TILTED SCHRÖDINGER BRIDGE MATCHING

Anonymous authors
Paper under double-blind review

ABSTRACT

Schrödinger bridges provide an entropy-regularized framework and a principled solution for unpaired domain translation. In practice, a pretrained bridge may need to be adapted to human preferences or physical constraints through a reward a problem closely related to reward tilting in diffusion models but underexplored for Schrödinger bridges. We introduce Tilted Schrödinger Bridge Matching (TSBM), a post-training method for fine-tuning a learned bridge P between source p_0 and target p_1 toward a reward-tilted target $p_1^r \propto p_1 e^r$, while preserving source p_0. We formulate this adaptation as alternating optimization initialized from P, provide theoretical justification, and derive a practical algorithm based on Adjoint Matching. We evaluate TSBM on unpaired image-to-image translation targeting digit properties in MNIST and facial attributes in CelebA.

1 INTRODUCTION

Schrödinger Bridges (SBs) seek a path law that minimizes relative entropy with respect to a reference process while matching prescribed endpoint distributions (Schrödinger, 1931). They connect stochastic control Bellman (1957), entropic optimal transport Léonard (2014), and diffusion-based generative modeling Song et al. (2021). SB methods learn *minimal energy* transport from a source distribution p_0 to a terminal distribution p_1, with applications including unpaired domain transfer (Shi et al., 2023; Gushchin et al., 2024a; Kholkin et al., 2026b), trajectory inference Tong et al. (2024) and sampling from Boltzmann distributions (Liu et al., 2025a; Tamogashev & Malkin, 2026).

After training, however, one may wish to adjust the trained model to account for human preferences Uehara et al. (2024), to enforce physical constraints Li et al. (2025); Chung et al. (2023a) or presence of image attributes Dhariwal & Nichol (2021). The standard way to induce such alignment is via diffusion model reward finetuning or reward tilting Domingo-Enrich et al. (2025), where the goal is to adjust the diffusion model output distribution from p_0 to $p_1^r(x) \propto p_1(x)e^{r(x)}$, where r is a real-valued differentiable reward function.

However, while there are a lot research for reward finetuning for Flow Matching Liu et al. (2025b) and Diffusion Models Domingo-Enrich et al. (2025); Black et al. (2024); Clark et al. (2024) the reward finetuning for Schrödinger Bridges is underexplored. For memoryless diffusion references Song et al. (2021), a terminal reward yields the desired tilt $p_1^r \propto p_1 e^r$. In diffusion bridges Liu et al. (2023), however, dependence between X_0 and X_1 generally biases the resulting terminal marginal. Existing reward-finetuning Li et al. (2025) and inference-time alignment Chung et al. (2023b) methods are approximate and introduce different biases, which may lead to artifacts or reward hacking.

In this work, we tackle the problem of **reward tilting the Schrödinger Bridge models**. Our contributions are:

1. **Theory.** We show that a Schrödinger bridge with target tilted by differentiable reward $p_1^r(x) \propto p_1(x)e^{r(x)}$ can be cast as stochastic optimal control relative to an exact pretrained bridge with target p_1 (§ 3), enabling finetuning to the new target while preserving the source marginal.
2. **Method.** We introduce Tilted Schrödinger Bridge Matching (TSBM), a finetuning procedure which learns an incremental control via alternating Controller and Corrector Matching (§ 4). We validate TSBM reward steering for reward tilted unpaired image translation on Colored MNIST and CelebA (§ 6) datasets.

Notation. We work in $\mathbb{R}^d$, which is the D-dimensional Euclidean space equipped with the Euclidean norm $\|\cdot\|$. We use Ω to denote the space of trajectories, i.e., continuous $\mathbb{R}^D$-valued functions of

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

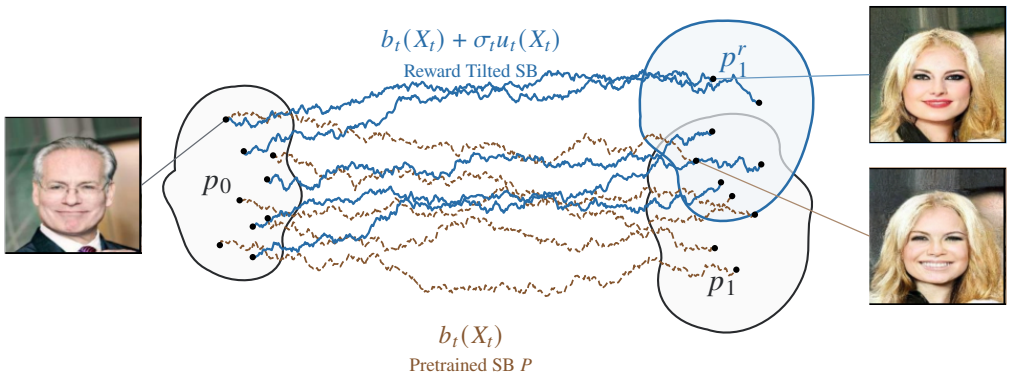


Figure 1: **Reward tilting a pretrained Schrödinger bridge.** The pretrained SB P transports p_0 to p_1 with drift b_t (brown). TSBM learns an incremental control u_t, resulting in drift $b_t + \sigma_t u_t$ (blue), to target the reward $r(x)$ tilted marginal, i.e., $p_1^r(x) \propto p_1(x)r(x)$.

$t \in [0, 1]$, then $X \in \Omega$ is the trajectory and $X_t \in \mathbb{R}^d$ is its slice at time t. We write $Q \in P(\Omega)$ to denote a path law Q, which is the element of probability distributions set on the trajectories Ω. For a path law Q, let $p_t^Q(x)$ denote the density of X_t and let $p_{t|s}^Q(y \mid x)$ denote the conditional density of X_t given $X_s = x$, for $0 \le s < t \le 1$. We assume that these distributions admit densities with respect to Lebesgue measure whenever the densities are used.

In the whole manuscript we set the reference process R, which we assume can be represented as a diffusion with globally Lipschitz in x_t drift function and continuous, finite and positive volatility coefficient σ_t:

$$R : \mathrm{d}X_t = f_t(X_t)\,\mathrm{d}t + \sigma_t\,\mathrm{d}W_t, \quad X_0 \sim p_0. \tag{1}$$

The diffusion R then can be augmented by controller $u_t : \mathbb{R}^d \times \mathbb{R}_+ \to \mathbb{R}^d$, and the *controlled diffusion* is denoted as R^u:

$$R^v : \mathrm{d}X_t = [f_t(X_t) + \sigma_t u_t(X_t)]\,\mathrm{d}t + \sigma_t\,\mathrm{d}W_t, \quad X_0 \sim p_0, \tag{2}$$

where W_t is standard Brownian motion.

2 Background

Stochastic optimal control. In our work we consider commonly used in generative modeling Stochastic optimal control (SOC) problem Bellman (1957) the quadratic cost control-affine problem formulation Domingo-Enrich et al. (2024). For the original reference R, our goal is add such a control u_t to it to minimize the following objective:

$$\begin{aligned} u^* = \min_v \mathbb{E}_{R^u}\left[\frac{1}{2}\int_0^1 \|u_t(X_t)\|^2\,\mathrm{d}t - r(X_1)\right] \\ \text{s.t. } \mathrm{d}X_t = \left[f_t(X_t) + \sigma_t u_t(X_t)\right]\mathrm{d}t + \sigma_t\,\mathrm{d}W_t, \qquad X_0 \sim p_0, \end{aligned} \tag{3}$$

where X_t is the process trajectory slice, r is a terminal reward, f_t is reference drift, σ_t is reference volatility, which are not optimized and u^* is called optimal controller.

Under the usual regularity assumptions the optimal distribution on endpoints has form:

$$p_{0,1}^{R^{v_g}}(X_1|X_0) = p_{0,1}^R(X_1|X_0)\exp[r(X_1) + V_0(X_0)]\,, V_0(x) = -\log \mathbb{E}_R\left[e^{r(X_1)} \mid X_0 = x\right], \tag{4}$$

when $p_{0,1}^R$ is memoryless, i.e., $p_{0,1}^R(X_0, X_1) = p_0^R(X_0)p_1^R(X_1)$ the SOC problem admits a particularly simple characterization: the terminal marginal of the controlled process satisfies $p_1^{R^{v_g}}(X_1) \propto p_1^R(X_1)r(X_1) = p_1^r(X_1)$. In contrast, for non-memoryless models, such as Diffusion Bridges Liu et al. (2023) or Schrödinger Bridges De Bortoli et al. (2021), this simplification no longer holds and the resulting terminal marginal in general does not coincide with the desired reward-tilted distribution p_1^r. We provide further discussion in Appendix B.1.

Controller Matching. In this work, we are focused on Adjoint Matching Domingo-Enrich et al. (2025) algorithm for solving SOC problems thanks to its scalability and robustness. Given a SOC problem Eq 3 one can define a "lean" adjoint state $\tilde{a}$:

$$\frac{\mathrm{d}\tilde{a}_t(X)}{\mathrm{d}t} = -\nabla_x f_t(X_t)^\top \tilde{a}_t(X), \quad \tilde{a}_1(X) = -\nabla_x r(X_1), \tag{5}$$

where $\nabla_x f_t(X_t)$ is the Jacobian of reference drift $f_t(X_t)$. Which in turn allows to define the optimal controller u^*, as the argmin of matching objective:

$$u^* = \arg\min_u \mathbb{E}_{X\sim P^{\overline{u}}}\left[\int_0^1 \|u(X_t,t) + \sigma_t \tilde{a}_t(X)\|^2 \,\mathrm{d}t\right], \quad \overline{u} = \text{stopgrad}(u), \tag{6}$$

where differentiation through the generation trajectory $X \sim P^{\overline{u}}$ may be ommitted, which makes algorithm scalable and popular in different applications Havens et al. (2025); Liu et al. (2025a); Gutjahr et al. (2026). Although the SDE simulation for sampling $X \sim P^{\overline{u}}$ and ODE simulation to calculate the "lean" adjoint $\tilde{a}_t$ are still required and can be rather expensive.

Schrödinger bridges. While regular SOC problem Eq 3 has only one marginal constraint, i.e., $X_0 \sim p_0$, the Schrödinger Bridge Problem imposes two marginal constraints on the endpoints, i.e., $X_0 \sim p_0$ and $X_1 \sim p_1$, lefts out the terminal reward $r(x)$, but keeps the $u_t(X_t, t)$ minimization term, which can be read as minimal energy constraint:

$$\begin{aligned} u^* = \arg\min_v \mathbb{E}_{R^u}\left[\frac{1}{2}\int_0^1 \|u_t(X_t)\|^2 \,\mathrm{d}t\right] &= \arg\min_v \mathrm{KL}(R^u \| R) \\ \text{s.t. } \mathrm{d}X_t = \big[f_t(X_t) + \sigma_t u_t(X_t)\big]\,\mathrm{d}t + \sigma_t\,\mathrm{d}W_t, &\qquad X_0 \sim p_0, \quad X_1 \sim p_1. \end{aligned} \tag{7}$$

We optimize over admissible finite-energy controls and assume the bridge is attained in this class. The resulting controlled process P is called Schrödinger Bridge between source distribution p_0 and target distributionp_1 with reference R and is denoted in our paper as:

$$P = \mathrm{SB}_R(p_0, p_1) \tag{8}$$

In other words, it takes the reference diffusion R and searches for closest to it, in KL divergence, diffusion that satisfies marginal constrains (Léonard, 2014). The most popular choice of R is Wiener process with constant volatility ϵ, i.e., W^ϵ. In this case, minimizing the path-space KL divergence is equivalent to minimizing the expected control energy required to transport p_0 to p_1, providing a principled formulation of unpaired domain translation Shi et al. (2023).

Schrödinger bridges and Stochastic Optimal Control. Under mild regularity assumptions this problem can be converted into regular SOC problem with one marginal constraint Liu et al. (2025a):

$$\begin{aligned} u^\star = \arg\min_u \mathbb{E}_{R^u}\left[\frac{1}{2}\int_0^1 \|u_t(X_t)\|^2 \,\mathrm{d}t + \log\frac{\widehat{\varphi}_1(X_1)}{p_1(X_1)}\right], \\ \text{s.t.} \quad \mathrm{d}X_t = \big[f_t(X_t) + \sigma_t u_t(X_t)\big]\,\mathrm{d}t + \sigma_t\,\mathrm{d}W_t, \qquad X_0 \sim p_0. \end{aligned} \tag{9}$$

where $\widehat{\varphi}_1(X_1)$ is the Schrödinger Potential Léonard (2014). Which is defined as follows:

$$\varphi_t(x_t) = \int p^R_{1|t}(x_1 \mid x_t)\varphi_1(x_1)\,\mathrm{d}x_1, \qquad \varphi_0(x_0)\widehat{\varphi}_0(x_0) = p_0(x_0), \tag{10a}$$

$$\widehat{\varphi}_t(x_t) = \int p^R_{t|0}(x_t \mid x_0)\widehat{\varphi}_0(x_0)\,\mathrm{d}x_0, \qquad \varphi_1(x_1)\widehat{\varphi}_1(x_1) = p_1(x_1). \tag{10b}$$

The Eq 10a propagates φ_1 backward in time, while Eq 10b propagates $\widehat{\varphi}_0$ forward in the density convention. Their product gives the bridge marginal at time t: $p^P_t = \varphi_t\widehat{\varphi}_t$. One can as well say that Schrödinger Potential $\widehat{\varphi}_1(X_1)$ helps to *debias* the SOC problem solution to fit the marginal p_1. Such result is widely used and can lead the application of SOC problems solving algorithms, such as Domingo-Enrich et al. (2025); Gushchin et al. (2023), to solving the Schrodinger Bridge Problem.

Corrector Matching. Several works solve the Schrödinger bridge problem through the SOC formulation Eq 9 (Gushchin et al., 2023; Liu et al., 2025a). This formulation requires the terminal cost $\log(\widehat{\varphi}_1/p_1)$, which is initially unknown. Gushchin et al. (2023) learn this cost through the entropic optimal transport dual formulation. In contrast, Liu et al. (2025a) use Adjoint Matching Eq 6, which requires only its gradient. When the target density p_1 is known up to a normalizing constant, its score $\nabla_x \log p_1$ is available, leaving only the *corrector* $h^* := \nabla_x \log \widehat{\varphi}_1$ to be learned. The corrector admits the regression characterization:

$$h^*(x) := \nabla_x \log \varphi(x) = \arg\min_h \mathbb{E}_{X_1, X_0 \sim P^{u^*}}[\|h(X_1) - \nabla_{x_1} \log p^R(X_1|X_0)\|^2]. \tag{11}$$

Note that this objective requires samples from the optimal bridge P^{u^*}, which is itself unknown. To resolve this interdependence, controller and corrector updates are alternated, using trajectories generated by the current controller to update the corrector (Liu et al., 2025a).

3 TILTING A SCHRÖDINGER BRIDGE

We first define the Tilted Schrödinger Bridge Problem (TiltedSBP) in § 3.1, then relate formulations using the base process R and pretrained bridge P as references. In § 3.2, we show equivalence of the corresponding path-space solutions (Prop. 1), relate their Schrödinger potentials (Prop. 2), and derive a relative SOC formulation that avoids explicit knowledge of p_1 (Prop. 3).

3.1 TILTED SCHRÖDINGER BRIDGE PROBLEM

First, let us recap the basic Schrodinger Bridge Problem Eq 7 between distributions p_0, p_1 and its solution $P = \mathrm{SB}_R(p_0, p_1)$, which we call **pretrained Schrödinger Bridge**. Then let us tilt the target distribution by some reward function $r(x)$, i.e., $p_1^r(x) \propto p_1(x)e^{r(x)}$. Then the central problem that we target is this paper is finding a Schrödinger Bridge between source distribution p_0 and target p_1^r, or more formally:

Definition 1 (TiltedSBP)**.** Let $R \in P(\Omega)$ be a reference diffusion law on $\Omega = C([0,1], \mathbb{R}^d)$. Given a target and source probability densities $p_0, p_1 \in P(\mathbb{R}^d)$ and a measurable reward $r : \mathbb{R}^d \to \mathbb{R}$, assume $0 < \int p_1(x)e^{r(x)}dx < \infty$ and define the reward-tilted target density:

$$p_1^r(x) \propto p_1(x)e^{r(x)} \tag{12}$$

The *Tilted Schrödinger Bridge problem* is to find:

$$u^\star = \arg\min_u \mathbb{E}_{R^u}\left[\frac{1}{2}\int_0^1 \|u_t(X_t)\|^2 \,\mathrm{d}t\right] \tag{13}$$

$$\text{s.t.} \quad \mathrm{d}X_t = \big[f_t(X_t) + \sigma_t u_t(X_t)\big]\,\mathrm{d}t + \sigma_t\,\mathrm{d}W_t, \qquad X_0 \sim p_0, \quad X_1 \sim \underline{p_1^r}.$$

Where the optimal solution is denoted by $P^{u^*} = \mathrm{SB}_R(p_0, p_1^r)$. While problem formulation being fully valid the practical solution can have problems. In many practical cases the distribution p_1 distribution can be given by empirical samples, while $r(x)$ can be given by reward, which would complicate the distribution p_1^r representation.

3.2 RELATIVE STOCHASTIC OPTIMAL CONTROL FORMULATION FOR TILTEDSBP

As is standard in many reward finetuning methods Uehara et al. (2024); Domingo-Enrich et al. (2025), which also target reward tilted target distribution p_1^r, the tilting problem defined relative the to the base process, i.e., $P = \mathrm{SB}_R(p_0, p_1)$ in our case. Which gives us motivation to define the TiltedSBP problem rather with as P as a reference, i.e., which we call **relative problem**, rather than with R as a reference. We first state conditions that will be actively used throughout the paper and show that changing the reference preserves the TiltedSBP solution.

Assumption 1 (Reference and reward)**.** *The reference starts from p_0, $p_{0,1}^R$ and $p_0 \otimes p_1$ define equivalent measures with KL$(p_0 \otimes p_1 \| p_{0,1}^R) < \infty$ and r is bounded above.*

Proposition 1 (Change of reference)**.** *Under Assumptions 1,*

$$\mathrm{SB}_P(p_0, p_1^r) = \mathrm{SB}_R(p_0, p_1^r) = P^r. \tag{14}$$

Thus, terminal *retargeting* can use the pretrained bridge as its reference without changing the desired solution. This permits learning an incremental control around the pretrained drift of P. The detailed proof is given in Appendix A. However, $\mathrm{SB}_P(p_0, p_1^r)$ and $\mathrm{SB}_R(p_0, p_1^r)$ problems while being equivalent have different Schrodinger Potentials Eq 10.

