DISTRIBUTIONALLY ROBUST SCHRÖDINGER BRIDGE

Anonymous authors
Paper under double-blind review

ABSTRACT

Schrödinger bridge (SB) learns stochastic transport between prescribed initial and target distributions. When the initial distribution shifts at test time, the learned dynamics can fail to recover the target distribution. We introduce the Distributionally Robust Schrödinger Bridge (DRSB), which learns a single controller that accounts for uncertainty in the initial distribution. The DRSB objective consists of control energy and a KL penalty between the resulting terminal distribution and the target distribution. DRSB seeks a single controller that minimizes the worst-case value of this objective as the initial distribution varies within an ambiguity set around the nominal distribution. We derive an exact variational formulation of this objective and connect its fixed-terminal-cost subproblem to stochastic optimal control and distributionally robust optimization. This formulation motivates an alternating algorithm that updates the adversarial initial distribution, estimates the terminal log-density ratio, and trains the controller. We develop Wasserstein and Sinkhorn variants using stochastic control optimality conditions to approximate the gradients required for adversarial updates. Experiments on two-dimensional transport tasks and image-to-image translation show improved robustness to input perturbations relative to standard SB, with a tradeoff in nominal performance. On Gaussian mixture transport, Sinkhorn DRSB also achieves lower mean sliced Wasserstein distance than fixed-level noise augmentation at both tested unseen noise levels.

1 INTRODUCTION

Generative modeling is fundamentally a task of learning a transport map from a source distribution to a target distribution. While early approaches, such as GANs (Goodfellow et al., 2014), focused on learning static maps, recent approaches such as diffusion models (Ho et al., 2020; Song et al., 2021) learn dynamic transport maps via stochastic differential equations (SDEs). The Schrödinger Bridge (SB) problem generalizes standard diffusion models and likewise seeks an optimal transport mapping described by an SDE.

By definition, SB seeks a stochastic process that connects two arbitrary distributions and minimizes the KL divergence with respect to a reference stochastic process (Schrödinger, 1931; Léonard, 2014). SB has been used in applications such as sampling (Liu et al., 2025), image-to-image translation (Theodoropoulos et al., 2024), and image restoration (Liu et al., 2023). However, the SB dynamics are trained for a prescribed initial distribution. When this distribution shifts at test time, the same dynamics produce a different terminal distribution, violating the original target marginal constraint. This motivates a distributionally robust SB formulation that accounts for uncertainty in the initial distribution.

In this paper, we formulate the **Distributionally Robust Schrödinger Bridge (DRSB)** problem to achieve robust generation under initial distribution shifts. The goal of DRSB is to find a single controller that transports every initial distribution in a prescribed set close to the target distribution. This set, called the ambiguity set, consists of distributions within a prescribed distance of the nominal initial distribution. We connect DRSB to stochastic optimal control through a terminal cost that penalizes mismatch between the induced terminal distribution and the target distribution. This cost takes the form of the log-density ratio between these distributions, which we approximate with a neural network to provide a training signal for the controller. Since the induced terminal distribution depends on both the controller and the initial distribution, the terminal cost must be re-estimated as these change.

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.

For a fixed terminal cost, the resulting subproblem has the standard distributionally robust optimization (DRO) form (Mohajerin Esfahani & Kuhn, 2018; Gao & Kleywegt, 2023), where the control variable serves as the DRO decision variable. Strong duality results from the DRO literature provide a dual formulation of the inner worst-case maximization over the ambiguity set. We update the adversarial initial distribution using this dual formulation and train the controller via adjoint matching (Domingo-Enrich et al., 2024) to minimize the expected transport cost under the current adversarial distribution. We alternate these updates with re-estimation of the terminal cost, training its network as a binary classifier that distinguishes the resulting terminal samples from target samples.

We develop Wasserstein and Sinkhorn variants of DRSB, with adversarial input updates derived from their DRO dual formulations. Using an optimality condition from stochastic optimal control theory (Kappen, 2005), we approximate cost gradients with controller evaluations. The Wasserstein variant iteratively updates adversarial inputs. For the Sinkhorn variant, a first-order approximation of the cost yields a Gaussian sampling rule for each nominal input and a closed-form update of the dual variable. This approximation allows adversarial inputs to be sampled without iterative inner optimization.

