hat{\psi},\mathrm{glob}}$. Given a particle prefix $x_{1:t-1}$, the proposal that eliminates conditional weight variation at the next step is, as shown in Lemma 4:

$$q_t^{\hat{\psi},*}(x_t|x_{1:t-1}) := \frac{p^{\mathrm{ref}}(x_t|x_{t-1})\,\hat{\psi}_t(x_{1:t})}{Z_t^{\hat{\psi}}(x_{1:t-1})}, \qquad Z_t^{\hat{\psi}}(x_{1:t-1}) := \int p^{\mathrm{ref}}(x_t|x_{t-1})\hat{\psi}_t(x_{1:t})\,\mathrm{d}x_t. \tag{51}$$

Substituting (51) into (48) gives $w_t = Z_t^{\hat{\psi}}(x_{1:t-1})/\hat{\psi}_{t-1}(x_{1:t-1})$, which is independent of the proposed state x_t. This is the standard locally optimal proposal for the chosen twisted intermediate targets. For a generic learned twist $\hat{\psi}$, the density $q_t^{\hat{\psi},*}$ is available only up to the local normalizer $Z_t^{\hat{\psi}}(x_{1:t-1})$. Sampling exactly from this tilted transition, or evaluating its density inside an SMC correction, therefore requires an additional normalizing-constant computation. The nested sampler in Takes 1–3 avoids this by using an inner IS to construct an unbiased estimate of $Z_t^{\hat{\psi}}$. If the twist is the optimal twist ψ^*, however, the Bellman recursion from Lemma 5 gives $Z_t^{\psi^*}(x_{1:t-1}) = \psi_{t-1}^*(x_{1:t-1})$. The locally optimal proposal then becomes the Doob transform

$$q_t^{\mathrm{Doob}}(x_t|x_{1:t-1}) := \frac{p^{\mathrm{ref}}(x_t|x_{t-1})\,\psi_t^*(x_{1:t})}{\psi_{t-1}^*(x_{1:t-1})}. \tag{52}$$

Thus q_t^{Doob} is the exact zero-variance proposal associated with the optimal twist. Replacing ψ^* by $\hat{\psi}$ gives an approximate locally optimal proposal, but not the exact Doob transform.

Over a small interval Δt, the Doob proposal has density proportional to

$$\mathcal{N}(x_t;\, x_{t-1} + b(x_{t-1})\Delta t, \sigma^2 \Delta t I)\; \psi_t^*(x_{1:t}).$$

Taylor-expanding $\log \psi_t^*(x_{1:t})$ around x_{t-1} and completing the square shifts the Gaussian mean by $\sigma^2 \Delta t\, \nabla_{x_t} \log \psi_t^*(x_{1:t})$, yielding the drift $b + \sigma^2 \nabla \log \psi^*$ as approximation. This identifies the scalar object learned by Take 3 ψ_t^* with the "optimal control" in the proposal.

In the extension considered below, assuming the MD trajectory we are generating from has small Δt, we first use the existing reverse diffusion sampler (from Take 1 or 2) to generate a good proposal $y_t^k \sim p^{\mathrm{ref}}(\cdot|x_{t-1}^k)$, then with $u_t^k \leftarrow \nabla_{x_t} \log \hat{\psi}_t(x_{1:t-1}^k, y_t^k)$, apply a small one-step controlled move:

$$x_t^k \leftarrow y_t^k + \big(b_0(y_t^k, x_{t-1}^k) + \sigma^2 u_t^k\big)\eta + \sigma\sqrt{\eta}\,\varepsilon_t^k \quad , \varepsilon_t^k \sim \mathcal{N}(0, I)\,.$$

where b_0 is the last-step clean score drift. The outer SMC weight is then approximated by

$$w_t^k \leftarrow \exp\left[\frac{\sigma^2\eta}{2}\|u_t^k\|^2 - (u_t^k)^\top \left(x_t^k - y_t^k - b_0(y_t^k, x_{t-1}^k)\eta\right)\right] \frac{\hat{\psi}_t(x_{1:t}^k)}{\hat{\psi}_{t-1}(x_{1:t-1}^k)}$$

using (48) and (50).

D.3. Training the control

For completeness, we sketch how the controlled proposal could be trained. There are two natural variants.

Plug-in gradient control. If the learned twist is differentiable and stable, one can set $u_t(x_{1:t}) = \nabla_{x_t} \log \hat{\psi}_t(x_{1:t})$ as a control. This reuses the Take 3 twist and does not require additional learning.

Separate control network. Alternatively, one can learn a vector field $u_t^\phi(x_{1:t})$ separately from the twist. The control objective is to maximize the KL-regularized reward:

$$J(\phi) = \mathbb{E}_{Q^{u^\phi}}\left[\log r(x_{1:T}) - \sum_{t=0}^{T-1} \frac{\sigma^2 \Delta t}{2}\|u_t^\phi(x_{1:t})\|^2\right]. \tag{53}$$

Given the current control u^ϕ, the algorithm will run the SMC algorithm with proposals $q_t^{u^\phi}$ to collect weighted particles $\{(\bar{w}_t^k, x_{1:t}^k)\}_{t=1}^T$. The gradient $\nabla_\phi J$ above can be estimated from the SMC particles with a self-normalized policy gradient estimator and used for updating the control and iterate.

This extension connects our framework to variational Doob-transform and path-integral control methods (Domingo-Enrich et al., 2025; Havens et al., 2025; Levine, 2018; Uehara et al., 2025). We leave a full empirical study of controlled proposals to future work.

D.4. Variance trade-off against Takes 1–3

The variance decomposition in Theorem 1 separates two effects: the resampling term R_t, controlled by the twist sequence, and the inner proposal term $D_t^{\hat{\psi}}/M$, controlled by how well the proposal matches the locally optimal kernel $q_t^* \propto p^{\mathrm{ref}}\hat{\psi}_t$. Replacing p^{ref} by a controlled proposal changes the latter term to the same expression with $\chi^2(q_t^*(\cdot|x_{1:t-1}) \,\|\, q_t^u(\cdot|x_{1:t-1}))$ in place of $\chi^2(q_t^* \| p^{\mathrm{ref}})$. If u reproduces the Doob proposal, this chi-squared term vanishes. With the same twist, the resampling term R_t is unchanged; the gain is purely better proposal quality.

This gain comes with costs. First, it needs either $\nabla_{x_t} \log \hat{\psi}_t$ or a separately trained vector-field control, whereas Take 3 only evaluates the scalar twist. Second, exact correction requires a tractable proposal density ratio such as (50); this is straightforward for Gaussian Euler steps but less clean for coarse learned transition models. Therefore this method is most attractive when the reference proposal is very inefficient, the inner budget M is small, and the dynamics are represented at a time scale where the controlled-transition density ratio is accurate. In the regime studied in this paper, Take 3 is simpler and more robust: it keeps the reference proposal unchanged with the learned score, preserves exact SMC correction with minimal additional modeling assumptions, and reduces variance by learning the scalar twist.