Let us further explore the connections between the TiltedSBP problem with R as reference and TiltedSBP problem with P as reference, in the context of Schrödinger Potentials. Let $\widehat{\varphi}_1^{\mathrm{old}}$, $\widehat{\varphi}_1^r$ and $\widehat{\psi}_1^r$ denote the terminal Schrodinger Potentials of $\mathrm{SB}_R(p_0, p_1)$, $\mathrm{SB}_R(p_0, p_1^r)$, $\mathrm{SB}_P(p_0, p_1^r)$ problems correspondingly, as defined in Eq 10. The relative SOC terminal cost $f_{\mathrm{rel}}^r(x) := \log \frac{\widehat{\psi}_1^r(x)}{p_1^r(x)}$ and can be expressed without p_1 factor:

Proposition 2 (Relative terminal potential relation)**.** *The terminal cost term of SOC formulated relative SB problem* $\mathrm{SB}_P(p_0, p_1^r)$ *admits:*

$$f^r_{\mathrm{rel}}(x) := \log \frac{\widehat{\psi}^r_1(x)}{p^r_1(x)} = \log \widehat{\varphi}^r_1(x) - \log \widehat{\varphi}^{\mathrm{old}}_1(x) - r(x) + C, \tag{15}$$

where C is a constant that doesn't depend on x.

The explicit target density p_1 cancels from the terminal cost, while its influence remains in the old potential $\widehat{\varphi}^{\mathrm{old}}_1$. The proof is given in Appendix A. Then to construct the terminal cost f^r_{rel} for relative SOC problem Eq 9 one needs only the reward r and Schrodinger Potentials of absolute problems $\log \widehat{\varphi}^r_1(x)$ with R as reference, i.e., $\log \widehat{\varphi}^r_1(x)$ or $\log \widehat{\varphi}^{\mathrm{old}}_1(x)$.

Under the assumption that reward r and the log-potentials admit continuous differentiable versions on a region, where $p_1(x) > 0$ the identity also given the terminal-cost gradient there:

$$\nabla f^r_{\mathrm{rel}}(x) = \nabla \log \widehat{\varphi}^r_1(x) - \nabla \log \widehat{\varphi}^{\mathrm{old}}_1(x) - \nabla r(x) = h^r(x) - h_{\mathrm{old}}(x) - \nabla r(x), \tag{16}$$

where $\nabla \log \widehat{\varphi}^r_1$ is h^r, $\nabla \log \widehat{\varphi}^{\mathrm{old}}_1$ is h_{old}. Section 4 makes use of Eq 16 and develops a procedure to learn the terminal corrector $\nabla f^r_{\mathrm{rel}}$. Then finally, we can construct the relative to P SOC Problem Eq 9 with optimal control $u^\star_r$:

Proposition 3 (Relative Schrödinger Bridge Stochastic Optimal Control Problem)**.** *Given the Assumptions of Def 1, Schrödinger Bridge $P = \mathrm{SB}_R(p_0, p_1)$ with optimal control $u^\star$ and corresponding drift b_t define such a controlled by u_t diffusion P^{u_t}:*

$$P^{u_t}: \quad \mathrm{d}X_t = [\underbrace{f_t(X_t) + \sigma_t u^\star_t(X_t)}_{b_t(X_t)} + \sigma_t u_t(X_t)]\,\mathrm{d}t + \sigma_t\,\mathrm{d}W_t, \qquad X_0 \sim p_0, \tag{17}$$

where $b_t(X_t)$ is the resulting controlled drift of P. Then optimal control $u^\star_r$ provided as a solution to the following optimization problem:

$$u^\star_r = \arg\min_u \mathbb{E}_{P^u}\left[\frac{1}{2}\int_0^1 \|u_t(X_t)\|^2\,\mathrm{d}t + \log \widehat{\varphi}^r_1(x) - \log \widehat{\varphi}^{\mathrm{old}}_1(x) - r(x)\right], \tag{18}$$

would result in solution to the TiltedSBP, i.e., $P^{u^\star_r} = \mathrm{SB}_R(p_0, p^r_1)$.

This formulation requires the pretrained bridge P, its terminal potential $\log \widehat{\varphi}^{\mathrm{old}}_1$, and reward r, but importantly no further access to the base target p_1 through samples or density evaluations. The remaining unknown potential $\log \widehat{\varphi}^r_1$ is addressed in the next section.

4 Tilted Schrödinger Bridge Matching

In this section, we introduce an algorithm for solving the TiltedSBP. Prop. 3 gives a feasible SOC formulation, but obtaining the optimal controller u requires the corrector $\log \widehat{\varphi}^r_1(x)$. Since the controller and corrector are interdependent, we propose an alternating optimization procedure: update the controller $u^{(k)}$ with fixed corrector $h^{(k)}$, then update the corrector $h^{(k+1)}$ with fixed controller $u^{(k)}$. For controller optimization, we use Controller Matching Eq 6 and define its iterative fixed-corrector version as the **Controller Update**:

$$u^{(k)} = \arg\min_u \mathbb{E}_{P^{\overline{u}}}[\mathcal{L}_{\mathrm{ctrl}}(u, X, h^{(k)})] =: \arg\min_u \mathbb{E}_{P^{\overline{u}}} \int_0^1 \left\| u(X_t, t) + \sigma_t \tilde{a}^{(k)}_t(X) \right\|^2 \mathrm{d}t, \tag{19}$$

$$\begin{aligned} \tilde{a}_1(X) &= \nabla_x f^r_{\mathrm{rel}}(X_1) = -\nabla_x r(X_1) + h^{(k)}(X_1) - h_{\mathrm{old}}(X_1), \\ -\frac{\mathrm{d}\tilde{a}^{(k)}_t(X)}{\mathrm{d}t} &= \nabla_x b_t(X_t)^\top \tilde{a}^{(k)}_t(X), \end{aligned} \tag{20}$$

where $\overline{u} = \mathrm{stopgrad}(\overline{u})$ and for optimization of corrector we choose Corrector Matching § 2 and call its iterative version as **Corrector Update**:

$$h^{(k+1)} = \arg\min_h \mathbb{E}_{P^{u^{(k)}}}[\mathcal{L}_{\mathrm{corr}}(h, X_0, X_1)] =: \arg\min_h \mathbb{E}_{P^{u^{(k)}}}[\|h(X_1) - \nabla_{x_1} \log p^R(X_1|X_0)\|^2]. \tag{21}$$

Algorithm 1 Tilted Schrödinger Bridge Matching (TSBM)

Require: frozen old drift b, old corrector h_{old}, reference process R, reward r, stages K, number of controller updates M_{ctrl}, number of corrector updates M_{corr}

initialize $u^{(0)} \leftarrow 0$ and $h^{(0)} \leftarrow h_{\text{old}}$
for $k = 0, \dots, K-1$ **do**
  $u \leftarrow \text{copy}(u^{(k)})$
  for $m = 0, \dots, M_{\text{ctrl}} - 1$ **do**
    Sample *paths* $(X_t)_{t\in[0,1]}$ from $P^{\overline{u}}$ by the inference of SDE with drift $b + \sigma u$;
    $\tilde{a}_1^{(k)} \leftarrow -\nabla r(X_1) + h^{(k)}(X_1) - h_{\text{old}}(X_1)$;
    Propagate $\tilde{a}_t^{(k)}$ by solving backward adjoint lean ODE using Eq 5;
    Make a gradient step on $\mathcal{L}_{\text{ctrl}}$ w.r.t. u parameters;
  Sample *endpoints* (X_0, X_1) from $P^{\overline{u}}$ by the inference of SDE with drift $b + \sigma u$;
  $h \leftarrow \text{copy}(h^{(k)})$
  for $m = 0, \dots, M_{\text{corr}} - 1$ **do**
    Make a gradient step on $\mathcal{L}_{\text{corr}}$ w.r.t. h parameters;
  $u^{(k+1)} \leftarrow u$, $h^{(k+1)} \leftarrow h$

return $u^{(K)}, h^{(K)}$

Now both Controller and Corrector Matching updates do relax their interdependency into dependency on the previous iteration (k) results and yield a fixed point alternative updates algorithm. We call this iterative procedure **TSBM**, each iteration k we call **stage**. Next we show that this algorithm indeed converges to the Tilted Schrodinger Bridge:

Theorem 1 (TSBM convergence)**.** *Let $P = \text{SB}_R(p_0, p_1)$. Under Assumptions 1 and 2, the exact population updates in Eq 19 and Eq 21, initialized by $u^{(0)} = 0$ and $h^{(0)} = h_{\text{old}}$, satisfy*

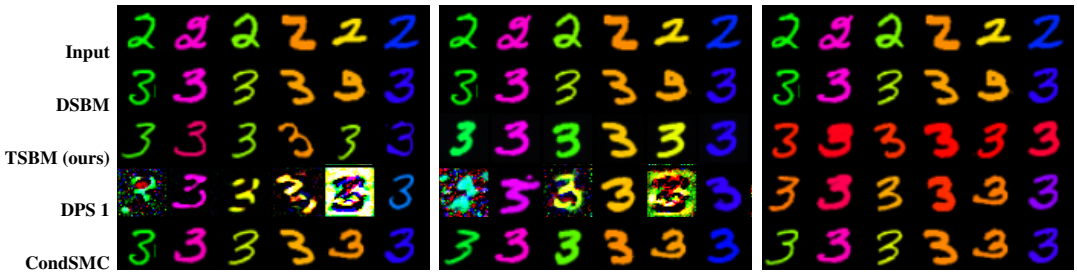
$$\text{KL}(P^{u^*} \| P^{u^{(k)}}) \longrightarrow 0, \qquad P^{u^*} = \text{SB}_R(p_0, p_1^r). \tag{22}$$

The proof follows the ASBS/IPF strategy (Liu et al., 2025a) in our relative setting and is given in Appendix A, with the pseudo-algorithm in Algorithm 1 and implementation details in Appendix C. In practice, we finetune a pretrained Schrödinger bridge P, typically represented by forward and backward drift networks learned iteratively (Shi et al., 2023; Vargas et al., 2023; Liu et al., 2025a; Tamogashev & Malkin, 2026; Kholkin et al., 2026b). The forward drift approximates b and is used in the Controller Update (Eq. 19), while the backward drift approximates the pretrained corrector h_{old} Liu et al. (2025a) and is used in both updates (Eqs. 19 and 21). We use the same parameterization for TSBM. Replay buffers, network architectures, and other details are deferred to Appendix C.

5 Related Work

Schrödinger bridge solvers. Schrödinger Bridge solvers are mostly based on diffusion models Song et al. (2021). Some methods make use of Entropic Optimal Transport connection Gushchin et al. (2024a; 2023), while most popular methods are based on iterative procedures, such as Iterative Proportional Fitting (IPF) De Bortoli et al. (2021); Vargas et al. (2021), Iterative Markovian Fitting (IMF) Shi et al. (2023); Gushchin et al. (2024b); Ksenofontov & Korotin (2025) or their hybrid Kholkin et al. (2026b). The Adjoint Matching is utilized as a part of SB solvers Liu et al. (2025a); Havens et al. (2025). In practice, the SB models are utilized for unpaired domain translation Shi et al. (2023), trajectory inference Tong et al. (2024); Noble et al. (2026), sampling from Boltzman distributions Liu et al. (2025a); Guo et al. (2026); Havens et al. (2025); Tamogashev & Malkin (2026), Mutual Information Estimation Kholkin et al. (2026a); Zabarianska et al. (2026) or even data generation Shin et al. (2026). However, there are no current methods that do support exact marginal reward tilting of the pretrained SB models.

Reward steering and finetuning. Reward steering adapts pretrained generative models to objectives beyond matching the training distribution, such as human preferences in image generation (Black et al., 2024), DNA and protein design (Wang et al., 2025), and physical constraints (Tauberschmidt et al., 2026). Methods include inference-time guidance, such as classifier and classifier-free guidance



(a) Thin stroke **(b)** Thick stroke **(c)** Red chroma

Figure 2: **Qualitative results on Colored MNIST.** Rows show the source input, pretrained DSBM, TSBM, DPS with $\gamma = 1$, and CondSMC with $N = 8$ particles for the thin-stroke, thick-stroke, and red-chroma targets. Comparisons with additional baselines are in Appendix F.

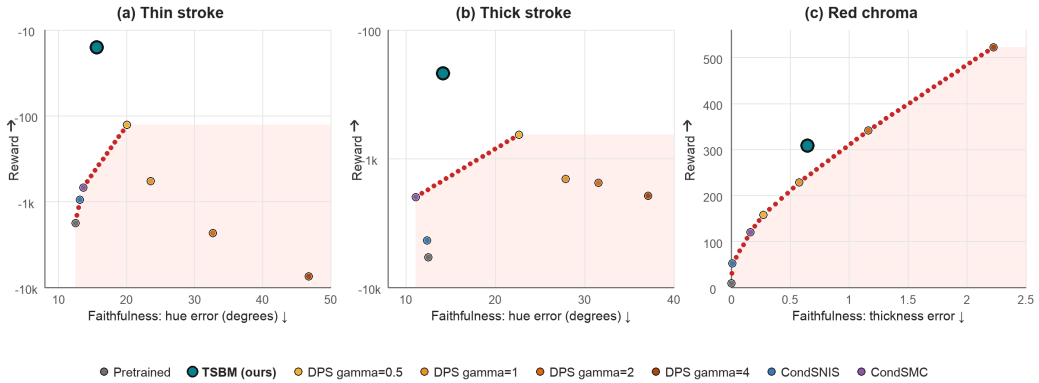


Figure 3: **Quantitative analysis on Colored MNIST.** Reward ↑ is the mean unscaled terminal reward $r(x)$. The two stroke panels use logarithmic spacing by reward magnitude. Faithfulness ↓ is foreground hue error relative to the input for thin and thick strokes, and mean thickness deviation from the input for red chroma. The red dotted lines connect the non-TSBM Pareto points, while pale red shading marks values below them.

(Dhariwal & Nichol, 2021; Ho & Salimans, 2022), diffusion posterior sampling (Chung et al., 2023a), and particle-based methods (Del Moral et al., 2006; Wu et al., 2023a), which can increase sampling costs or introduce artifacts under strong guidance. Finetuning alternatives include reward optimization (Black et al., 2024; Clark et al., 2024) and preference learning (Wallace et al., 2024; Kim et al., 2024). However, accounting for the pretrained distribution further allows for regularization and targets its exponential tilt rather than only maximizing reward (Domingo-Enrich et al., 2025; Uehara et al., 2024; Zhao et al., 2025; Liu et al., 2025b). Stochastic optimal control methods emerged as alternative branch of methods allowing for targeting an exponential tilt, while the methods include matching based methods Domingo-Enrich et al. (2024; 2025); Bergmeister et al. (2026) or differentiation through trajectory based methods Wang et al. (2025); Vargas et al. (2023). However, these works do target mostly only noise-to-data generative models.

Reward steering and diffusion bridges. While reward steering is well studied for standard diffusion models, extending it to diffusion bridges, including Schrödinger bridges, requires accounting for source target dependence. In particular, conditional tilting, $q_{1|0}(x_1|x_0) \propto p^P_{1|0}(x_1|x_0)\exp(r(x_1))$, generally does not produce the desired terminal marginal $q_1(x_1) \propto p^P_1(x_1)\exp(r(x_1))$ when the source marginal is fixed. For bridges trained on paired data (Zhou et al., 2024; Liu et al., 2023), CDDB applies inference-time data-consistency guidance with a DPS-like variant (Chung et al., 2023b;a), He et al. (2026) shows the practical applicability of SMC-like Del Moral et al. (2006) population steering, but doesn't evaluate it, while PalSB uses physics-informed finetuning through truncated trajectory differentiation (Li et al., 2025). Schrödinger Bridges were also reward aligned to learn Boltzmann samplers with evaluable target energies (Liu et al., 2025a; Guo et al., 2026; Havens et al., 2025), a different setting from reward tilting of a pretrained bridge whose terminal density is implicit. These works do not establish the joint guarantee considered in our work: a prescribed terminal *marginal* reward tilt with preservation of the source marginal when finetuning a general pretrained Schrödinger Bridge model.

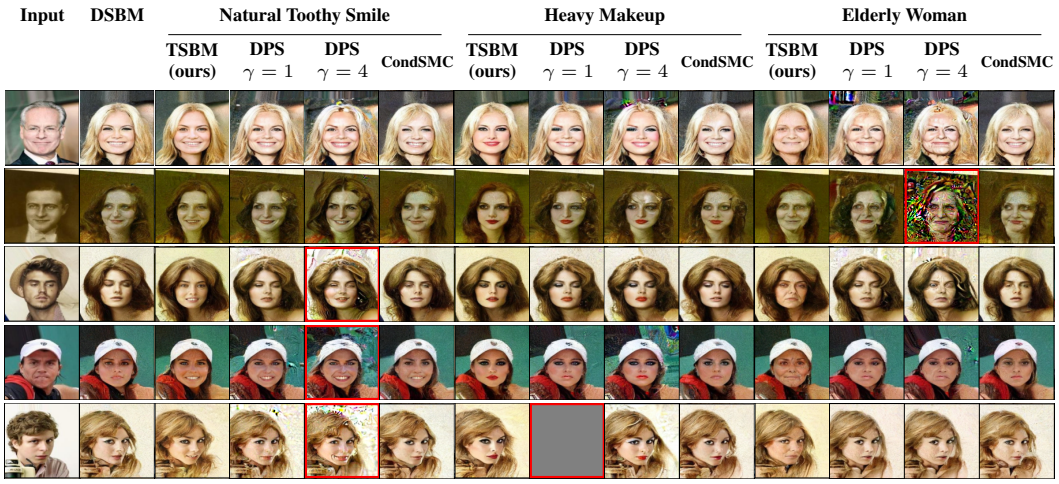


Figure 4: **Qualitative results for CelebA experiment.** Each row uses the same source across the pretrained DSBM and three CelebA prompt groups. Within each group, we compare TSBM, DPS with $\gamma \in \{1, 4\}$, and CondSMC. Red borders highlight selected outputs with visible artifacts.

6 Experiments

In this section we experimentally evaluate the TSBM, as the Schrodinger Bridges reward finetuning method. In all the cases we do start from Schrodinger Bridge trained via DSBM method Shi et al. (2023). We test TSBM on unpaired image translation problems: 1) Colored MNIST dataset image translation between digit "2" to digit "3" with reward tilt towards digit thickness and color; 2) Celeba dataset male to female image translation with reward tilts toward different image attributes: "natural toothy smile", "heavy makeup" and "elderly woman" with reward functions provided by ImageReward Xu et al. (2023) with corresponding prompts. In addition during training the reward functions are scaled linearly by *strength* parameter λ, i.e., $\lambda * r(x)$. Additional experimental details are presented in Appendix E and more experiments are presented in Appendix G.

Competitors. As was noted in the § 5 the reward finetuning of Schrodinger Bridge models is underexplored and has lack of established methods. However, there are approximate methods for *tilting the conditional distribution*, such as CDDB Chung et al. (2023b), which we call just **DPS** in the experimental section and test with different *strength* parameter γ. Additionally, we test the conditional population based inference time adaptation methods for *tilting the conditional distribution*: Sequential Monte Carlo (**CondSMC**) Del Moral et al. (2006) and Self Normalized Importance Sampling (**CondSNIS**). Competitor algorithms in particular are described in Appendix B.