We evaluate DRSB on 2D Gaussian-to-Gaussian and GMM-to-GMM transport tasks and on image-to-image translation. The results show that DRSB improves robustness to noisy and shifted initial distributions compared with standard SB. On the 2D transport tasks, Sinkhorn DRSB outperforms Wasserstein DRSB under most tested perturbation conditions.

The main contributions of this paper are summarized as follows.

- We formulate Distributionally Robust Schrödinger Bridge problems to address initial distribution shift.
- We derive an exact variational formulation of DRSB and connect its fixed-terminal-cost subproblem to SOC and DRO.
- We develop block update algorithms for the Wasserstein and Sinkhorn variants using SOC optimality conditions.

2 PRELIMINARIES

2.1 SCHRÖDINGER BRIDGE UNDER INITIAL DISTRIBUTION UNCERTAINTY

The standard Schrödinger bridge (SB) problem (Schrödinger, 1931; Léonard, 2014) is

Problem 2.1 (Schrödinger Bridge)**.**

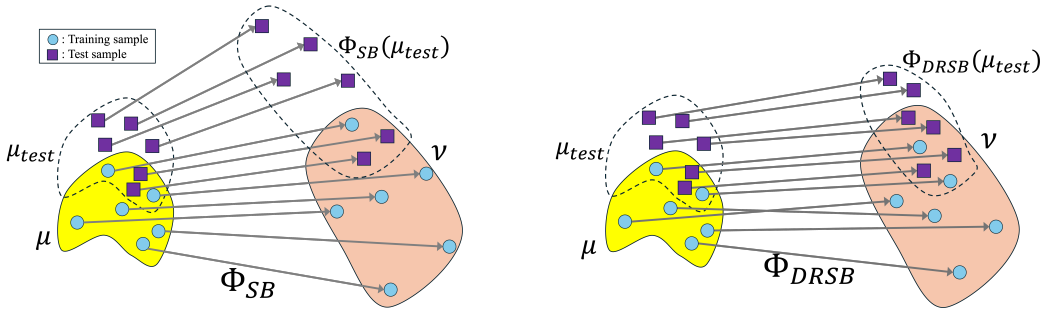
$$\min_{u_t} \sigma^2 \mathrm{KL}(p^{u,\mu} \| p^{\text{base}}) = \min_{u_t} \mathbb{E}_{X\sim p^{u,\mu}} \left[\int_0^1 \frac{1}{2} \|u_t^\theta(X_t)\|^2 dt \right], \tag{1a}$$

$$\text{s.t.} \quad dX_t = u_t^\theta(X_t)dt + \sigma dW_t, \quad X_0 \sim \mu(X_0), \quad X_1 \sim \nu(X_1). \tag{1b}$$

Here, $p^{u,\mu}$ denotes the path measure induced by Equation (1b) with the initial law μ, and $p^{\text{base}} := p^{u:=0}$ denotes the base path measure when $u_t := 0$. The base path measure has the same initial law μ. The equality assumes the conditions for Girsanov's theorem. $X_t \in \mathbb{R}^d$ is the d-dimensional state vector at time t. The control $u^\theta : [0,1] \times \mathbb{R}^d \to \mathbb{R}^d$ is parameterized by θ. W_t is a standard d-dimensional Brownian motion, and σ is a constant positive diffusion coefficient. $p_0^{u,\mu}$ and $p_1^{u,\mu}$ are the marginal distributions of $p^{u,\mu}$ at times $t = 0$ and $t = 1$, respectively. Then, by definition, $p_0^{u,\mu} = \mu$ always holds. The Schrödinger bridge (SB) problem seeks the optimal control by minimizing the KL divergence between $p^{u,\mu}$ and p^{base} while transporting the initial data distribution μ to the target data distribution ν.

Deployment with an unknown initial distribution. Let $u^\star$ solve Problem 2.1 for the nominal initial distribution μ. For this fixed controller, the Fokker–Planck equation defines the map Φ_{SB} from initial to terminal distributions,

$$\Phi_{\text{SB}} : p_0^{u^\star,\mu} \mapsto p_1^{u^\star,\mu}, \quad \Phi_{\text{SB}}(\mu) = \nu. \tag{2}$$



(a) Standard SB (b) DRSB (ours)

Figure 1: Deployment under initial distribution uncertainty. Blue circles denote nominal samples from μ, and purple squares denote shifted test samples from μ_{test}. The desired target ν is unchanged. (a) A standard SB controller matches ν for μ, but this matching need not persist under an initial distribution shift. (b) DRSB trains one controller to reduce the worst-case sum of control energy and terminal mismatch over an ambiguity set of initial distributions.