6.1 Colored MNIST unpaired translation attributes manipulation

Setup. We test the TSBM on images, starting from Colored MNIST digit "2" to digit "3" unpaired domain translation. We take the Colored MNIST translation setup from Gushchin et al. (2024b). DSBM pretrain si trained by 20 stages with $\sigma_t = 1$ SB volatility coefficient, while TSBM is trained for 5 stages with strength $\lambda = 1000$. The reward functions are: 1) "red chroma" which favors the red rgb channel vs others 2) "thin stroke" and "thick stroke", which do calculate the approximate thickness of a digit by computing the ratio between strokes area and foreground perimeter to then penalize the thickness different from 3.65 and 7 for "thin stroke" and "thick stroke" correspondingly. See Appendix D for more information of reward functions, Appendix G for ablation studies on number of TSBM stages and corrector necessity and Appendix F for qualitative results.

Results. Figure 2 compares outputs from TSBM and the baselines with the source inputs and pretrained DSBM. TSBM produces thinner, thicker, or redder digits as specified by the reward function, while largely preserving attributes unrelated to the reward: thin- and thick-stroke outputs retain the source color, whereas red-chroma outputs retain approximately the source thickness. The quantitative results support this balance between reward optimization and faithfulness to the source. In comparison, DPS introduces visible artifacts, while CondSMC makes limited changes to the targeted attributes. Figure 3 reports quantitative results, i.e., *reward* and *faithfulness*, and one can see that **TSBM improves upon the Pareto frontier**. These results demonstrate that **TSBM** extends to image translation and achieves the most favorable **trade-off between reward optimization and source faithfulness** among the evaluated methods.

Table 1: **Quantitative results for CelebA experiment**. Means are over 1024 fixed inputs per prompt, with TSBM evaluated after stage 5. ImageReward and PickScore use translation, cropping, and horizontal-flip augmentation; CLIP-IQA measures naturalness. All metrics are higher-is-better; bold marks the largest mean. (*) indicates the appearance of visual artifacts.

	Natural Toothy Smile			**Heavy Makeup**			**Elderly Woman**		
Method	ImageReward (train)↑	PickScore (eval)↑	CLIP-IQA ↑	ImageReward (train)↑	PickScore (eval)↑	CLIP-IQA ↑	ImageReward (train)↑	PickScore (eval)↑	CLIP-IQA ↑
Pretrained	-1.2348	17.6109	0.5950	-1.4054	17.3777	**0.5950**	-1.4960	18.1592	0.5950
TSBM (ours)	0.1398	**18.2345**	**0.6595**	0.9146	**18.0390**	0.3508	-0.0204	**18.8200**	**0.6340**
DPS* $\gamma = 1$	0.0622	17.7347	0.6192	0.8576	17.5803	0.3789	-0.7328	18.2559	0.5623
DPS* $\gamma = 2$	0.5993	17.6831	0.6312	1.2507	17.6124	0.3401	-0.2527	18.3329	0.5218
DPS* $\gamma = 4$	**0.9783**	17.4688	0.6164	**1.5151**	17.6236	0.3133	**0.2239**	18.3346	0.4618
CondSMC	-0.5728	17.8375	0.6107	-0.1525	17.5395	0.4805	-1.2800	18.1762	0.5927

6.2 CelebA unpaired translation attributes manipulation

Setup. Next, we test TSBM on unpaired male-to-female CelebA translation at 128×128 resolution. Starting from a pretrain DSBM trained for 20 stages with $\sigma_t = 1$, we train a TSBM for 5 stages. We use a frozen ImageReward model (Xu et al., 2023), which is normalized w.r.t. our CelebA data, to define our *train reward* for optimization and baselines sample generation with the corresponding prompts: *Natural Toothy Smile*, *Heavy Makeup*, and *Elderly Woman*. While PickScore Kirstain et al. (2023) with the same prompts is used for evaluation for exposing the potential *reward hacking*. The reward function details can be seen at Appendix D.3. In addition we report CLIP-IQA (naturalness) metric for assessment of resulting image quality. All the methods use reward strength $\lambda = 30$ and 100 NFE for evaluation. See Appendix F for additional qualitative results.

Results. Figure 4 shows the samples on the same five sources for each prompt. TSBM follows the requested changes in smile, makeup, or apparent age, while other methods may struggle. Stronger DPS guidance shows itself also succesfull and slighly weaker than TSBM, but introduces visible artifacts in some examples, while CondSMC produces weaker tilt. Table 1 quantifies the reward-faithfulness trade-off: TSBM improves ImageReward over the pretrained DSBM for all three train reward, but not as strongly as DPS. However, TSBM obtains the **highest eval PickScore reward** in each comparison, which suggests that **TSBM results generalize well**. While DPS huge gains in train ImageReward reward does not result in visual artifacts and modest eval reward growth, which suggests that DPS is doing rather *reward hacking*. TSBM also has the highest CLIP-IQA for the smile and elderly prompts, while the heavy-makeup result is less favorable and on contrary lowers CLIP-IQA, which may indicate that heavy make up tilting is not doesn't improve CLIP-IQA in general. In addition, TSBM is faster than baselines on inference, see Appendix G.2. Thus, **TSBM improves prompt alignment, which generalizes well and keeps images visually high-fidelity**.

7 Discussion

Potential impact. For the problem of reward finetuning of pretrained Schrödinger Bridges we introduce TSBM, the first practical algorithm for this problem, which alternates controller and corrector matching. On Colored MNIST and CelebA, TSBM steers attributes in unpaired image translation while largely preserving source content. These results suggest that pretrained bridges can be adapted to new preferences without training a new bridge from scratch.

Limitations and future work. We assume exact optimization and validity of all theoretical assumptions, which may fail in practice. The method also requires an accessible endpoint transition score and differentiable reward, motivating extensions to non-differentiable rewards Bergmeister et al. (2026). Possible extensions include adapting unpaired scientific translation models to physical constraints Li et al. (2025), image aesthetics improvement Domingo-Enrich et al. (2025) or counterfactual explanations Jeanneret et al. (2022). Further methodological research includes other training objectives, such as other adjoint algorithms Domingo-Enrich et al. (2024); Vargas et al. (2023) or log variance losses Tamogashev & Malkin (2026). Training TSBM requires repeated bridge simulations and adjoint ODE solves, making finetuning computationally demanding, while distillation-based variant (Gushchin et al., 2025) may offer a way to reduce this cost. Recent work on discrete state space reward optimization and adjoint matching (Wang et al., 2025; So et al., 2026; Guo et al., 2026) also suggests extending TSBM to discrete state spaces.

AI USE STATEMENT

In preparing this submission, we used generative AI tools to assist drafting the paper sections, then the authors have fully rewritten the text and partly polished it by generative AI tools. Generative AI tools were used to launch the experiments, write code, draft proofs and draw figures, however the research ideation came from authors and all the code and theoretical results were checked manually by authors. Generative AI was used to assist finding related work, which was manually inspected by the authors and carefully cited in this manuscript.

ETHICS STATEMENT

This work studies reward finetuning of pretrained Schrödinger bridges. Our experiments use synthetic two-dimensional data, Colored MNIST digits, and CelebA face images. We did not recruit participants or collect new personal data.

REPRODUCIBILITY STATEMENT

Section 3 presents the theoretical construction, with proofs in Appendix A, while practical construction in detail is given in Appendix C. Section 6 presents the experiments, Appendix E reports the experimental settings, Appendix D defines the rewards, and Appendix G provides additional results. We submit the code as supplementary material.

REFERENCES

Richard Bellman. *Dynamic Programming*. Princeton University Press, 1957.

Andreas Bergmeister, Stefanie Jegelka, Nikolas Nüsken, Carles Domingo-Enrich, and Jakiw Pidstrigach. Reinforce adjoint matching: Scaling RL post-training of diffusion and flow-matching models, 2026. URL https://arxiv.org/abs/2605.10759.

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *The Twelfth International Conference on Learning Representations*, 2024.

Hyungjin Chung, Jeongsol Kim, Michael Thompson McCann, Marc Louis Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *The Eleventh International Conference on Learning Representations*, 2023a.

Hyungjin Chung, Jeongsol Kim, and Jong Chul Ye. Direct diffusion bridge using data consistency for inverse problems. In *Advances in Neural Information Processing Systems*, volume 36, 2023b.

Kevin Clark, Paul Vicol, Kevin Swersky, and David J. Fleet. Directly fine-tuning diffusion models on differentiable rewards. In *The Twelfth International Conference on Learning Representations*, 2024.

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

Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. Sequential monte carlo samplers. *Journal of the Royal Statistical Society: Series B*, 68(3):411–436, 2006.

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

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

Carles Domingo-Enrich, Michal Drozdzal, Brian Karrer, and Ricky T. Q. Chen. Adjoint matching: Fine-tuning flow and diffusion generative models with memoryless stochastic optimal control. In *The Thirteenth International Conference on Learning Representations*, 2025.

Promit Ghosal and Marcel Nutz. On the convergence rate of sinkhorn's algorithm. *Mathematics of Operations Research*, 2025. doi: 10.1287/moor.2024.0427. URL https://doi.org/10.1287/moor.2024.0427.

Wei Guo, Yuchen Zhu, Xiaochen Du, Juno Nam, Yongxin Chen, Rafael Gómez-Bombarelli, Guan-Horng Liu, Molei Tao, and Jaemoo Choi. Discrete adjoint schr\" odinger bridge sampler. *arXiv preprint arXiv:2602.08243*, 2026.

Nikita Gushchin, Alexander Kolesov, Alexander Korotin, Dmitry P. Vetrov, and Evgeny Burnaev. Entropic neural optimal transport via diffusion processes. In *Advances in Neural Information Processing Systems*, volume 36, pp. 75517–75544, 2023.

Nikita Gushchin, Sergei Kholkin, Evgeny Burnaev, and Alexander Korotin. Light and optimal schrödinger bridge matching. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 17100–17122, 2024a.

Nikita Gushchin, Daniil Selikhanovych, Sergei Kholkin, Evgeny Burnaev, and Alexander Korotin. Adversarial schrödinger bridge matching. In *Advances in Neural Information Processing Systems*, volume 37, pp. 89612–89651, 2024b.

Nikita Gushchin, David Li, Daniil Selikhanovych, Evgeny Burnaev, Dmitry Baranchuk, and Alexander Korotin. Inverse bridge matching distillation. *arXiv preprint arXiv:2502.01362*, 2025.

Sven Gutjahr, Riccardo De Santi, Luca Schaufelberger, Kjell Jorner, and Andreas Krause. Constrained flow optimization via sequential fine tuning for molecular design. *arXiv preprint arXiv:2605.30610*, 2026.

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

Jiajun He, José Miguel Hernández-Lobato, Yuanqi Du, and Francisco Vargas. RNE: Plug-and-play diffusion inference-time control and energy-based training. In *The Fourteenth International Conference on Learning Representations*, 2026.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. CLIPScore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 7514–7528, 2021. doi: 10.18653/v1/2021.emnlp-main.595.

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

Guillaume Jeanneret, Loïc Simon, and Frédéric Jurie. Diffusion models for counterfactual explanations. In *Proceedings of the Asian conference on computer vision*, pp. 858–876, 2022.

Sergei Kholkin, Ivan Butakov, Evgeny Burnaev, Nikita Gushchin, and Alexander Korotin. InfoBridge: Mutual information estimation via bridge matching. In *The Fourteenth International Conference on Learning Representations*, 2026a.

Sergei Kholkin, Grigoriy Ksenofontov, David Li, Nikita Kornilov, Nikita Gushchin, Alexandra Suvorikova, Alexey Kroshnin, Evgeny Burnaev, and Alexander Korotin. Diffusion & adversarial schrödinger bridges via iterative proportional markovian fitting. In *The Fourteenth International Conference on Learning Representations*, 2026b.

Minu Kim, Yongsik Lee, Sehyeok Kang, Jihwan Oh, Song Chong, and Se-Young Yun. Preference alignment with flow matching. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-Pic: An open dataset of user preferences for text-to-image generation. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Grigoriy Ksenofontov and Alexander Korotin. Categorical Schrödinger bridge matching. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 31727–31751. PMLR, 2025. URL https://proceedings.mlr.press/v267/ksenofontov25a.html.

Christian Léonard. A survey of the schrödinger problem and some of its connections with optimal transport. *Discrete and Continuous Dynamical Systems A*, 34(4):1533–1574, 2014.

Zeyu Li, Hongkun Dou, Shen Fang, Wang Han, Yue Deng, and Lijun Yang. Physics-aligned field reconstruction with diffusion bridge. In *The Thirteenth International Conference on Learning Representations*, pp. 58864–58900, 2025.

Guan-Horng Liu, Arash Vahdat, De-An Huang, Evangelos Theodorou, Weili Nie, and Anima Anandkumar. I^2SB: Image-to-image Schrödinger bridge. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pp. 22042–22062. PMLR, 2023. URL https://proceedings.mlr.press/v202/liu23ai.html.

Guan-Horng Liu, Jaemoo Choi, Yongxin Chen, Benjamin Kurt Miller, and Ricky T. Q. Chen. Adjoint schrödinger bridge sampler. *arXiv preprint arXiv:2506.22565*, 2025a.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-GRPO: Training flow matching models via online RL. *arXiv preprint arXiv:2505.05470*, 2025b.

Maxence Noble, Marie Scheid, Yazid Janati, Eric Moulines, and Alain Durmus. Twisted schrödinger bridge matching. *arXiv preprint arXiv:2607.16987*, 2026.

Marcel Nutz. Introduction to entropic optimal transport, 2022. URL https://www.math.columbia.edu/~mnutz/docs/EOT_lecture_notes.pdf. Lecture notes, version December 5, 2022.

Erwin Schrödinger. Über die umkehrung der naturgesetze. *Sitzungsberichte der Preussischen Akademie der Wissenschaften, Physikalisch-mathematische Klasse*, pp. 144–153, 1931.

Yuyang Shi, Valentin De Bortoli, Andrew Campbell, and Arnaud Doucet. Diffusion schrödinger bridge matching. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Jeongwoo Shin, Jinhwan Sul, Joonseok Lee, Jaewoong Choi, and Jaemoo Choi. Efficient generative modeling beyond memoryless diffusion via adjoint Schrödinger bridge matching. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2602.15396.

Oswin So, Brian Karrer, Chuchu Fan, Ricky TQ Chen, and Guan-Horng Liu. Discrete adjoint matching. *arXiv preprint arXiv:2602.07132*, 2026.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=PxTIG12RRHS.

Kirill Tamogashev and Nikolay Malkin. Data-to-energy stochastic dynamics. In *International Conference on Learning Representations*, volume 2026, pp. 99944–99964, 2026.

Jan Tauberschmidt, Sophie Fellenz, Sebastian J. Vollmer, and Andrew B. Duncan. Physics-constrained fine-tuning of flow-matching models for generation and inverse problems. In *The Fourteenth International Conference on Learning Representations*, 2026.

Alexander Y. Tong, Nikolay Malkin, Kilian Fatras, Lazar Atanackovic, Yanlei Zhang, Guillaume Huguet, Guy Wolf, and Yoshua Bengio. Simulation-free Schrödinger bridges via score and flow matching. In *Proceedings of the 27th International Conference on Artificial Intelligence and Statistics*, volume 238, pp. 1279–1287. PMLR, 2024. URL https://proceedings.mlr.press/v238/tong24a.html.

Masatoshi Uehara, Yulai Zhao, Kevin Black, Ehsan Hajiramezanali, Gabriele Scalia, Nathaniel Lee Diamant, Alex M. Tseng, Tommaso Biancalani, and Sergey Levine. Fine-tuning of continuous-time diffusion models as entropy-regularized control. *arXiv preprint arXiv:2402.15194*, 2024.

Francisco Vargas, Pierre Thodoroff, Austen Lamacraft, and Neil D. Lawrence. Solving schrödinger bridges via maximum likelihood. *Entropy*, 23(9):1134, 2021. doi: 10.3390/e23091134.

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

Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8228–8238, 2024.

Chenyu Wang, Masatoshi Uehara, Yichun He, Amy Wang, Tommaso Biancalani, Avantika Lal, Tommi Jaakkola, Sergey Levine, Hanchen Wang, and Aviv Regev. Fine-tuning discrete diffusion models via reward optimization with applications to DNA and protein design. In *The Thirteenth International Conference on Learning Representations*, 2025.

Luhuan Wu, Brian L. Trippe, Christian A. Naesseth, David M. Blei, and John P. Cunningham. Practical and asymptotically exact conditional sampling in diffusion models. In *Advances in Neural Information Processing Systems*, volume 36, 2023a.

Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023b.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. ImageReward: Learning and evaluating human preferences for text-to-image generation. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Iryna Zabarianska, Sergei Kholkin, Grigoriy Ksenofontov, Ivan Butakov, and Alexander Korotin. Discrete bridges for mutual information estimation. In *2nd DeLTa Workshop at the Fourteenth International Conference on Learning Representations*, 2026.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 586–595, 2018.

Siyan Zhao, Devaansh Gupta, Qinqing Zheng, and Aditya Grover. d1: Scaling reasoning in diffusion large language models via reinforcement learning. *arXiv preprint arXiv:2504.12216*, 2025.

Linqi Zhou, Aaron Lou, Samar Khanna, and Stefano Ermon. Denoising diffusion bridge models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=FKksTayvGo.

APPENDIX CONTENTS

A Proofs **15**

A.1 Proof of Proposition 1 15

A.2 Proof of Proposition 2 16

A.3 Proof of Proposition 3 16

A.4 Assumptions and controlled diffusions for convergence 16

A.5 Controller Matching and the forward half bridge 18

A.6 Corrector Matching and the backward half bridge 19

A.7 Proof of Theorem 1 19

B Competitors **23**

B.1 Reward tilting with non-memoryless priors 23

B.2 DPS-style guidance 24

B.3 Conditional self-normalized importance sampling (CondSNIS) 24

B.4 Conditional sequential Monte Carlo (CondSMC) 24

C General design choices for TSBM **25**

D Reward Functions **25**

D.1 Toy illustrative experiment 25

D.2 Colored MNIST unpaired translation 26

D.3 CelebA unpaired translation 27

E Experimental details **28**

F Additional qualitative results **30**

F.1 Colored MNIST 30

F.2 CelebA 31

G Additional experiments **34**

G.1 Toy illustrative experiment 34

G.2 CelebA-128 inference time and working memory 34

G.3 Colored MNIST Red-Chroma. Static TSBM corrector and TSBM stages ablation. . 35

G.4 Colored MNIST Thick-target. TSBM stages ablation. 36

G.5 CelebA additional quantitative analysis 37

G.6 Additional CelebA rewards 39

A PROOFS

A.1 PROOF OF PROPOSITION 1

Proof. Write $(\varphi_t^{\mathrm{old}}, \widehat{\varphi}_t^{\mathrm{old}})$ for the potentials of P relative to R in Eq 10. Under Assumption 1, Léonard (2014, Proposition 2.3), together with the endpoint factorization and integrability results of Nutz (2022, Theorem 2.1 and Lemma 2.21), gives

$$\frac{\mathrm{d}P}{\mathrm{d}R} = \frac{\varphi_1^{\mathrm{old}}(X_1)}{\varphi_0^{\mathrm{old}}(X_0)} > 0, \qquad \int |\log \varphi_i^{\mathrm{old}}(x)| p_i(x)\,\mathrm{d}x < \infty, \quad i \in \{0,1\}.$$

Here $\mathrm{d}P/\mathrm{d}R$ denotes the path-law density, and Eq 10 gives

$$\varphi_0^{\mathrm{old}} \widehat{\varphi}_0^{\mathrm{old}} = p_0 \quad \Longrightarrow \quad \frac{\widehat{\varphi}_0^{\mathrm{old}}}{p_0} = \frac{1}{\varphi_0^{\mathrm{old}}}.$$

Since r is bounded above, for some finite M,

$$0 < w(x) := \frac{p_1^r(x)}{p_1(x)} \le M,$$
$$\int |\log \varphi_1^{\mathrm{old}}(x)| p_1^r(x)\,\mathrm{d}x \le M \int |\log \varphi_1^{\mathrm{old}}(x)| p_1(x)\,\mathrm{d}x < \infty.$$

The boundedness of w and $w|\log w|$, together with Assumption 1, gives

$$\mathrm{KL}(p_0 \otimes p_1^r \| p_{0,1}^R) = \iint w(x_1) p_0(x_0) p_1(x_1) \log \frac{p_0(x_0) p_1(x_1)}{p_{0,1}^R(x_0, x_1)}\,\mathrm{d}x_0 \mathrm{d}x_1 + \int w(x) \log w(x) p_1(x)\,\mathrm{d}x < \infty.$$

Applying Nutz (2022, Theorem 2.1) and Léonard (2014, Proposition 2.3) to (p_0, p_1^r) yields the unique tilted bridge P^r, with

$$\mathrm{KL}(P^r \| R) < \infty, \qquad \frac{\mathrm{d}P^r}{\mathrm{d}R} = \frac{\widehat{\varphi}_0^r(X_0)}{p_0(X_0)} \varphi_1^r(X_1) > 0,$$

where $(\varphi_t^r, \widehat{\varphi}_t^r)$ are its potentials relative to R.

For any path law Q with marginals p_0, p_1^r that is absolutely continuous with respect to R, the chain rule gives

$$\begin{aligned} \mathrm{KL}(Q\|P) &= \mathrm{KL}(Q\|R) - \mathbb{E}_Q \log \frac{\mathrm{d}P}{\mathrm{d}R} \\ &= \mathrm{KL}(Q\|R) + \int \log \varphi_0^{\mathrm{old}}(x) p_0(x)\,\mathrm{d}x \\ &\quad - \int \log \varphi_1^{\mathrm{old}}(x) p_1^r(x)\,\mathrm{d}x \\ &= \mathrm{KL}(Q\|R) + C(p_0, p_1, r), \end{aligned} \tag{23}$$

where $C(p_0, p_1, r)$ is constant, which does not depend on Q. Therefore both Schrödinger Bridge Problmes do have the same unique minimizer P^r:

$$\mathrm{SB}_P(p_0, p_1^r) = \mathrm{SB}_R(p_0, p_1^r) = P^r. \tag{24}$$

□

An analogous change-of-reference argument appears in Léonard (2014, equation (2.6)): reweighting the reference by endpoint factors shifts the KL objective by terms that are constant when the marginals are fixed.

A.2 Proof of Proposition 2

Proof. Let $(\varphi_t^r, \widehat{\varphi}_t^r)$ and $(\psi_t^r, \widehat{\psi}_t^r)$ be the potentials of P^r relative to R and P, respectively. The Eq 10 and Proposition 1 give:

$$\frac{\mathrm{d}P}{\mathrm{d}R} = \frac{\widehat{\varphi}_0^{\mathrm{old}}(X_0)}{p_0(X_0)}\varphi_1^{\mathrm{old}}(X_1), \qquad \frac{\mathrm{d}P^r}{\mathrm{d}R} = \frac{\widehat{\varphi}_0^r(X_0)}{p_0(X_0)}\varphi_1^r(X_1),$$
$$\frac{\mathrm{d}P^r}{\mathrm{d}P} = \frac{\widehat{\varphi}_0^r(X_0)}{\widehat{\varphi}_0^{\mathrm{old}}(X_0)}\frac{\varphi_1^r(X_1)}{\varphi_1^{\mathrm{old}}(X_1)} = \frac{\widehat{\psi}_0^r(X_0)}{p_0(X_0)}\psi_1^r(X_1).$$

Since $p_{0,1}^P$ and $p_0 \otimes p_1$ have the same null sets, separation of the endpoint factors implies, for some constant $c > 0$,

$$\psi_1^r(x) = c\,\frac{\varphi_1^r(x)}{\varphi_1^{\mathrm{old}}(x)} \qquad p_1^r\text{-almost everywhere.}$$

Using the terminal marginal identities,

$$\psi_1^r\widehat{\psi}_1^r = p_1^r, \qquad \varphi_1^r\widehat{\varphi}_1^r = p_1^r, \qquad \varphi_1^{\mathrm{old}}\widehat{\varphi}_1^{\mathrm{old}} = p_1,$$

we obtain

$$\begin{aligned} f_{\mathrm{rel}}^r &= -\log\psi_1^r = -\log\varphi_1^r + \log\varphi_1^{\mathrm{old}} - \log c \\ &= \log\widehat{\varphi}_1^r - \log\widehat{\varphi}_1^{\mathrm{old}} + \log\frac{p_1}{p_1^r} - \log c \\ &= \log\widehat{\varphi}_1^r - \log\widehat{\varphi}_1^{\mathrm{old}} - r + C, \end{aligned}$$

where the last equality uses $p_1^r \propto p_1 e^r$ and C absorbs the target and potential normalizations. This proves Eq 15 p_1^r-almost everywhere. Under the stated smoothness conditions, continuity extends the identity throughout the open region and differentiation gives Eq 16. □

A.3 Proof of Proposition 3

Proof. With the terminal terms evaluated at X_1, Proposition 2 shows that the objective in Eq 18 differs by a control-independent constant from

$$\mathbb{E}_{P^u}\left[\frac{1}{2}\int_0^1 \|u_t(X_t)\|^2\,\mathrm{d}t + \log\frac{\widehat{\psi}_1^r(X_1)}{p_1^r(X_1)}\right]. \tag{25}$$

Applying Eq 9 with reference P and target p_1^r, over its admissible controls, then Proposition 1, gives

$$P^{u_r^\star} = \mathrm{SB}_P(p_0, p_1^r) = \mathrm{SB}_R(p_0, p_1^r). \tag{26}$$

□

A.4 Assumptions and controlled diffusions for convergence

We introduce the forward and backward controlled diffusions used in the proof of Theorem 1:

$$\begin{aligned} P^u:&\quad \mathrm{d}X_t = [b_t(X_t) + \sigma_t u_t(X_t)]\,\mathrm{d}t + \sigma_t\,\mathrm{d}W_t, && X_0 \sim p_0, \\ Q^v:&\quad \mathrm{d}Y_s = [-f_{1-s}(Y_s) + \sigma_{1-s}v_s(Y_s)]\,\mathrm{d}s + \sigma_{1-s}\,\mathrm{d}W_s, && Y_0 \sim p_1^r. \end{aligned} \tag{27}$$

Here $s = 1 - t$ and $Y_s = X_{1-s}$. All density subscripts and KL divergences use forward time t, so $p_0^{P^u} = p_0$ and $p_1^{Q^v} = p_1^r$. All optimization variables below are controls.

We retain the reference conditions in the notation paragraph and Assumption 1. The following additional assumptions specify the setting of the two half-bridge results and the proof of Theorem 1. The convergence of controls in that theorem is understood in the integrated mean-square sense of Eq 65.

Assumption 2 (Population matching and convergence)**.**

1. Endpoint moments and reward. *For some* $\lambda > 0$ *and finite constant* $r_{\max}$,

$$\int e^{\lambda\|x\|^2}\big(p_0(x) + p_1(x)\big)\,\mathrm{d}x < \infty, \qquad r \in C^1(\mathbb{R}^d), \qquad r(x) \le r_{\max}. \tag{28}$$

2. Reference transition scores. *The density $p^R_{1|0}$ is positive and continuously differentiable in both endpoints, and, for a constant $C < \infty$ independent of the endpoints,*

$$\|\nabla_{x_0} \log p^R_{1|0}(x_1 \mid x_0)\| + \|\nabla_{x_1} \log p^R_{1|0}(x_1 \mid x_0)\| \le C(1 + \|x_0\| + \|x_1\|). \tag{29}$$

3. Admissibility and change of drift. *The controlled SDEs below have unique nonexplosive solutions. The admissible control classes contain the half-bridge solutions and the optimal relative controller u^*. For each comparison of P^u with $P^{\widetilde{u}}$ used below,*

$$\begin{aligned} \mathbb{E}_{P^{\widetilde{u}}} \exp\left(\frac{1}{2}\int_0^1 \|u_t(X_t) - \widetilde{u}_t(X_t)\|^2 \,\mathrm{d}t\right) < \infty, \\ \mathbb{E}_{P^u} \int_0^1 \|u_t(X_t) - \widetilde{u}_t(X_t)\|^2 \,\mathrm{d}t < \infty. \end{aligned} \tag{30}$$

These comparisons are $(u, \widetilde{u}) = (u, 0)$ for the SOC candidates and $(u^*, u^{(k)})$ for the final control-energy identity. The analogous conditions are imposed on R^u relative to R when using the control formulation Eq 7.

4. Differentiation and matching targets. *The pretrained drift b_t is continuously differentiable in space, and the adjoint ODE below is well posed. The propagated potentials have the classical time and space derivatives used in their diffusion representations on $0 < t < 1$, with the endpoint gradients used below. For every $k \ge 0$ and compact $K \subset \mathbb{R}^d$, there is a nonnegative integrable function $D_{k,K}$ such that*

$$\begin{aligned} \sup_{x_1 \in K} \|\nabla_{x_1} p^R_{1|0}(x_1 \mid x_0)\| \widehat{\varphi}^{(k)}_0(x_0) \le D_{k,K}(x_0), \\ \int D_{k,K}(x_0) \,\mathrm{d}x_0 < \infty. \end{aligned} \tag{31}$$

Here $\widehat{\varphi}^{(k)}_0$ is the stage factor in Eq 35. For the lean adjoint $\widetilde{a}^{(k)}$ in Eq 34,

$$\mathbb{E}_{P^{u^{(k+1)}}}\left[\int_0^1 \sigma_t^2 \|\widetilde{a}^{(k)}_t\|^2 \,\mathrm{d}t + \|\nabla_{x_1} \log p^R_{1|0}(X_1 \mid X_0)\|^2\right] < \infty. \tag{32}$$

The domination and moment conditions justify the kernel differentiation and conditional regressions in the half-bridge proofs; Eq 30 justifies their control-energy identities. The endpoint moments and score bound are used in the convergence proof.

In particular, for $w := p^r_1/p_1$, Assumption 1 gives a finite M such that

$$0 < w(x) \le M, \qquad \int e^{\lambda\|x\|^2} p^r_1(x) \,\mathrm{d}x \le M \int e^{\lambda\|x\|^2} p_1(x) \,\mathrm{d}x < \infty. \tag{33}$$

For this derivation, index a complete stage by $(u^{(k)}, h^{(k)}) \mapsto (u^{(k+1)}, h^{(k+1)})$, starting at $u^{(0)} = 0$, $h^{(0)} = h_{\mathrm{old}}$. Its exact population updates are

$$\begin{aligned} u^{(k+1)}_t(x) &= -\sigma_t \mathbb{E}_{P^{u^{(k+1)}}}[\widetilde{a}^{(k)}_t \mid X_t = x], \\ \widetilde{a}^{(k)}_1 &= h^{(k)}(X_1) - h_{\mathrm{old}}(X_1) - \nabla r(X_1), \qquad -\frac{\mathrm{d}\widetilde{a}^{(k)}_t}{\mathrm{d}t} = \nabla_x b_t(X_t)^\top \widetilde{a}^{(k)}_t, \\ h^{(k+1)} &\in \arg\min_h \mathbb{E}_{P^{u^{(k+1)}}} \left\| h(X_1) - \nabla_{x_1} \log p^R_{1|0}(X_1 \mid X_0) \right\|^2. \end{aligned} \tag{34}$$

For an exact stage controller $u^{(k)}$, write $(\varphi^{(k)}_t, \widehat{\varphi}^{(k)}_t)$ for its potentials relative to R, so that

$$\begin{aligned} \frac{\mathrm{d}P^{u^{(k)}}}{\mathrm{d}R} &= \frac{\widehat{\varphi}^{(k)}_0(X_0)}{p_0(X_0)} \varphi^{(k)}_1(X_1), & p^{P^{u^{(k)}}}_t &= \varphi^{(k)}_t \widehat{\varphi}^{(k)}_t, \\ \varphi^{(k)}_t(x) &= \int p^R_{1|t}(y \mid x) \varphi^{(k)}_1(y) \,\mathrm{d}y, & \widehat{\varphi}^{(k)}_0 &= p_0/\varphi^{(k)}_0, \\ \widehat{\varphi}^{(k)}_t(x) &= \int p^R_{t|0}(x \mid y) \widehat{\varphi}^{(k)}_0(y) \,\mathrm{d}y, \quad b_t + \sigma_t u^{(k)}_t = f_t + \sigma_t^2 \nabla \log \varphi^{(k)}_t. \end{aligned} \tag{35}$$

At $k = 0$, these are the old potentials. The forward half-bridge proof below establishes this representation inductively at each subsequent stage; it does not assume that intermediate diffusions already have terminal marginal p_1^r.

Time reversal of $P^{u^{(k)}}$, as used in Liu et al. (2025a, proof of Theorem 4.2), gives the backward control

$$\begin{aligned} v_s^{(k)}(x) &= \frac{f_{1-s}(x) - b_{1-s}(x)}{\sigma_{1-s}} - u_{1-s}^{(k)}(x) + \sigma_{1-s}\nabla \log p_{1-s}^{P^{u^{(k)}}}(x) \\ &= \sigma_{1-s}\nabla \log \widehat{\varphi}_{1-s}^{(k)}(x). \end{aligned} \tag{36}$$

The diffusion $Q^{v^{(k)}}$ uses this control and starts from p_1^r in reverse time. Therefore it preserves the conditional trajectories given X_1:

$$\begin{aligned} Q^{v^{(k)}}(\cdot \mid X_1) &= P^{u^{(k)}}(\cdot \mid X_1), \\ \frac{\mathrm{d}Q^{v^{(k)}}}{\mathrm{d}P^{u^{(k)}}} &= \frac{p_1^r(X_1)}{p_1^{P^{u^{(k)}}}(X_1)}, \\ \frac{\mathrm{d}Q^{v^{(k)}}}{\mathrm{d}R} &= \frac{\widehat{\varphi}_0^{(k)}(X_0)}{p_0(X_0)} \frac{p_1^r(X_1)}{\widehat{\varphi}_1^{(k)}(X_1)}. \end{aligned} \tag{37}$$

Conditional expressions here refer to the same controlled diffusions, conditioned on their indicated endpoint.

A.5 Controller Matching and the forward half bridge

Proposition 4 (Controller Matching solves the forward half bridge)**.** *Under Assumption 2, the exact population Controller Update in Eq 34 satisfies*

$$u^{(k+1)} \in \arg\min_u \mathrm{KL}(P^u \| Q^{v^{(k)}}), \tag{38}$$

where $v^{(k)}$ is defined by Eq 36.

Proof. At stage k, $h^{(k)} = \nabla \log \widehat{\varphi}_1^{(k)}$: this holds at initialization and follows from Proposition 5 after each corrector update. Define the stage terminal cost

$$g^{(k)} := \log \widehat{\varphi}_1^{(k)} - \log \widehat{\varphi}_1^{\mathrm{old}} - r, \qquad \nabla g^{(k)} = h^{(k)} - h_{\mathrm{old}} - \nabla r.$$

Using Eq 37, $P^{u^{(0)}} = P$, and $p_1 = \varphi_1^{\mathrm{old}}\widehat{\varphi}_1^{\mathrm{old}}$, we obtain for a candidate controller u,

$$\begin{aligned} \mathrm{KL}(P^u \| Q^{v^{(k)}}) &= \mathrm{KL}(P^u \| P^{u^{(0)}}) + \mathbb{E}_{P^u} \log \frac{\mathrm{d}P^{u^{(0)}}/\mathrm{d}R}{\mathrm{d}Q^{v^{(k)}}/\mathrm{d}R} \\ &= \mathrm{KL}(P^u \| P^{u^{(0)}}) + \int p_0 \log \frac{\widehat{\varphi}_0^{\mathrm{old}}}{\widehat{\varphi}_0^{(k)}} \,\mathrm{d}x + \mathbb{E}_{P^u} \log \frac{\widehat{\varphi}_1^{(k)}(X_1)\varphi_1^{\mathrm{old}}(X_1)}{p_1^r(X_1)} \\ &= \mathbb{E}_{P^u}\left[\frac{1}{2}\int_0^1 \|u_t(X_t)\|^2 \,\mathrm{d}t + g^{(k)}(X_1)\right] + C_k. \end{aligned}$$

Here C_k is independent of u; the last equality uses $p_1^r \propto p_1 e^r$ and the control-energy identity for the reference $P^{u^{(0)}}$. Adjoint Matching characterizes the SOC optimizer through the self-consistent update Eq 34, with reference drift b and terminal gradient $\nabla g^{(k)}$ (Liu et al., 2025a, Appendix A.1, equations (26)–(27)). Thus

$$u^{(k+1)} \in \arg\min_u \mathrm{KL}(P^u \| Q^{v^{(k)}}).$$

To identify the resulting controlled diffusion explicitly, set

$$\begin{aligned} \varphi_1^{(k+1)} &:= \frac{p_1^r}{\widehat{\varphi}_1^{(k)}}, \qquad \varphi_t^{(k+1)}(x) := \int p_{1|t}^R(y \mid x)\varphi_1^{(k+1)}(y)\,\mathrm{d}y, \\ \widehat{\varphi}_0^{(k+1)} &:= \frac{p_0}{\varphi_0^{(k+1)}}, \qquad u_t^{(k+1)} = \sigma_t \nabla \log \frac{\varphi_t^{(k+1)}}{\varphi_t^{\mathrm{old}}}. \end{aligned} \tag{39}$$

The SOC endpoint tilt then yields

$$\frac{\mathrm{d}P^{u^{(k+1)}}}{\mathrm{d}R} = \frac{\varphi_1^{(k+1)}(X_1)}{\varphi_0^{(k+1)}(X_0)}, \qquad p_0^{Q^{v^{(k)}}} = \widehat{\varphi}_0^{(k)}\varphi_0^{(k+1)},$$
$$\frac{\mathrm{d}P^{u^{(k+1)}}}{\mathrm{d}Q^{v^{(k)}}} = \frac{p_0(X_0)}{p_0^{Q^{v^{(k)}}}(X_0)}.$$

This proves the stage representation Eq 35 and identifies the forward half bridge: the initial marginal is replaced by p_0, with conditional trajectories given X_0 preserved. □

A.6 Corrector Matching and the backward half bridge

Proposition 5 (Corrector Matching identifies the backward half bridge)**.** *Under Assumption 2, after the exact Controller Update, the backward control in Eq 36 satisfies*

$$v^{(k+1)} \in \arg\min_v \mathrm{KL}(Q^v \| P^{u^{(k+1)}}), \qquad v_0^{(k+1)} = \sigma_1 h^{(k+1)}, \tag{40}$$

where $h^{(k+1)}$ is the exact Corrector Update in Eq 34.