At deployment, inputs may instead follow an unknown distribution $\mu_{\text{test}} \neq \mu$. Applying the same controller need not preserve terminal matching, *i.e.*

$$\Phi_{\text{SB}}(\mu_{\text{test}}) \neq \nu \quad \text{in general.} \tag{3}$$

We consider a deployment scenario where the test-time initial law is not known during training and a single learned controller must be used without test-time retraining.

Figure 1 illustrates this deployment test setting using point clouds. In Figure 1a, blue circles represent samples from the nominal initial distribution μ, and purple squares represent samples from a perturbed distribution μ_{test}. The learned dynamics are designed to transport the nominal initial distribution to the target distribution ν. If the initial distribution changes, applying the same dynamics may produce a terminal distribution that no longer matches ν, even though the desired target remains the same. DRSB learns a single controller to steer plausible initial distributions within an ambiguity set toward the same target ν (Figure 1b) by minimizing the worst-case sum of control energy and the KL divergence between the resulting terminal distribution and ν.

2.2 Distributionally Robust Optimization

In stochastic optimization, the underlying data distribution is often unknown, and only a few data samples are available. DRO hedges against this uncertainty by optimizing under the worst-case distribution within a prescribed set of distributions called the ambiguity set (Wiesemann et al., 2014). The DRO problem takes the minimax form

$$\min_{u \in \mathcal{U}} \max_{\mu \in \mathfrak{M}_\epsilon(\hat{\mu})} \mathbb{E}_{X_0 \sim \mu}[V(u, X_0)], \tag{4}$$

where the outer minimization selects the decision variable u, and the inner maximization identifies the worst-case distribution μ within the ambiguity set $\mathfrak{M}_\epsilon(\hat{\mu}) := \{\mu : D(\mu, \hat{\mu}) \leq \epsilon\}$, defined as the set of distributions within distance ϵ of the nominal distribution $\hat{\mu}$ under distributional distance D.

3 Distributionally Robust Schrödinger Bridge

3.1 Distributional Robustness in SB and Connection to SOC

We formulate and solve the Distributionally Robust Schrödinger Bridge (DRSB) problem, which seeks a control that can robustly steer shifted test-time distributions μ_{test} toward the target distribution. All proofs are provided in Appendix A.

The DRSB formulation is

Problem 3.1 (Distributionally Robust Schrödinger Bridge)**.**

$$\min_{u} \max_{\mu \in \mathfrak{M}_\epsilon(\hat{\mu})} \mathbb{E}_{X \sim p^{u,\mu}} \left[\int_0^1 \frac{1}{2} \|u_t^\theta(X_t)\|^2 dt \right] + \rho \mathrm{KL}(p_1^{u,\mu} \| \nu), \tag{5a}$$

$$\text{s.t.} \quad dX_t = u_t^\theta(X_t)dt + \sigma dW_t, \quad X_0 \sim \mu, \tag{5b}$$

$$\mathfrak{M}_\epsilon(\hat{\mu}) = \{\mu : D(\mu, \hat{\mu}) \leq \epsilon\}. \tag{5c}$$

Here, the initial distribution μ is no longer assumed to be known exactly. Instead, the inner maximization in Equation (5a) finds the worst-case initial distribution from the ambiguity set $\mathfrak{M}_\epsilon(\hat{\mu})$. $\hat{\mu}$ is the empirical distribution $\hat{\mu} = \frac{1}{N} \sum_{i=1}^N \delta_{\hat{X}_0^i}$, where $\delta_{\hat{X}_0^i}$ is a Dirac delta function at each sample point $\hat{X}_0^i$. The ambiguity set is centered at $\hat{\mu}$ with radius ϵ under the distributional distance D.