Proof. For the controller $u^{(k+1)}$ from Proposition 4, the conditional density of X_0 given $X_1 = x$ follows from Eq 35:

$$p_{0|1}^{P^{u^{(k+1)}}}(x_0 \mid x) = \frac{p_{1|0}^R(x \mid x_0)\widehat{\varphi}_0^{(k+1)}(x_0)}{\widehat{\varphi}_1^{(k+1)}(x)}.$$

The moment bound Eq 32 and domination Eq 31 give

$$\begin{aligned} h^{(k+1)}(x) &= \mathbb{E}_{P^{u^{(k+1)}}}\Big[\nabla_x \log p_{1|0}^R(x \mid X_0) \mid X_1 = x\Big] \\ &= \frac{\int \nabla_x p_{1|0}^R(x \mid x_0)\widehat{\varphi}_0^{(k+1)}(x_0)\,\mathrm{d}x_0}{\widehat{\varphi}_1^{(k+1)}(x)} \\ &= \nabla \log \widehat{\varphi}_1^{(k+1)}(x) = \frac{v_0^{(k+1)}(x)}{\sigma_1}. \end{aligned}$$

The full backward control is given by Eq 36. For any candidate v, the KL chain rule at X_1 gives

$$\begin{aligned} \mathrm{KL}(Q^v \| P^{u^{(k+1)}}) &= \mathrm{KL}(p_1^r \| p_1^{P^{u^{(k+1)}}}) \\ &\quad + \int p_1^r(x)\,\mathrm{KL}\Big(Q^v(\cdot \mid X_1 = x) \| P^{u^{(k+1)}}(\cdot \mid X_1 = x)\Big)\,\mathrm{d}x \\ &\geq \mathrm{KL}(p_1^r \| p_1^{P^{u^{(k+1)}}}). \end{aligned}$$

By Eq 37, equality is attained by $v = v^{(k+1)}$, proving Eq 40. □

A.7 Proof of Theorem 1

We prove the theorem under Assumption 2.

Proof. Let u^* be the optimal relative controller, so that $P^{u^*} = \mathrm{SB}_P(p_0, p_1^r)$. The initialization gives

$$P^{u^{(0)}} = P, \qquad v_s^{(0)} = \sigma_{1-s}\nabla \log \widehat{\varphi}_{1-s}^{\mathrm{old}}, \qquad \frac{\mathrm{d}Q^{v^{(0)}}}{\mathrm{d}P^{u^{(0)}}} = \frac{p_1^r(X_1)}{p_1(X_1)}. \tag{41}$$

Propositions 4–5 yield

$$\begin{aligned} \frac{\mathrm{d}P^{u^{(k+1)}}}{\mathrm{d}Q^{v^{(k)}}} &= \frac{p_0(X_0)}{p_0^{Q^{v^{(k)}}}(X_0)}, \\ \frac{\mathrm{d}Q^{v^{(k+1)}}}{\mathrm{d}P^{u^{(k+1)}}} &= \frac{p_1^r(X_1)}{p_1^{P^{u^{(k+1)}}}(X_1)}. \end{aligned} \tag{42}$$

In particular, Eq 42 gives the endpoint updates

$$\begin{aligned} p_{0,1}^{Q^{v^{(k)}}}(x_0, x_1) &= p_{0,1}^{P^{u^{(k)}}}(x_0, x_1) \frac{p_1^r(x_1)}{p_1^{P^{u^{(k)}}}(x_1)}, \\ p_{0,1}^{P^{u^{(k+1)}}}(x_0, x_1) &= p_{0,1}^{Q^{v^{(k)}}}(x_0, x_1) \frac{p_0(x_0)}{p_0^{Q^{v^{(k)}}}(x_0)}. \end{aligned} \tag{43}$$

The stage factors in Eq 35 give the endpoint densities relative to the same reference kernel:

$$\begin{aligned} p_{0,1}^{P^{u^{(k)}}}(x_0, x_1) &= p_{1|0}^R(x_1 \mid x_0) \widehat{\varphi}_0^{(k)}(x_0) \varphi_1^{(k)}(x_1), \\ p_{0,1}^{P^{u^{(0)}}}(x_0, x_1) &= p_{1|0}^R(x_1 \mid x_0) \widehat{\varphi}_0^{\mathrm{old}}(x_0) \varphi_1^{\mathrm{old}}(x_1). \end{aligned} \tag{44}$$

Thus the log-factors relative to the pretrained diffusion are

$$\alpha^{(k)} := \log \frac{\widehat{\varphi}_0^{(k)}}{\widehat{\varphi}_0^{\mathrm{old}}}, \qquad \beta^{(k)} := \log \frac{\varphi_1^{(k)}}{\varphi_1^{\mathrm{old}}}, \qquad \alpha^{(0)} = \beta^{(0)} = 0. \tag{45}$$

Dividing the two densities in Eq 44 cancels their common kernel:

$$\begin{aligned} \frac{p_{0,1}^{P^{u^{(k)}}}(x_0, x_1)}{p_{0,1}^{P^{u^{(0)}}}(x_0, x_1)} &= \frac{\widehat{\varphi}_0^{(k)}(x_0)}{\widehat{\varphi}_0^{\mathrm{old}}(x_0)} \frac{\varphi_1^{(k)}(x_1)}{\varphi_1^{\mathrm{old}}(x_1)} \\ &= e^{\alpha^{(k)}(x_0) + \beta^{(k)}(x_1)}. \end{aligned} \tag{46}$$

For the two marginal normalizations, Eq 35 gives

$$\begin{aligned} \int p_{0,1}^{P^{u^{(0)}}}(x_0, x_1) e^{\alpha^{(k)}(x_0)} \, \mathrm{d}x_0 &= \varphi_1^{\mathrm{old}}(x_1) \int p_{1|0}^R(x_1 \mid x_0) \widehat{\varphi}_0^{(k)}(x_0) \, \mathrm{d}x_0 \\ &= \varphi_1^{\mathrm{old}}(x_1) \widehat{\varphi}_1^{(k)}(x_1), \\ \int p_{0,1}^{P^{u^{(0)}}}(x_0, x_1) e^{\beta^{(k+1)}(x_1)} \, \mathrm{d}x_1 &= \widehat{\varphi}_0^{\mathrm{old}}(x_0) \int p_{1|0}^R(x_1 \mid x_0) \varphi_1^{(k+1)}(x_1) \, \mathrm{d}x_1 \\ &= \widehat{\varphi}_0^{\mathrm{old}}(x_0) \varphi_0^{(k+1)}(x_0). \end{aligned} \tag{47}$$

The endpoint replacements in Eq 42, with the factor normalization fixed in Eq 39, therefore yield

$$\begin{aligned} \beta^{(k+1)}(x_1) &= \log \frac{\varphi_1^{(k+1)}(x_1)}{\varphi_1^{\mathrm{old}}(x_1)} = \log \frac{p_1^r(x_1)}{\widehat{\varphi}_1^{(k)}(x_1) \varphi_1^{\mathrm{old}}(x_1)} \\ &= \log p_1^r(x_1) - \log \int p_{0,1}^{P^{u^{(0)}}}(x_0, x_1) e^{\alpha^{(k)}(x_0)} \, \mathrm{d}x_0, \\ \alpha^{(k+1)}(x_0) &= \log \frac{\widehat{\varphi}_0^{(k+1)}(x_0)}{\widehat{\varphi}_0^{\mathrm{old}}(x_0)} = \log \frac{p_0(x_0)}{\varphi_0^{(k+1)}(x_0) \widehat{\varphi}_0^{\mathrm{old}}(x_0)} \\ &= \log p_0(x_0) - \log \int p_{0,1}^{P^{u^{(0)}}}(x_0, x_1) e^{\beta^{(k+1)}(x_1)} \, \mathrm{d}x_1. \end{aligned} \tag{48}$$

By Eq 41, $p_{0,1}^{Q^{v^{(0)}}} = w p_{0,1}^{P^{u^{(0)}}}$. Shifting the terminal factor in Eq 48 gives

$$\begin{aligned} \widetilde{\beta}^{(k)}(x_1) &:= \beta^{(k)}(x_1) - \log w(x_1), \\ \widetilde{\beta}^{(k+1)}(x_1) &= \log p_1^r(x_1) - \log \int p_{0,1}^{Q^{v^{(0)}}}(x_0, x_1) e^{\alpha^{(k)}(x_0)} \, \mathrm{d}x_0, \\ \alpha^{(k+1)}(x_0) &= \log p_0(x_0) - \log \int p_{0,1}^{Q^{v^{(0)}}}(x_0, x_1) e^{\widetilde{\beta}^{(k+1)}(x_1)} \, \mathrm{d}x_1. \end{aligned} \tag{49}$$

Since $p_1^{Q^{v^{(0)}}} = p_1^r$, Eq 49 gives $\widetilde{\beta}^{(1)} = 0$. These are the Sinkhorn updates initialized at the first backward half bridge, with endpoint cost

$$c(x_0, x_1) := \log \frac{p_0(x_0) p_1(x_1)}{p_{0,1}^{P^{u^{(0)}}}(x_0, x_1)}. \tag{50}$$

To verify the growth conditions for the endpoint cost in Eq 50, set $c_R(x_0, x_1) := -\log p^R_{1|0}(x_1 \mid x_0)$. Integrating Eq 29 along line segments gives

$$\begin{aligned} |c_R(x_0, x_1) - c_R(0, x_1)| &\le C'\big(\|x_0\| + \|x_0\|^2 + \|x_0\|\|x_1\|\big), \\ |c_R(0, x_1)| &\le C'(1 + \|x_1\|^2). \end{aligned} \tag{51}$$

The same bounds hold with the endpoints exchanged. The old Schrödinger system Eq 10 can be written as

$$\begin{aligned} -\log \varphi_0^{\mathrm{old}}(x_0) &= -\log \int e^{-c_R(x_0,x_1) - \log \widehat{\varphi}_1^{\mathrm{old}}(x_1)} p_1(x_1)\,\mathrm{d}x_1, \\ -\log \widehat{\varphi}_1^{\mathrm{old}}(x_1) &= -\log \int e^{-c_R(x_0,x_1) - \log \varphi_0^{\mathrm{old}}(x_0)} p_0(x_0)\,\mathrm{d}x_0. \end{aligned} \tag{52}$$

Assumption 1 supplies finite entropy; Eq 28 and Eq 51 supply the moment and growth hypotheses. Applied to Eq 52, Ghosal & Nutz (2025, Lemma 5.2 and Theorem 5.7) give, after fixing the additive constant,

$$|\log \varphi_0^{\mathrm{old}}(x_0)| \le C_0(1 + \|x_0\|^2), \qquad |\log \widehat{\varphi}_1^{\mathrm{old}}(x_1)| \le C_0(1 + \|x_1\|^2). \tag{53}$$

Substitution of Eq 44 and the terminal identity $p_1 = \varphi_1^{\mathrm{old}} \widehat{\varphi}_1^{\mathrm{old}}$ into Eq 50 gives

$$\begin{aligned} c(x_0, x_1) &= c_R(x_0, x_1) + \log \varphi_0^{\mathrm{old}}(x_0) + \log \widehat{\varphi}_1^{\mathrm{old}}(x_1) \\ &= a_0(x_0) + a_1(x_1) + \widehat{c}(x_0, x_1), \\ a_0(x_0) &:= \log \varphi_0^{\mathrm{old}}(x_0), \\ a_1(x_1) &:= c_R(0, x_1) + \log \widehat{\varphi}_1^{\mathrm{old}}(x_1), \\ \widehat{c}(x_0, x_1) &:= c_R(x_0, x_1) - c_R(0, x_1). \end{aligned} \tag{54}$$

By Eq 51 and Eq 53, the terms in Eq 54 satisfy $|a_i(x)| \le C_2(1+\|x\|^2)$ for $i = 0, 1$, and $|\widehat{c}(x_0, x_1)| \le C'(\|x_0\| + \|x_0\|^2 + \|x_0\|\|x_1\|)$. Eq 50 and these bounds, together with the endpoint moments in Eq 28 and Eq 33, give

$$e^{-c(x_0,x_1)} p_0(x_0) p_1^r(x_1) = p_{0,1}^{Q^{v^{(0)}}}(x_0, x_1), \qquad \mathrm{KL}(p_0 \otimes p_1^r \| p_{0,1}^{Q^{v^{(0)}}}) = \iint c\, p_0 p_1^r < \infty. \tag{55}$$

Eq 55 identifies the Sinkhorn initialization with the first backward half bridge and supplies a finite-entropy feasible coupling. Let α^*, β^* be the optimal relative endpoint factors, so that

$$\begin{aligned} p_{0,1}^{P^{u^*}}(x_0, x_1) &= p_{0,1}^{P^{u^{(0)}}}(x_0, x_1) e^{\alpha^*(x_0) + \beta^*(x_1)} \\ &= p_{0,1}^{Q^{v^{(0)}}}(x_0, x_1) e^{\alpha^*(x_0) + \widetilde{\beta}^*(x_1)}, \\ \widetilde{\beta}^* &:= \beta^* - \log w. \end{aligned} \tag{56}$$

Eq 50–Eq 54 establish quadratic growth of the separate endpoint terms and the required bound on their interaction. Together with Eq 55 and the sub-Gaussian moments in Eq 28 and Eq 33, these results verify the hypotheses of Ghosal & Nutz (2025, Theorem 5.7 and Corollary 5.8). Applied to the iterates in Eq 49 and the optimal factor in Eq 56, those results give the following uniform quadratic bound, which implies the exponential-moment bound:

$$\begin{aligned} |\alpha^{(k)}(x)| + |\alpha^*(x)| &\le C_3(1 + \|x\|^2), \qquad k \ge 0, \\ M_\gamma := \sup_{k \ge 0} \int e^{\gamma |\alpha^{(k)} - \alpha^*|} p_0\,\mathrm{d}x &\le e^{\gamma C_3} \int e^{\lambda \|x\|^2} p_0(x)\,\mathrm{d}x < \infty, \qquad 0 < \gamma C_3 \le \lambda. \end{aligned} \tag{57}$$

The constant $C_3 > 0$ is independent of k.

Define the endpoint entropies

$$\begin{aligned} F_k &:= \mathrm{KL}\left(p_{0,1}^{P^{u^*}} \| p_{0,1}^{P^{u^{(k)}}}\right), \quad S_k := \mathrm{KL}\left(p_{0,1}^{P^{u^*}} \| p_{0,1}^{Q^{v^{(k)}}}\right), \\ d_k &:= \mathrm{KL}(p_1^r \| p_1^{P^{u^{(k)}}}), \qquad m_k := \mathrm{KL}(p_0^{Q^{v^{(k)}}} \| p_0). \end{aligned} \tag{58}$$

Proposition 1 and the endpoint KL chain rule give $F_0 \le \mathrm{KL}(P^{u^*} \| P) < \infty$. Using the definitions in Eq 58, the endpoint updates Eq 43 imply

$$\begin{aligned} F_k - S_k &= \iint p_{0,1}^{P^{u^*}} \log \frac{p_{0,1}^{Q^{v^{(k)}}}}{p_{0,1}^{P^{u^{(k)}}}} \,\mathrm{d}x_0 \mathrm{d}x_1 = \int p_1^r \log \frac{p_1^r}{p_1^{P^{u^{(k)}}}} \,\mathrm{d}x_1 = d_k, \\ S_k - F_{k+1} &= \int p_0 \log \frac{p_0}{p_0^{Q^{v^{(k)}}}} \,\mathrm{d}x_0 = \mathrm{KL}(p_0 \| p_0^{Q^{v^{(k)}}}) \ge 0, \\ \sum_{k=0}^{n} d_k &\le F_0 - F_{n+1} \le F_0 < \infty. \end{aligned} \tag{59}$$

Eq 59 implies $d_k \to 0$. Marginalizing the terminal replacement in Eq 43 then bounds m_k from Eq 58:

$$\begin{aligned} 0 \le m_k &\le \mathrm{KL}\left(p_{0,1}^{Q^{v^{(k)}}} \| p_{0,1}^{P^{u^{(k)}}} \right) \\ &= \int p_1^r \log \frac{p_1^r}{p_1^{P^{u^{(k)}}}} \,\mathrm{d}x_1 = d_k \longrightarrow 0. \end{aligned} \tag{60}$$

The entropy inequality and Eq 57 give

$$\int |\alpha^{(k)} - \alpha^*| p_0^{Q^{v^{(k)}}} \,\mathrm{d}x \le \gamma^{-1}(m_k + \log M_\gamma) < \infty. \tag{61}$$

Eq 57 ensures that $C_4 := 2\gamma^{-1}(3/2 + \log M_\gamma)$ is finite and independent of k. Combining the stage factorization Eq 46 with the terminal replacement Eq 43 and the optimal factorization Eq 56 gives

$$\begin{aligned} p_{0,1}^{Q^{v^{(k)}}}(x_0, x_1) &= p_{0,1}^{Q^{v^{(0)}}}(x_0, x_1) e^{\alpha^{(k)}(x_0) + \widetilde{\beta}^{(k+1)}(x_1)}, \\ \log \frac{p_{0,1}^{Q^{v^{(k)}}}(x_0, x_1)}{p_{0,1}^{P^{u^*}}(x_0, x_1)} &= \alpha^{(k)}(x_0) - \alpha^*(x_0) + \widetilde{\beta}^{(k+1)}(x_1) - \widetilde{\beta}^*(x_1). \end{aligned} \tag{62}$$

Both laws in Eq 62 have terminal marginal p_1^r. Conditioning on X_1, their terminal factor cancels in the symmetric KL; Eq 61 justifies the remaining integral. With $F_{k+1} \le S_k$ from Eq 59 and $m_k \to 0$ from Eq 60, we obtain

$$\begin{aligned} 0 \le F_{k+1} \le S_k &\le \mathrm{KL}\left(p_{0,1}^{Q^{v^{(k)}}} \| p_{0,1}^{P^{u^*}} \right) + S_k \\ &= \int (\alpha^{(k)} - \alpha^*)(p_0^{Q^{v^{(k)}}} - p_0) \,\mathrm{d}x \\ &\le C_4 \left(\sqrt{m_k} + \frac{1}{2} m_k \right) \longrightarrow 0. \end{aligned} \tag{63}$$

The last inequality is the weighted entropy estimate of Ghosal & Nutz (2025, Lemma 3.1), with its exponential-moment hypothesis verified in Eq 57. This proves endpoint entropy convergence directly from the two half-bridge updates.

Eq 63 proves $F_k \to 0$, where F_k is the endpoint KL defined in Eq 58. Each multiplier in Eq 42 depends only on the endpoints, and the optimal bridge also preserves the pretrained conditional trajectories given both endpoints. The conditional term in the KL chain rule therefore vanishes, so

the same bound gives path-space convergence, for $k \geq 1$:

$$\begin{aligned} P^{u^{(k)}}(\cdot \mid X_0, X_1) &= Q^{v^{(k)}}(\cdot \mid X_0, X_1) = P^{u^{(0)}}(\cdot \mid X_0, X_1), \\ P^{u^*}(\cdot \mid X_0, X_1) &= P^{u^{(0)}}(\cdot \mid X_0, X_1), \\ \mathrm{KL}(P^{u^*} \| P^{u^{(k)}}) &= \mathrm{KL}\Big(p_{0,1}^{P^{u^*}} \| p_{0,1}^{P^{u^{(k)}}}\Big) \\ &\qquad + \mathbb{E}_{P^{u^*}} \underbrace{\mathrm{KL}\Big(P^{u^*}(\cdot \mid X_0, X_1) \| P^{u^{(k)}}(\cdot \mid X_0, X_1)\Big)}_{=0} \\ &= F_k \leq C_4 \left(\sqrt{m_{k-1}} + \frac{1}{2} m_{k-1}\right) \longrightarrow 0. \end{aligned} \tag{64}$$

This is the endpoint-to-diffusion lifting described by De Bortoli et al. (2021, Section 3.5).

Finally, both controlled diffusions start from p_0 and have volatility σ_t. The control-energy identity justified by Eq 30, together with the KL convergence in Eq 64, gives

$$\begin{aligned} \mathbb{E}_{P^{u^*}} \int_0^1 \|u_t^{(k)}(X_t) - u_t^*(X_t)\|^2 \, \mathrm{d}t = 2\mathrm{KL}(P^{u^*} \| P^{u^{(k)}}) \longrightarrow 0, \\ P^{u^*} = \mathrm{SB}_P(p_0, p_1^r) \overset{\text{Proposition 1}}{=} \mathrm{SB}_R(p_0, p_1^r). \end{aligned} \tag{65}$$

Thus the controllers converge in integrated mean square under the optimal controlled diffusion, with the limit solving the tilted SB problem. □

B COMPETITORS

Shared pretrained sampler. All competitors use the frozen bridge P. For source x_0 and grid $0 = t_0 < \cdots < t_L = 1$, its Euler step is

$$X_{i+1} = X_i + \Delta_i b_{t_i}(X_i) + \sigma \sqrt{\Delta_i}\, \xi_i, \qquad \Delta_i = t_{i+1} - t_i, \quad \xi_i \sim \mathcal{N}(0, I). \tag{66}$$

Image runs omit noise in the final step. Let r denote the terminal log reward: $r = -\lambda E$ for images and the terminal log density ratio for the two-dimensional experiment. The pretrained drift predicts the endpoint as

$$\widehat{x}_1(x, t) = x + (1 - t) b_t(x), \qquad \widehat{x}_1(x, 1) = x. \tag{67}$$

B.1 REWARD TILTING WITH NON-MEMORYLESS PRIORS

Reward finetuning for the memoryless diffusion models Song et al. (2021) is widely explored research field Uehara et al. (2024); Domingo-Enrich et al. (2025); Black et al. (2024). A lot of memoryless diffusion reward finetuning methodology is based on a fact of independence of start of generation X_0 and the end of generation X_1, i.e., memorylessness property:

Let R_{diff} be a memoryless diffusion with marginals p_0 and p_1:

$$\begin{aligned} R_{\mathrm{diff}}: \quad & \mathrm{d}X_t = f_t^{\mathrm{diff}}(X_t)\, \mathrm{d}t + \sigma_t^{\mathrm{diff}}\, \mathrm{d}W_t, \\ & p_{0,1}^{R_{\mathrm{diff}}}(x_0, x_1) = p_0(x_0) p_1(x_1). \end{aligned} \tag{68}$$

Then considering the diffusion bridges Zhou et al. (2024), which in general do not have a memoryless property:

$$\begin{aligned} R_{\mathrm{bridge}}: \quad & \mathrm{d}X_t = b_t(X_t)\, \mathrm{d}t + \sigma_t\, \mathrm{d}W_t, \\ & p_{0,1}^{R_{\mathrm{bridge}}}(x_0, x_1) = p_0(x_0) p_{1|0}^{R_{\mathrm{bridge}}}(x_1 \mid x_0) \neq p_0(x_0) p_1(x_1) \quad \text{in general.} \end{aligned} \tag{69}$$

For either reference $S \in \{R_{\mathrm{diff}}, R_{\mathrm{bridge}}\}$, let S^r be the SOC solution with terminal reward r in Eq 3. When the controlled solution exists and the conditional exponential moment is finite and positive,

Eq 4 gives:

$$V_0^S(x_0) = -\log \int p_{1|0}^S(y \mid x_0) e^{r(y)} \, \mathrm{d}y,$$
$$p_{0,1}^{S^r}(x_0, x_1) = p_0(x_0) p_{1|0}^S(x_1 \mid x_0) e^{r(x_1) + V_0^S(x_0)}. \tag{70}$$

Write $\bar{Z}_r = \int p_1(y) e^{r(y)} \, \mathrm{d}y$, so that $p_1^r \propto p_1 e^r$. For the memoryless diffusion, $V_0^{R_{\mathrm{diff}}}(x_0) = -\log \bar{Z}_r$, and therefore the SOC solution marginalization gives unbiased reward tilted x_1 distribution $p_1^{R_{\mathrm{diff}}^r}(x_1) = p_1^r(x_1)$:

$$p_1^{R_{\mathrm{diff}}^r}(x_1) = \int p_0(x_0) p_1(x_1) e^{r(x_1) - \log \bar{Z}_r} \, \mathrm{d}x_0 = \frac{p_1(x_1) e^{r(x_1)}}{\bar{Z}_r} = p_1^r(x_1). \tag{71}$$

For the diffusion bridge, $V_0^{R_{\mathrm{bridge}}}(x_0)$ generally depends on the source. Marginalizing the SOC solution gives

$$\begin{aligned} p_1^{R_{\mathrm{bridge}}^r}(x_1) &= e^{r(x_1)} \int p_0(x_0) p_{1|0}^{R_{\mathrm{bridge}}}(x_1 \mid x_0) e^{V_0^{R_{\mathrm{bridge}}}(x_0)} \, \mathrm{d}x_0 \\ &= p_1(x_1) e^{r(x_1)} \mathbb{E}_{R_{\mathrm{bridge}}}\left[e^{V_0^{R_{\mathrm{bridge}}}(X_0)} \mid X_1 = x_1 \right] \\ &= p_1^r(x_1) \underbrace{\bar{Z}_r \, \mathbb{E}_{R_{\mathrm{bridge}}}\left[e^{V_0^{R_{\mathrm{bridge}}}(X_0)} \mid X_1 = x_1 \right]}_{\mathcal{B}_r(x_1)}. \end{aligned} \tag{72}$$

The factor $\mathcal{B}_r$ need not be constant, so the **reward alone does not produce p_1^r in general**.

B.2 DPS-style guidance

DPS differentiates a likelihood at a predicted endpoint (Chung et al., 2023a) and CDDB-deep applies this idea to diffusion bridges for inverse problems (Chung et al., 2023b). Our methodology is similar to CDDB, we guide the frozen bridge with the terminal reward:

$$X_{i+1} = X_i + \Delta_i b_{t_i}(X_i) + \gamma \sigma^2 \Delta_i \, \nabla_x r(\widehat{x}_1(x, t_i))|_{x = X_i} + \sigma \sqrt{\Delta_i} \, \xi_i. \tag{73}$$

γ is the guidance scale, separate from reward strength. We use the full Jacobian of $\widehat{x}_1$, with $\gamma \in \{0.5, 1, 2, 4\}$.

B.3 Conditional self-normalized importance sampling (CondSNIS)

For each fixed source x_0, CondSNIS samples a population of K independent old-bridge trajectories and targets:

$$p_{1|0}^r(y \mid x_0) \propto e^{r(y)} p_{1|0}^P(y \mid x_0) \qquad Z_r(x_0) = \mathbb{E}_P[e^{r(X_1)} \mid X_0 = x_0]. \tag{74}$$

For terminals $Y^1, \ldots, Y^K$, the weighted population is

$$\widehat{p}_{1|0}^r(y \mid x_0) = \sum_{j=1}^K w_j \delta_{Y^j}(\mathrm{d}y), \qquad w_j = \frac{e^{r(Y^j)}}{\sum_{k=1}^K e^{r(Y^k)}}, \tag{75}$$

where δ is the Dirac delta function. As the number of particles $N \to \infty$ this procedure results in conditional tilt $p_{1|0}^r(y \mid x_0) \propto e^{r(y)} p_{1|0}^P(y \mid x_0)$ Del Moral et al. (2006). At the end of the procedure we sample the single final sample with log weights w.

B.4 Conditional sequential Monte Carlo (CondSMC)

CondSMC propagates N old-bridge particles per source (Del Moral et al., 2006). For this base proposal, RNE's bridge-compatible path-ratio weights reduce to our potential ratios (He et al.,

2026). We define $\rho_i(x)$ as particle i score and ℓ_i as particle i log importance weights, where $\rho_i(x) = r(\widehat{x}_1(x, t_i))$, $\rho_L(x) = r(x)$ and $\ell_0^j = \rho_0(x_0)$. With s the preceding potential step, update:

$$\ell_i^j = \ell_s^j + \rho_i(X_i^j) - \rho_s(X_s^j), \qquad \mathrm{ESS}_i = \left(\sum_{j=1}^{N} \mathrm{softmax}(\ell_i)_j^2\right)^{-1}. \tag{76}$$

Each step resamples systematically if $\mathrm{ESS}_i < N/2$, resetting weights but retaining potentials. As the number of particles $N \to \infty$ this procedure results in conditional tilt $p^r_{1|0}(y \mid x_0) \propto e^{r(y)} p^P_{1|0}(y \mid x_0)$ Del Moral et al. (2006). At the end of the procedure we sample the single final sample with log weights l.

C General design choices for TSBM

§ 4 states the TSBM updates in terms of an incremental controller u and a full terminal corrector h. Here we describe the neural parameterization and sampling procedure used in the experiments.

Neural parameterization. DSBM pretraining (Shi et al., 2023) provides a forward drift b_ψ and an old-bridge corrector $h_{\mathrm{old},\psi}$. We freeze both networks during TSBM training. We initialize a trainable full forward drift b_θ and full corrector h_ϕ from these pretrained networks, then carry their parameters forward between stages. The incremental controller in § 4 is represented implicitly by

$$u_\theta(x, t) = \frac{b_\theta(x, t) - b_\psi(x, t)}{\sigma_t}. \tag{77}$$

Thus, simulating the current process requires one evaluation of b_θ per time step rather than separate evaluations of b_ψ and a controller network. The frozen b_ψ is still used to form the controller-matching loss and the lean-adjoint target. Likewise, the frozen $h_{\mathrm{old},\psi}$ is subtracted from h_ϕ in the terminal adjoint condition.

Replay buffers. We maintain a buffer of N sampling trajectories and their corresponding back-propagated "lean" adjoints from the current drift $b_\psi^{(k)}$ and replace it in full every L controller-gradient steps. Each step uses a uniformly sampled minibatch of stored trajectories. After updating the drift, we draw endpoint pairs (X_0, X_1) under that drift $b_\psi^{(k)}$ for the corrector $h_\phi^{(k)}$ update and do not refresh their buffer due to the start of the next stage $k + 1$.

Reference process. In all our experiments the reference process is Wiener, i.e., $R = W^\epsilon$. Which leads to $\sigma_t = \epsilon$ and $s_R(X_0, X_1) = -\frac{X_1 - X_0}{\epsilon^2}$.

Simulation and regression targets. We simulate the controlled SDE with drift b_θ by Euler-Maruyama. Along each sampled trajectory, we integrate the lean-adjoint ODE backward from

$$\widetilde{a}_1 = -\nabla r(X_1) + h_\phi(X_1) - h_{\mathrm{old},\psi}(X_1), \qquad -\frac{\mathrm{d}\widetilde{a}_t}{\mathrm{d}t} = \nabla_x b_\psi(X_t, t)^\top \widetilde{a}_t. \tag{78}$$

The Jacobian-vector product uses the frozen pretrained drift, not the trainable drift. We regress $b_\phi(X_t, t)^{(k)}$ against $b_\theta - \sigma_t^2 \widetilde{a}_t$ using Eq 6. For the corrector update, we regress $h_\phi(X_1)$ against the endpoint transition score $s_R(X_0, X_1) = s_{W^\epsilon}(X_0, X_1) = -\frac{X_1 - X_0}{\epsilon^2}$ of the original reference process W^ϵ, as in Eq 21.

D Reward Functions

D.1 Toy illustrative experiment

For the 2D toy experiment, we use the log-density-ratio reward

$$r(x) = \log \frac{p_1^{\mathrm{tilt}}(x)}{p_1(x)}, \quad p_1 = \sum_{k=1}^{4} \alpha_k \mathcal{N}(\mu_k, \sigma_k), \quad p_1^{\mathrm{tilt}} = \sum_{k=1}^{4} \alpha_k^{\mathrm{tilt}} \mathcal{N}(\mu_k, \sigma_k) \tag{79}$$

Algorithm 2 Tilted Schrödinger Bridge Matching (TSBM) Practical

Require: frozen old drift b_θ, old corrector $h_{\text{old},\theta}$, reference process R, reward r, stages K, number of controller updates M_{ctrl}, number of corrector updates M_{corr}, freq of Replay Buffer updates N

initialize $b_\psi^{(0)} \leftarrow b_\theta$, $h_\psi^{(0)} \leftarrow h_{\text{old},\theta}$, $S \leftarrow \emptyset$
for $k = 0, \dots, K-1$ **do**
  $b_\phi \leftarrow \text{copy}(b_\phi^{(k)})$
  for $m = 0, \dots, M_{\text{ctrl}} - 1$ **do**
    if $m \bmod N = 0$ **then**
      Sample *paths* $(X_t)_{t\in[0,1]}$ by Euler-Maruyama inference of the SDE with drift b_ϕ
      $\tilde{a}_1^{(k)} \leftarrow -\nabla r(X_1) + h_\phi^{(k)}(X_1) - h_{\text{old},\theta}(X_1)$
      Propagate $\tilde{a}_t^{(k)}$ by solving the backward lean-adjoint ODE using Eq 5
      Add paths $(X_t)_{t\in[0,1]}$ and lean adjoints $(\tilde{a}_t^{(k)})_{t\in[0,1]}$ to replay buffer S
    Sample *paths* $(X_t)_{t\in[0,1]}$ and lean adjoints $\tilde{a}$ from replay buffer S
    Make a gradient step on $\nabla_\phi \mathcal{L}_{\text{ctrl}}$ w.r.t b_ϕ parameters;
  Sample many *endpoints* (X_0, X_1) by the inference of SDE with drift b_ϕ;
  $h_\phi \leftarrow \text{copy}(h_\phi^{(k)})$
  for $m = 0, \dots, M_{\text{corr}} - 1$ **do**
    Sample a minibatch of endpoint pairs
    Make a gradient step on $\nabla_\phi \mathcal{L}_{\text{corr}}$ w.r.t b_ϕ parameters;
  $b_\phi^{(k+1)} \leftarrow b_\phi$, $h_\phi^{(k+1)} \leftarrow h_\phi$

return $u^{(K)}, h^{(K)}$

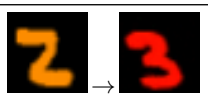
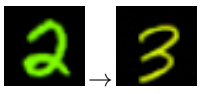
where the mixture weights change from $\alpha = (0.25, 0.25, 0.25, 0.25)$ to $\alpha^{\text{tilt}} = (0, 0.2, 0.2, 0.6)$. Thus, $p_1(x)e^{r(x)} = p_1^{\text{tilt}}(x)$, while the Gaussian means and covariances remain unchanged.

$$E(x) = -r(x) = \log p_1(x) - \log p_1^{\text{tilt}}(x) \tag{80}$$

D.2 Colored MNIST unpaired translation

This appendix gives the terminal energies used for the image experiments in Section 6.

Table 2: Colored MNIST terminal energies and desired outputs. Each pair shows a source x_0 (left) and the terminal TSBM generation from the same column of the corresponding comparison grid (right).

Reward	Intended terminal attribute	Desired output
Red chroma	A digit 3 with pronounced red chroma and a neutral background.	
Thin stroke (target 3.65)	A digit 3 with estimated stroke thickness close to 3.65.	
Thick stroke (target 7)	A digit 3 with estimated stroke thickness close to 7.	

Red-chroma reward. Let $x \in [-1, 1]^{3\times H\times W}$ and index spatial locations by i. The red-chroma score favors red image channel x_R relative to green and blue channels , i.e., x_G and x_B, while neutral

gray backgrounds do self cancel and do contribute to this reward:

$$c_{\mathrm{red}}(x) = \frac{1}{HW} \sum_i \left[x_{R,i} - \frac{x_{G,i} + x_{B,i}}{2} \right], \tag{81}$$

$$E_{\mathrm{red}}(x) = -c_{\mathrm{red}}(x). \tag{82}$$

Stroke-thickness energies. For normalized to $[0, 1]$ image $u = (x + 1)/2 \in [0, 1]^{3\times H\times W}$, we first construct a differentiable, digit stroke area estimate:

$$I_i(x) = \sum_c u_{c,i} \frac{\exp(\kappa_{\mathrm{rgb}} u_{c,i})}{\sum_{c'} \exp(\kappa_{\mathrm{rgb}} u_{c',i})}, \qquad M_i(x) = \sigma(\kappa_m [I_i(x) - t_{\mathrm{fg}}]), \tag{83}$$

$$A(x) = \sum_i M_i(x), \tag{84}$$

where I_i is a *brightness-like score* at pixel i, $M_i(x)$ is the *soft foreground mask*, $A(x)$ estimates the *stroke's area*. I_i gives more weight to whichever color channel is strongest, acting like a smooth maximum of red, green, and blue. $M_i(x)$ acts as the soft foregrouhnd mask, where the sigmoid turns I_i into a value near 0 for a dark background pixel and near 1 for a bright stroke pixel. Other parameters: $u_{c,i}$ is the color c channel of pixel i, $\kappa_{\mathrm{rgb}} = 20$ is the strength of concentration on the brightest channel, $\kappa_{\mathrm{m}} = 40$ controls how sharply the mask changes around that threshold and the $t_{\mathrm{fg}} = 0.1$ is the foreground threshold applied to I_i. These parameters were selected empirically.

$$P(x) = \sum_i \sqrt{(K_x * M)_i^2 + (K_y * M)_i^2 + \epsilon}, \qquad w(x) = \frac{2A(x)}{P(x) + \epsilon}. \tag{85}$$

Here P approximates *foreground perimeter* obtained by adding the strength of the mask's edges and $w(x)$ is the ratio between strokes area and foreground perimeter or **resulting stroke-width estimate**, measured roughly in pixels. Other parameters and functions are: K_x and K_y are the 3×3 Sobel filters, scaled by $1/8$ and applied with replicate padding, $\epsilon = 10^{-8}$ a tiny constant that prevents numerical problems at zero.