DRSB addresses the choice of a single transport controller before the test-time input distribution is known. We consider input perturbations for which the desired target distribution ν remains unchanged. The control-energy term in Equation (5a) retains the transport-cost criterion of SB, while the soft penalty $\rho\,\mathrm{KL}(p_1^{u,\mu}\|\nu)$ relaxes the hard terminal constraint $p_1^{u,\mu} = \nu$ in Equation (1b). Optimality in Problem 3.1 is defined by minimizing the worst-case sum of control energy and the weighted terminal KL divergence over the ambiguity set. The resulting controller need not preserve the nominal SB transport or be optimal for each initial distribution separately. Nominal performance may therefore be traded for lower worst-case cost. This formulation provides a criterion for selecting one controller when adaptation to each deployment distribution is not possible.

Remark 3.2. For a nonnegative distance with $D(\mu, \hat{\mu}) = 0$ if and only if $\mu = \hat{\mu}$, when $\epsilon = 0$, the ambiguity set reduces to $\mathfrak{M}_0(\hat{\mu}) = \{\hat{\mu}\}$, eliminating the inner maximization. When additionally $\rho \to \infty$, the KL penalty enforces $p_1^{u,\mu} = \nu$ and we recover the standard SB in Problem 2.1.

To optimize the DRSB objective through stochastic optimal control, we require a terminal cost that represents the mismatch between $p_1^{u,\mu}$ and ν. We introduce a terminal cost neural network $g^\phi : \mathbb{R}^d \to \mathbb{R}$ parameterized by ϕ that approximates the log-density ratio between $p_1^{u,\mu}$ and ν.

Proposition 3.3 (Terminal cost approximation)**.** *For a fixed admissible pair (u, μ), suppose $p_1^{u,\mu}$ and ν have positive densities and finite KL divergence. Suppose the function class contains their log-density ratio and a terminal cost function $g^\phi : \mathbb{R}^d \to \mathbb{R}$ is an optimal solution of the optimization problem*

$$g^\phi = \arg\min_g \mathbb{E}_{X_1 \sim p_1^{u,\mu}} \left[\log\left(1 + e^{-g(X_1)}\right) \right] + \mathbb{E}_{X_1 \sim \nu} \left[\log\left(1 + e^{g(X_1)}\right) \right]. \tag{6}$$

Then, $g^\phi(x) = \log\left(\frac{p_1^{u,\mu}(x)}{\nu(x)}\right)$ *and*

$$\mathrm{KL}(p_1^{u,\mu}\|\nu) = \mathbb{E}_{X_1 \sim p_1^{u,\mu}}[g^\phi(X_1)] - \log \mathbb{E}_{X_1 \sim \nu}[e^{g^\phi(X_1)}]. \tag{7}$$

Proposition (3.3) shows that the KL penalty in Equation (5a) can be expressed in terms of the terminal cost g^ϕ using Equation (7). The terminal cost is obtained by solving (6). The discriminator g^ϕ distinguishes generated samples ($X_1 \sim p_1^{u,\mu}$) from target samples ($X_1 \sim \nu$). As a terminal cost, g^ϕ guides the terminal distribution $p_1^{u,\mu}$ toward the target distribution ν through this representation of the KL penalty.

Theorem 3.4 (Variational formulation of DRSB)**.** *For each admissible pair (u, μ), apply the variational representation of the terminal KL divergence. Then Problem 3.1 is equivalent to*

$$\min_u \max_{\mu \in \mathfrak{M}_\epsilon(\hat{\mu})} \sup_g \left\{ \mathbb{E}_{X_0 \sim \mu}[V(u, X_0; g)] - \rho \log \mathbb{E}_{X_1 \sim \nu}[e^{g(X_1)}] \right\}, \tag{8}$$

subject to Equations (5b) *and* (5c)*, where*

$$V(u, x; g) := \mathbb{E}_{X_{\cdot|0} \sim p^u_{\cdot|0}} \left[\int_0^1 \frac{1}{2} \|u_t^\theta(X_t)\|^2 dt + \rho g(X_1) \,\middle|\, X_0 = x \right]. \tag{9}$$

Here, $p^u_{\cdot|0}$ is the controlled path distribution conditioned on the initial state, and the supremum is over measurable g for which the expectations are well defined and $0 < \mathbb{E}_\nu[e^g] < \infty$. The optimizing terminal cost depends on (u, μ) and cannot generally be fixed across the outer optimization.

For fixed g, the normalization term is constant in (u, μ), giving the subproblem

$$\min_{u} \max_{\mu \in \mathfrak{M}_\epsilon(\hat{\mu})} \mathbb{E}_{X_0 \sim \mu}[V(u, X_0; g)], \qquad \text{s.t. Equations (5b) and (5c).} \tag{10}$$

We write $V(u, x) = V(u, x; g)$ when g is held fixed.