The **thin-stroke** and **thick-stroke** energies are:

$$r_{\mathrm{thin}}(x) = -\left(\frac{w(x) - 3.65}{0.9935646} \right)^2, \tag{86}$$

$$r_{\mathrm{thick}}(x) = -\left(\frac{w(x) - 7}{0.9935646} \right)^2, \tag{87}$$

which are just approximate stroke width $w(x)$ MSE towards targets, i.e., 3.65 and 7, normalized by 0.9935646, which is difference between the 25th and 75th quantiles of $w(x)$ measure on 32x32 Colored MNIST dataset digit 3s.

D.3 CelebA unpaired translation

Let $S(x, p)$ denote the contribution of a preference model Xu et al. (2023); Kirstain et al. (2023); Wu et al. (2023b) for image x and prompt p. The prompt-conditioned reward can be written as:

$$r(x; p^+, p^-) = S(x, p^+) - S(x, p^-). \tag{88}$$

For a positive-only target we set $S(x, p^-) = 0$.

Prompts. The positive prompts used for the three 128×128 male-to-female translation experiments are quoted verbatim from the run configurations:

- *Natural Toothy Smile*: "a natural portrait photo of a woman with a relaxed smile and naturally visible teeth"

- *Heavy Makeup*: "a natural portrait photo with glamorous eye makeup, eyeliner, and lipstick"

- *Elderly Woman*: "a natural portrait of an elderly woman with visible age lines and mature facial features"

The elderly-woman reward additionally uses the negative prompt **only during training** "a natural portrait of a young woman with smooth youthful facial features". The smile and makeup energies use only their positive prompts.

ImageReward reward. We use the frozen ImageReward-v1.0 text-image preference model (Xu et al., 2023) to define S in Eq. 88 as the preference score, with the run-specific calibration where configured. For scoring, images are clamped to $[-1, 1]$ and bicubic-resized to 224×224. The main-paper ImageReward column reports the raw *positive-prompt* score, not the calibrated reward or elderly-woman margin. For this evaluation, each saved output is clamped, then given a seeded reflect-padded translation of up to four pixels, random resized crop, and horizontal flip before the 224×224 resize.

Normalization. In addition, the ImageReward was normalized w.r.t. our CelebA data to stabilize the learning procedure and make choice of strength parameter λ easier. We pick 1024 train samples from the dataset, score them with unnormalized ImageReward and normalized by mean and std.

PickScore reward. PickScore-v1 (Kirstain et al., 2023) is a CLIP-H model finetuned on Pick-a-Pic preference comparisons. Its image–text similarity could provide S in Eq. 88, but PickScore is not the training reward in these three runs: we use it only for post-hoc evaluation. The reported PickScore column is the raw positive-prompt similarity, with no subtraction of the elderly-woman negative prompt. We apply the same seeded augmentation to each saved output: clamp to $[-1, 1]$, reflect-padded translation of up to four pixels, random resized crop, and horizontal flip, followed by bicubic resizing to 224×224 for scoring.

E EXPERIMENTAL DETAILS

Tables 3 and 4 summarize the hyperparameters of the TSBM and pretrained DSBM models respectively. Here ϵ is the diffusion coefficient in the SDE. Gaussian noise is omitted on the final sampling step for image experiments. Parameter counts are for each trainable network, excluding its frozen pretrained counterpart. The EMA states for Exponential Moving Average. The Adam is used for optimization Kingma & Ba (2015). The SDE was simulated using Euler Maryama and then "lean" adjoint ODE was integrated using the SDE trajectory and Euler integrator. Image data the was normalized to $[-1, 1]$. For DSBM hyperparamter names explanation visit corresponding Appendix in Shi et al. (2023).

For image experiments we've used the official DSBM github repository:

`https://github.com/yuyang-shi/dsbm-pytorch`

Table 3: TSBM hyperparamters.

Hyperparameter	Toy 2D	Colored MNIST	CelebA
Controller gradient steps / stage	1,000	1,000	1,000
Corrector gradient steps / stage	782	1,000	1,000
Training batch size	128	32	32
Controller replay size	None (fresh batch for every step)	512 (red); 1,024 (strokes)	1,024
Controller replay refresh	Every step	Every 50 steps	Every 100 steps
Corrector data-set size / stage	100,000 endpoints	4,096 endpoints	8,192 endpoints
EMA rate	None	0.99	0.99
Optimizer and learning rate	Adam, 10^{-3}	Adam, 10^{-5} (red); $3 \cdot 10^{-5}$ (strokes)	Adam, 10^{-5}
Controller / corrector network	Two-hidden-layer SiLU MLP, width 64	U-Net, 128 base channels, 2 residual blocks / scale	U-Net, 128 base channels, 2 residual blocks / scale
Controller parameters	4,546	≈ 39.6M	≈ 38.3M
Corrector parameters	4,482	≈ 39.6M	≈ 38.3M
Diffusion coefficient ϵ	1	1	1
Reward strength λ	1	1,000	30
Controller / corrector damping	1 / 1	0.9 / 0.9 (red, thick); 1 / 1 (thin)	0.9 / 0.9
NFE	40	30	100
CondSNIS particles K per input	64	8	4
CondSMC particles N per input	64	8	4
TSBM stages	20	5	5
Training wall clock time	15min	1h	13h
Hardware	CPU	2 GPU	2 GPU

Table 4: DSBM pretraining hyperparamters.

Hyperparameter	Toy 2D	Colored MNIST	CelebA
Forward / backward gradient steps, initial IMF iter	10,000	100,000	200,000
Forward / backward gradient steps	10,000	5,000	20,000
Training batch size	128	128	64
Coupling cache size	100,000 pairs	10,000 pairs	1,000 pairs
Coupling cache refresh	Once / direction / cycle	Every 1,000 steps	Every 1,000 steps
EMA rate	None	0.999	0.999
Optimizer and learning rate	Adam, 10^{-4}	Adam, 10^{-4}	Adam, 10^{-4}
Forward / backward network	Two-hidden-layer SiLU MLP, width 128	U-Net, 128 base channels, 2 residual blocks / scale	U-Net, 128 base channels, 2 residual blocks / scale
Forward model parameters	17,282	≈ 39.6M	≈ 38.3M
Backward model parameters	17,282	≈ 39.6M	≈ 38.3M
Diffusion coefficient ϵ	1	1	1
NFE	40	30	100
DSBM stages used	20	20	20

F ADDITIONAL QUALITATIVE RESULTS

F.1 COLORED MNIST

This section gives the complete qualitative Colored MNIST comparisons corresponding to Figure 2. In addition to the methods retained in the main text, the grids show 16 rather than six aligned sources and include DPS with $\gamma = 0.5$ and CondSNIS with $K = 8$ proposals per fixed source.



Figure 5: Full red-chroma qualitative comparison on Colored MNIST.

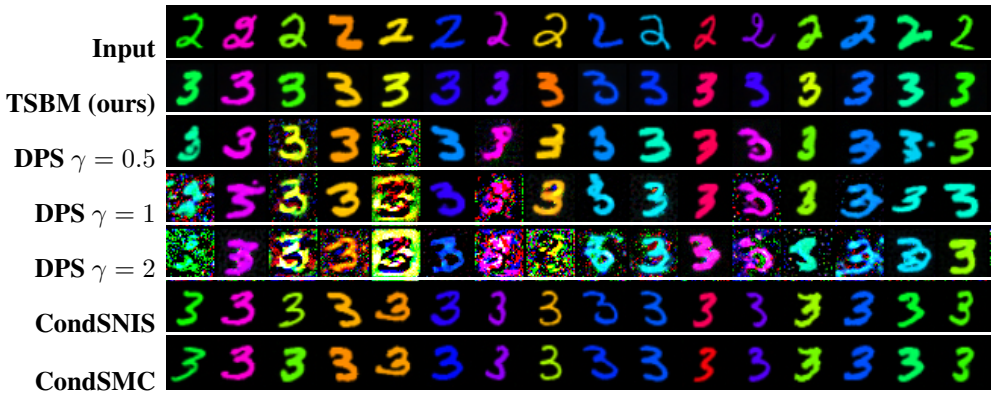


Figure 6: Full Thick-stroke qualitative comparison on Colored MNIST.



Figure 7: Full Thin-stroke qualitative comparison on Colored MNIST.

F.2 CELEBA

This section gives the complete qualitative CelebA comparisons corresponding to Figure 4. In addition to the methods retained in the main text we include DPS with $\gamma = 0.5, 1, 2$ and CondSNIS with $K = 4$ proposals per fixed source.



Figure 8: Full qualitative comparison for "Natural Toothy Smile" reward on CelebA. Gray outputs and visual artifacts are retained from the saved samples.

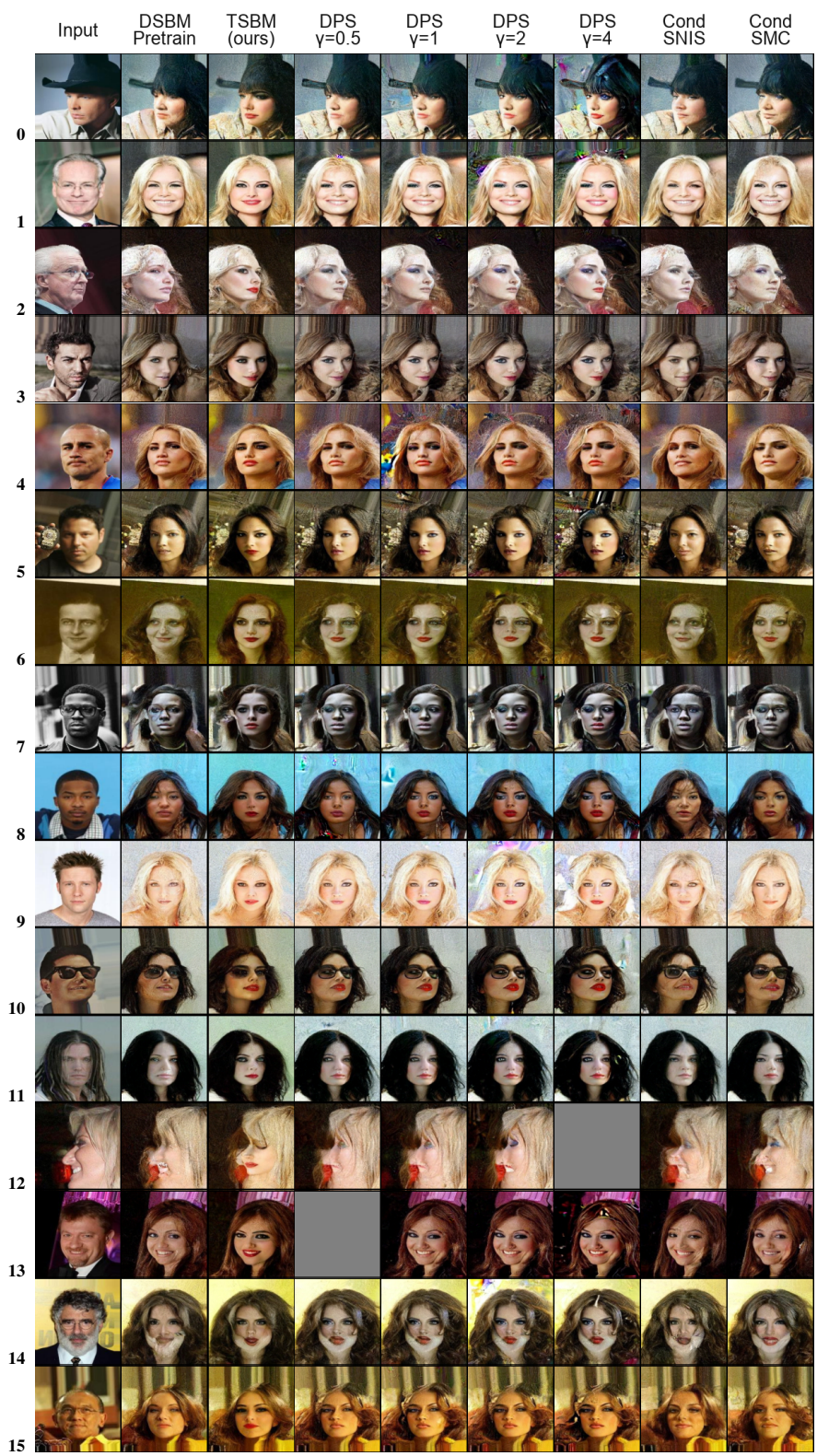


Figure 9: Full qualitative comparison for ”Heavy Makeup” reward on CelebA. Gray outputs and visual artifacts are retained from the saved samples.

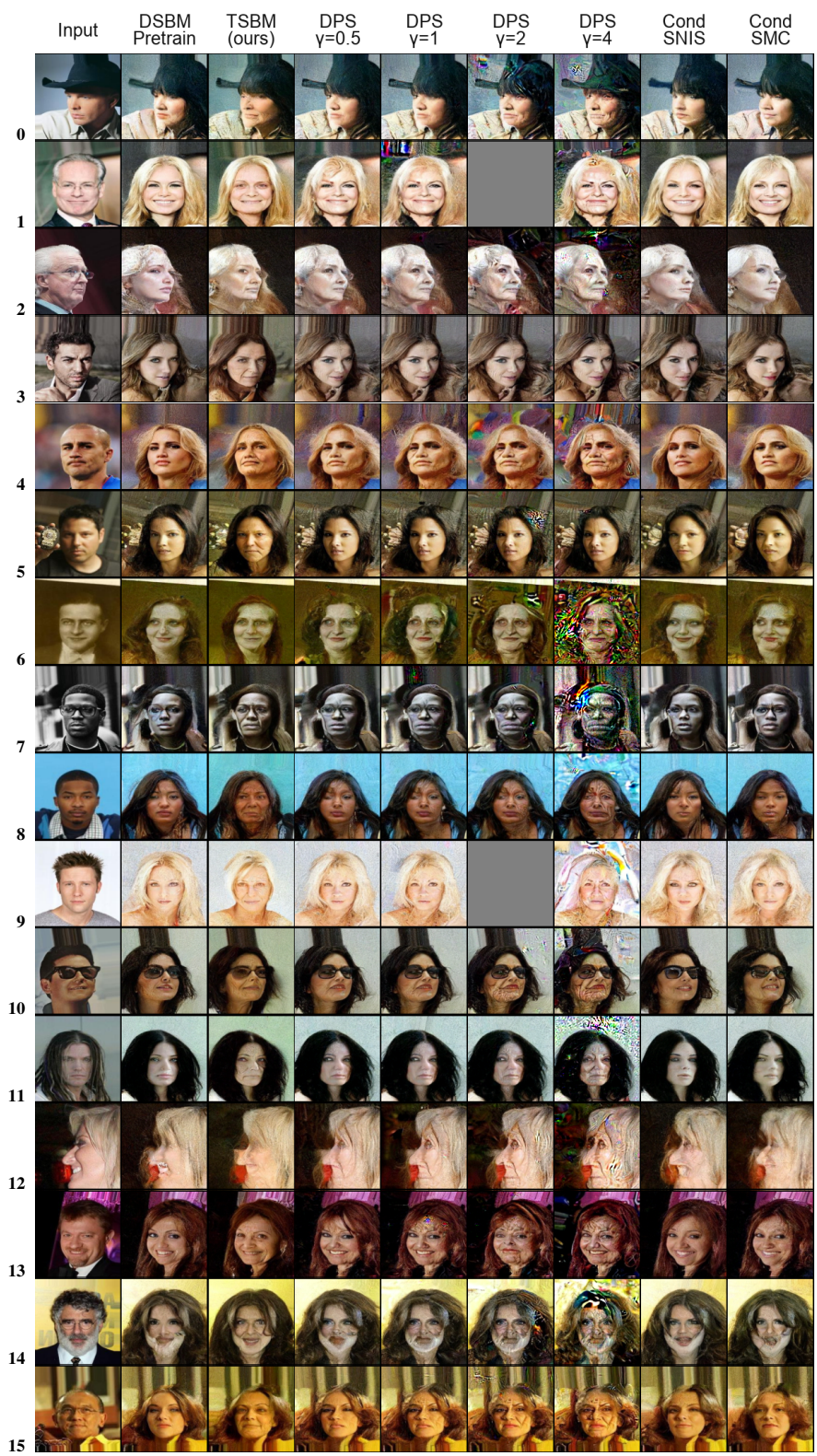
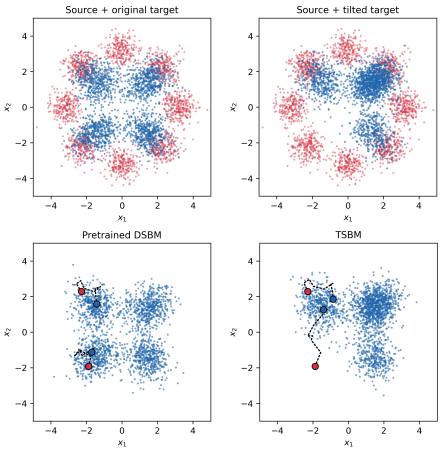


Figure 10: Full qualitative comparison for ”Elderly Woman” reward on CelebA. Gray outputs and visual artifacts are retained from the saved samples.



(a)

Method	TV ↓	sliced W_1 ↓	Cost
Exact Data Sinkhorn	-	-	2.300
DSBM	.354	.788	1.604
DPS $\gamma = 0.5$	.309	.560	1.785
DPS $\gamma = 1$	.281	.493	1.858
DPS $\gamma = 2$	.254	.441	1.896
DPS $\gamma = 4$	.234	.402	1.933
SNIS	.298	.536	1.788
SMC	.287	.511	1.828
TSBM (ours)	**.117**	**.117**	**2.180**

(b)

Figure 11: **Two-dimensional toy experiment.** (a) The first two panels overlay the red source with the blue original and tilted targets; the remaining panels show terminal samples and generation trajectories from DSBM and TSBM.Red and blue dots with black outlines mark trajectory starts and endpoints, respectively; black dotted lines trace the trajectories. (b) Total variation and sliced W_1 are measured against the ground-truth tilted distribution. Cost denotes $\mathbb{E}\|\hat{x}_1 - x_0\|_2$. Bold marks the best distributional metrics and the cost closest to the Sinkhorn reference.

G Additional Experiments

G.1 Toy illustrative experiment

Setup. The source is an equally weighted mixture of eight Gaussians, and the original target is an equally weighted mixture of four Gaussians. We train the pretrain bridge with 20 DSBM stages and $\sigma = 1$ using 10^5 training samples from each marginal. The reward changes the four target mixture weights from $(0.25, 0.25, 0.25, 0.25)$ to $(0, 0.2, 0.2, 0.6)$ and is kep with strength $\lambda = 1$. We finetune the frozen DSBM for 20 TSBM stages. Figure 11b reports all methods on the same 10^4 validation sources, with 40 Euler steps. Other experimental details are described in Appendix E.