From the SOC perspective, $V(u, x; g)$ in Equation (9) is the cost of a stochastic control problem with terminal cost ρg. For fixed (μ, g), minimizing it gives a control adapted to that initial distribution and terminal cost (Chen et al., 2021; Liu et al., 2022). A shift in the initial distribution can change the terminal marginal. DRSB instead minimizes the worst-case objective over the ambiguity set, retaining the optimization over g in Equation (8).

For fixed g, introducing a Lagrange multiplier $\lambda \geq 0$ gives the dual of Equation (10), with equality when strong duality holds:

$$\min_{u} \min_{\lambda \geq 0} \lambda\epsilon + \max_{\mu} \left\{ \mathbb{E}_{X_0 \sim \mu} \left[V(u, X_0) \right] - \lambda D(\mu, \hat{\mu}) \right\}, \quad \text{s.t.} \quad \text{(5b)}. \tag{11}$$

3.2 Iterative Optimization Scheme for DRSB

We use block updates for Equation (8) in the order $\mu^{(k)} \to g^{(k)} \to u^{(k)}$, holding the other two variables fixed at each step. First, with $(u^{(k-1)}, g^{(k-1)})$ fixed, the adversarial initial distribution is updated through the DRO dual:

$$\mu^{(k)}, \lambda^{(k)} \leftarrow \min_{\lambda \geq 0} \ \lambda\epsilon + \max_{\mu} \left\{ \mathbb{E}_{X_0 \sim \mu}[V(u^{(k-1)}, X_0; g^{(k-1)})] - \lambda D(\mu, \hat{\mu}) \right\}. \tag{12}$$

We simulate the dynamics with $u^{(k-1)}$ from $\mu^{(k)}$ and denote the resulting terminal marginal by $p_1^{(k)} = p_1^{u^{(k-1)}, \mu^{(k)}}$. The terminal cost update is

$$g^{(k)} = \arg\min_{g} \mathbb{E}_{X_1 \sim p_1^{(k)}} \left[\log \left(1 + e^{-g(X_1)} \right) \right] + \mathbb{E}_{X_1 \sim \nu} \left[\log \left(1 + e^{g(X_1)} \right) \right]. \tag{13}$$

At the population optimum, this loss recovers the same log-density ratio as the KL variational objective. With $(\mu^{(k)}, g^{(k)})$ fixed, the control subproblem is

$$u^{(k)} \leftarrow \arg\min_{u} \mathbb{E}_{X_0 \sim \mu^{(k)}}[V(u, X_0; g^{(k)})], \qquad \text{s.t.} \quad \text{(5b)}. \tag{14}$$

We use the adjoint matching objective (Domingo-Enrich et al., 2024):

$$u^{(k)} = \arg\min_{u} \mathbb{E}_{X_0 \sim \mu^{(k)}} \mathbb{E}_{t \sim \mathcal{U}[0,1]} \mathbb{E}_{X_t, X_1 \sim p^{\bar{u}}_{\cdot|0}} \left[\| u^\theta(t, X_t) + \rho \nabla g^{(k)}(X_1) \|^2 \right], \tag{15}$$

where $p^{\bar{u}}_{\cdot|0}$ is the path measure induced by $\bar{u} = \texttt{stopgrad}(u^\theta)$, conditioned on the initial state. This objective regresses the control toward the negative terminal cost gradient. In practice, each block uses finitely many optimization steps. These updates neither exchange the minimax operators nor establish convergence of the full nested problem.

The DRSB training procedure is summarized in Algorithm 1. Following the successive approximation method from robust control theory (Beard & McLain, 1998), we initialize u^θ with a pretrained standard SB model. Steps 3–5 remain the same across all DRSB variants. Only Step 2, the adversarial initial distribution update, depends on the choice of ambiguity set distance D. The cost of Step 2 depends on this choice, while the total training cost also includes simulation and network updates.