Results. Figure 11a shows the original and tilted targets alongside samples from the pretrained DSBM and TSBM. While Figure 11b shows quantitative evaluation w.r.t. tilted distribution: Total Variance, slide Wasserstein-1 distance and average cost of translation from source till target. One can see from quantitative results that TSBM shows the best target distributions fitting metrics, i.e., TV and sliced W_1, while outperfroming significantly both the inference time alignment method and TSBM without corrector. We also apply Sinkhorn to GT tilted validation samples to estimate the *transport cost* of the GT tilted SB. TSBM achieves the closest cost to this reference, supporting recovery of the reference GT bridge's transport behavior. These results show that **TSBM** is the only evaluated method that both **closely matches the tilted target marginal** and **approximates the reference bridge's transport cost**.

G.2 CelebA-128 inference time and working memory

We benchmark inference in the representative CelebA 128×128 male-to-female translation setting with the "Natural Toothy Smile" reward at strength $\lambda = 30$. The benchmark uses the corresponding stage-2 TSBM checkpoint, DPS with $\gamma = 1$, and one NVIDIA A100-SXM4-80GB. Every trajectory uses 100 drift-network evaluations, i.e., NFE. Model and checkpoint loading, dataset construction, post-hoc metrics, serialization, and etc are excluded from the timed region, and CUDA is synchronized around each trial. The single-output protocol runs the algorithm to process one input image, while for CondSNIS and CondSMC this means that they run processing of K parallel samples, i.e., on population. Table 5 reports mean and standard deviation over 16 timed trials after one warm-up. The batch-16 protocol test the *batched processing* and processes 16 input images and measures the time needed to produce one output for each image. CondSNIS and CondSMC use K particles per input,

with at most 16 trajectories evaluated simultaneously. Table 5 reports the median total time over three timed repetitions, divided by 16 to give the time per output. Working memory is the peak increase in live PyTorch GPU tensor memory during a trial, measured relative to the allocation immediately before it starts. It measures the extra memory inference needs, rather than the model's total GPU memory footprint.

Table 5: **Inference cost on CelebA** 128×128**.** Single-output wall-clock time is the mean $\pm$ standard deviation over 16 trials. Batch-16 time is the median total time divided by 16 source outputs. Relative time uses the single-output TSBM mean. An asterisk (*) indicates visual artifacts.

Method	K	Single-output time (s)	Batch-16 s/output	Relative to TSBM	Working memory (GiB)
Pretrained DSBM	1	1.468 ± 0.010	0.355	$1.00\times$	0.180
TSBM (ours)	1	1.460 ± 0.005	0.357	$1.00\times$	0.180
DPS* $\gamma = 1$	1	7.116 ± 0.011	1.836	$4.88\times$	2.229
SNIS	4	2.089 ± 0.004	1.452	$1.43\times$	0.379
SNIS	8	3.094 ± 0.001	2.905	$2.12\times$	0.758
SNIS	16	5.794 ± 0.002	5.811	$3.97\times$	1.512
SMC	4	4.282 ± 0.006	3.629	$2.93\times$	0.383
SMC	8	7.431 ± 0.002	7.256	$5.09\times$	0.767
SMC	16	14.459 ± 0.004	14.512	$9.91\times$	1.527

TSBM has essentially the same inference cost as the pretrained bridge because it uses the same network architecture and requires neither reward evaluation nor differentiation at sampling time. In the single-output protocol, DPS is $4.88\times$ slower and requires 2.229 GiB of working memory, compared with 0.180 GiB for TSBM. Its larger workspace comes from differentiating through the endpoint prediction and reward model at every step. SNIS and SMC costs increase with K because all particles are propagated for each fixed input. Their working memory grows approximately linearly with K. SMC is slower than SNIS at a fixed K because it additionally evaluates incremental rewards and performs sequential weighting and resampling.

All methods are measured in one process that keeps the DSBM, TSBM, and reward models resident. Consequently, working memory is the appropriate method comparison, whereas the absolute process peak is not a standalone deployment-memory requirement. We define:

$$M_{\text{working}} = M_{\text{peak allocated during trial}} - M_{\text{allocated immediately before trial}}.$$

G.3 Colored MNIST Red-Chroma. Static TSBM corrector and TSBM stages ablation.

In this section we provide additional analysis for the Colored MNIST translation with "red chroma" reward. We assert the stage-wise dynamics and convergence of TSBM in Figure 12, which tracks reward and red target hue error, i.e., deviation of translated image HUE from the target red one, across five TSBM stages, which asserts how much the model does follow the reward. Furthermore, we compare the full method (TSBM) with the no-corrector ablation. In that case, TSBM corrector isn't trained and $h^{(k)} = h_{\text{old}}$, while method is trained with the same computational budget in terms of controller optimization. Both metrics are evaluated on the same $1,024$ fixed sources with 30 solver steps. Lower values are better for both reward and red-target hue error. Finally, we present qualitative evaluation of TSBM vs TSBM with static corrector in Figure 13.

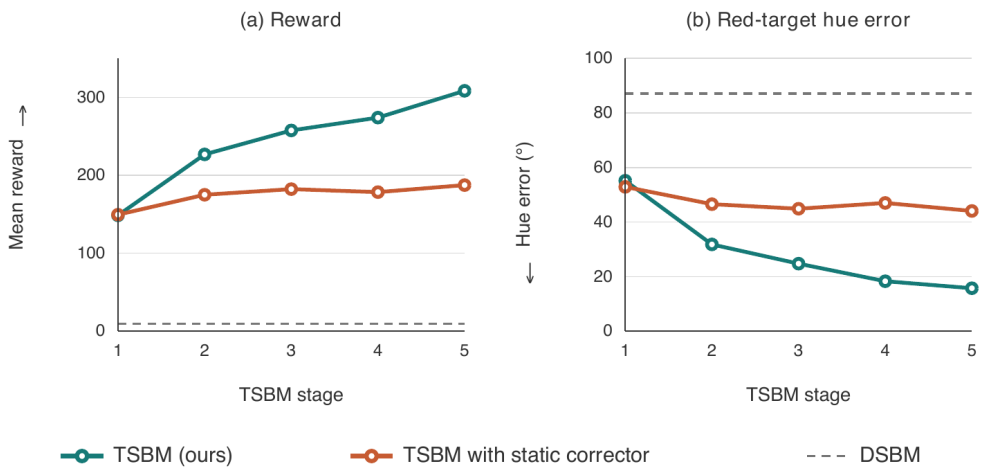


Figure 12: Quantitative comparison of TSBM and TSBM with static corrector on Colored MNIST dataset with "red chroma" reward. The left panel shows mean terminal reward and the right panel the red-target hue error. Solid curves are TSBM and the no-corrector ablation, while dashed gray lines mark the frozen Pretrained DSBM result.

One can see that TSBM improves both the reward and hue error with iteration, while TSBM with static corrector does improve with iterations, but much slower than regular TSBM, see Figure 12. Figure 13 confirms the idea that regular TSBM follows the reward noticeably better than TSBM with static corrector. These results supports the theory and the need for accurate corrector $h^{(k)}$.

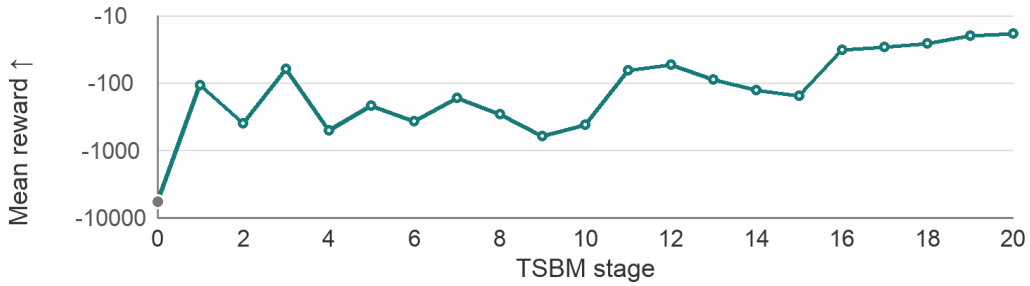


Figure 13: Qualitative comparison of TSBM and TSBM with static corrector samples on Colored MNIST dataset with "red chroma" reward.

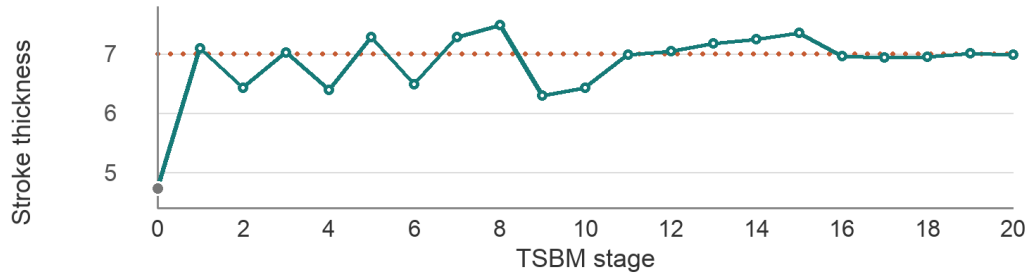
G.4 Colored MNIST Thick-target. TSBM stages ablation.

In this section we provide additional analysis on TSBM dynamics across stages on the Colored MNIST dataset with "thick stroke" reward. We train perform the 20 TSBM stages. In Figure 14 we show the reward, stroke thickness as *reward optimization metrics* and hue error w.r.t. input as *fidelity metric*.

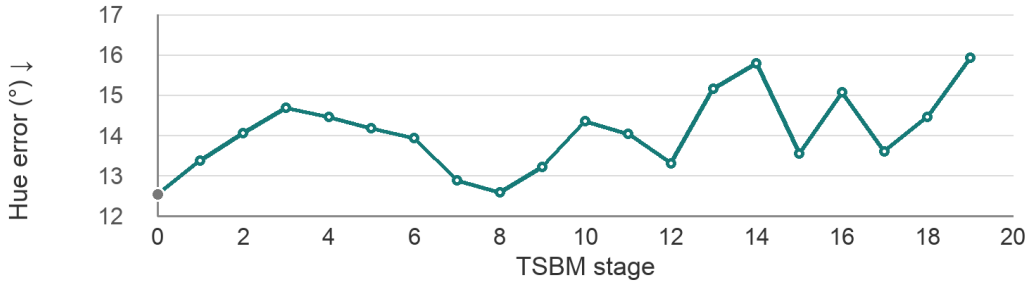
One can see that TSBM improves reward across the TSBM stages, while most of the gains are received in the first several stages. Furthermore, the actual thickness target seems almost achieved in the first TSBM stages, however it indeed improves within TSBM later stages. The hue error seems somewhat stochastic, however it is almost always within 13° and 16° degrees and doesn't diverge much. We assume that its stochasticy is the product of stocastic evaluation and reward side effects.



(a) Mean reward ↑; logarithmic magnitude scale.



(b) Stroke thickness.



(c) Foreground hue error ↓.

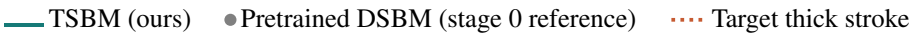


Figure 14: Quantitative analysis of different TSBM stages on Colored MNIST dataset with "thick stroke" reward. The panels show mean reward, stroke thickness, and foreground hue error over $1,024$ fixed test source images. The gray marker is the frozen pretrained DSBM. The dotted line marks the stroke-thickness target.

G.5 CelebA additional quantitative analysis

Here we present additional metrics complementing the study in Table 1 for CelebA dataset with all the rewards. Table 6 shows source-to-output distances, i.e., MSE and LPIPS Zhang et al. (2018), and two additional prompt-alignment scores, i.e., HPSV2.1 Wu et al. (2023b) and CLIPScore Hessel et al. (2021), which recieve data preprocessed the same way as for ImageReward and PickScore, see Appendix D.3. MSE and LPIPS measure how far an output y moves from its aligned input x, i.e., MSE(x, y) and LPIPS(x, y).

One can see that TSBM and DPS $\gamma = 4$ are the two best methods by HPS and CLIPScore. This confirms the strong TSBM generalization properties among several rewards, but shows DPS $\gamma = 4$ as competitor. However, as we noticed before DPS methods, especially with higher γ introduce a significant level of artifacts, see Figures 10, 8, 9, 4. Across all three prompts, TSBM has the largest source-to-output MSE but lower LPIPS than every DPS variant. It therefore changes more pixels while remaining closer to the input under the perceptual feature metric. CondSNIS and CondSMC

have smaller input-to-output distances but also weaker prompt scores, so those distances alone do not indicate better edits.

Table 6: Additional CelebA quantitative results. MSE and LPIPS compare each output with its input, smaller values indicate less change. HPS v2.1 and CLIPScore use the same prompts as in Appendix D.3 with the same translation, crop, and horizontal-flip augmentation, larger values indicate better prompt alignment. **Bold** and underline marks the largest and second largest HPS or CLIPScore mean within each prompt. (*) denotes the DPS settings marked for visual artifacts in Table 1.

Method	MSE input-output	LPIPS input-output	HPS v2.1 ↑	CLIPScore ↑
Natural Toothy Smile				
Pretrained DSBM	0.1677	0.2983	0.1806	19.6923
TSBM (ours)	0.1923	0.3074	**0.2012**	21.5642
DPS $\gamma = 0.5$	0.1815	0.3363	0.1867	20.2365
DPS* $\gamma = 1$	0.1827	0.3409	0.1928	20.8415
DPS* $\gamma = 2$	0.1843	0.3483	0.1982	21.3689
DPS* $\gamma = 4$	0.1901	0.3653	0.1997	**21.6390**
CondSNIS	0.1676	0.2981	0.1830	19.9052
CondSMC	0.1682	0.2975	0.1890	20.4165
Heavy Makeup				
Pretrained DSBM	0.1677	0.2983	0.1540	18.1912
TSBM (ours)	0.1955	0.3159	0.1858	**21.6400**
DPS $\gamma = 0.5$	0.1828	0.3402	0.1687	19.7912
DPS* $\gamma = 1$	0.1820	0.3460	0.1761	20.4257
DPS* $\gamma = 2$	0.1848	0.3532	0.1825	20.8767
DPS* $\gamma = 4$	0.1827	0.3528	**0.1872**	21.1343
CondSNIS	0.1677	0.2983	0.1564	18.4695
CondSMC	0.1686	0.3021	0.1653	19.5327
Elderly Woman				
Pretrained DSBM	0.1677	0.2983	0.1885	18.8912
TSBM (ours)	0.1878	0.3018	**0.2003**	**23.4400**
DPS $\gamma = 0.5$	0.1722	0.3160	0.1877	20.3012
DPS* $\gamma = 1$	0.1725	0.3279	0.1903	21.2336
DPS* $\gamma = 2$	0.1761	0.3452	0.1935	22.0910
DPS* $\gamma = 4$	0.1859	0.3811	0.1953	22.7870
CondSNIS	0.1675	0.2987	0.1884	19.2562
CondSMC	0.1666	0.2998	0.1875	19.8498

G.6 Additional CelebA rewards

We evaluate two further ImageReward preferences on CelebA 128×128 male-to-female translation. Both use the contrastive reward in Eq. 88 with $\lambda = 30$. The positive and negative prompts are, respectively:

- *Aesthetics*: "a sharp, clean, well-exposed photograph of a face" and "a blurry, noisy, distorted, low-quality photograph of a face".
- *Eyeglasses with Thick Frames*: "a natural portrait of a person wearing eyeglasses with thick visible frames" and "a natural portrait of a person without eyeglasses".

As in the main CelebA experiment, ImageReward and PickScore are evaluated on the positive prompt alone, after translation, crop, and horizontal-flip augmentation. CLIP-IQA measures naturalness. Each table reports means over the same *128* fixed inputs within its reward experiment, using 100 bridge steps. CondSNIS and CondSMC use four particles per input. The TSBM was ran for 5 stages.

Table 7: Aesthetics reward on CelebA-128. ImageReward and PickScore use the positive prompt with translation, crop, and horizontal-flip augmentation; CLIP-IQA measures naturalness. Means are over 128 fixed inputs, with TSBM evaluated after stage 5. All metrics are higher-is-better. Bold marks the largest mean. (*) indicates visible artifacts in the saved samples.

Method	ImageReward ↑	PickScore ↑	CLIP-IQA ↑
Pretrained DSBM	-0.2353	18.0223	0.5446
TSBM (ours)	**0.6283**	**18.6727**	**0.5738**
DPS* $\gamma = 0.5$	0.1858	18.1216	0.5453
DPS* $\gamma = 1$	0.2383	17.9490	0.4901
DPS* $\gamma = 2$	0.5524	17.5891	0.3833
DPS* $\gamma = 4$	0.5592	17.2191	0.2649
CondSNIS	-0.1098	18.0474	0.5684
CondSMC	-0.0164	18.2051	0.5632

Table 8: Eyeglasses with thick frames reward on CelebA-128. Metrics and augmentation follow Table 7. Means are over 128 fixed inputs, with TSBM evaluated after stage 5. All metrics are higher-is-better. Bold marks the largest mean. (*) indicates visible artifacts in the saved samples.

Method	ImageReward ↑	PickScore ↑	CLIP-IQA ↑
Pretrained DSBM	-1.8730	17.2151	0.5446
TSBM (ours)	**-0.0468**	**18.1322**	0.4754
DPS* $\gamma = 0.5$	-1.8729	17.1511	0.5538
DPS* $\gamma = 1$	-1.8260	17.0452	0.5499
DPS* $\gamma = 2$	-1.7740	16.9616	0.5365
DPS* $\gamma = 4$	-1.7151	16.8643	0.4952
CondSNIS	-1.8324	17.1915	0.5462
CondSMC	-1.8651	17.1799	**0.5547**

TSBM has the largest ImageReward and PickScore means for both rewards. For aesthetics it also has the largest CLIP-IQA naturalness mean. For eyeglasses, CLIP-IQA falls from 0.5446 for the pretrained DSBM to 0.4754 for TSBM, while CondSMC has the largest naturalness mean at 0.5547. The full sample comparisons in Figures 15 and 16 show the corresponding edits and the artifacts produced by DPS. The task of adding eyeglasses to a person appears to be particularly challenging: the baselines achieve poor reward alignment under both the training ImageReward and evaluation PickScore metrics. While TSBM substantially improves over the baselines, it does not fully solve the problem, as shown in Figure 16.



Figure 15: Full qualitative comparison for the aesthetics reward on CelebA. Rows show the first 16 fixed sources; columns show the input, pretrained DSBM, TSBM (ours) at stage 10, DPS with $\gamma \in \{0.5, 1, 2, 4\}$, CondSNIS, and CondSMC. Gray outputs and visual artifacts are retained from the saved samples.



Figure 16: Full qualitative comparison for the eyeglasses with thick frames reward on CelebA. Rows show the first 16 fixed sources; columns show the input, pretrained DSBM, TSBM (ours) at stage 5, DPS with $\gamma \in \{0.5, 1, 2, 4\}$, CondSNIS, and CondSMC. Gray outputs and visual artifacts are retained from the saved samples.