Algorithm 1 Distributionally Robust SB

Require: Drift u^θ, terminal cost g^ϕ, nominal distribution $\hat{\mu}$, target ν.
Initialize: u^θ with a pretrained SB model, g^ϕ, λ

1: **for** $k = 1, 2, \ldots, K$ **do**
2: $\quad \mu^{(k)}, \lambda^{(k)} \leftarrow$ Update via (12)
3: $\quad$ Simulate (5b) from $\mu^{(k)}, u^{(k-1)}$
4: $\quad$ Update $g^{(k)}$ via (13)
5: $\quad$ Update $u^{(k)}$ via (15)
6: **end for**

3.3 Variants of DRSB

Different choices of the distributional distance D can lead to different DRSB variants. We consider two variants based on the Wasserstein and Sinkhorn distances. We use a fixed-terminal-cost SOC optimality condition to simplify the adversarial updates. See Appendix B for details.

DRSB with Wasserstein Distance. The following proposition shows the dual formulation of DRSB with Wasserstein distance.

Proposition 3.5 (Dual form of Wasserstein DRSB)**.** *The fixed-g subproblem (10) with the Wasserstein-2 ambiguity set $\mathfrak{M}_\epsilon^W(\hat{\mu}) = \{\mu : W_2^2(\mu, \hat{\mu}) \le \epsilon\}$ admits strong duality (Zhang et al., 2025; Gao & Kleywegt, 2023), and the corresponding dual problem (11) specializes to*

$$\min_u \min_{\lambda \ge 0} \left\{ \lambda\varepsilon - \frac{1}{N}\sum_{i=1}^N \min_{X_0} \left\{ \lambda \|X_0 - \hat{X}_0^i\|^2 - V(u, X_0) \right\} \right\}, \quad \textit{s.t.} \quad \textit{(5b).} \tag{16}$$

Since the nominal distribution is a discrete empirical distribution, the inner minimization must be solved for every sample. Our implementation represents the adversary by one perturbed sample per nominal sample (Gao & Kleywegt, 2023).

Equation (16) is a nested optimization problem over λ and the per-sample initial state X_0. In practice, we use the value of λ from the previous iteration when optimizing the worst-case samples, just as we use the controller from the previous iteration in Equation (12). We optimize X_0 using gradient descent. For a smooth value function and an optimal controller u^* of the fixed-g SOC subproblem, the optimality condition is (Kappen, 2005):

$$u_0^*(x) = -\nabla V(u^*, x). \tag{17}$$

Replacing $-\nabla V(u, x)$ by $u_0(x)$ uses one network evaluation. This is exact at the fixed-g SOC optimum and is a surrogate for a controller after finite optimization. For outer iteration k, hold $u^{(k-1)}$ fixed and let j index the inner updates:

$$X_0^{i,(j+1)} \leftarrow X_0^{i,(j)} - \eta \left[u_0^{(k-1)}(X_0^{i,(j)}) + 2\lambda^{(j)}(X_0^{i,(j)} - \hat{X}_0^i) \right], \quad \eta : \text{step size.} \tag{18}$$

The dual update is

$$\lambda^{(j+1)} \leftarrow \lambda^{(j)} - \zeta \left(\varepsilon - \frac{1}{N}\sum_{i=1}^N \|X_0^{i,(j+1)} - \hat{X}_0^i\|^2 \right), \quad \zeta : \text{step size,} \tag{19}$$

which is valid as a dual update only while $\lambda^{(j+1)} \ge 0$.

DRSB with Sinkhorn Distance. The Sinkhorn distance (Cuturi, 2013) between μ and $\hat{\mu}$ is

$$\mathcal{W}_\kappa(\hat{\mu}, \mu) := \min_{\gamma \in \Gamma(\hat{\mu}, \mu)} \left\{ \mathbb{E}_{(\hat{X}_0, X_0)\sim\gamma} \left[\|\hat{X}_0 - X_0\|^2 + \kappa \log \left(\frac{d\gamma(\hat{X}_0, X_0)}{d\hat{\mu}(\hat{X}_0) dX_0} \right) \right] \right\}, \tag{20}$$

where $\Gamma(\hat{\mu}, \mu)$ is the set of joint distributions between $\hat{\mu}$ and μ, and $\kappa > 0$ is the entropy regularization parameter.

Proposition 3.6 (Dual form of Sinkhorn DRSB)**.** *The fixed-g subproblem (10) with the Sinkhorn ambiguity set $\mathfrak{M}_\epsilon^S(\hat{\mu}) = \{\mu : \mathcal{W}_\kappa(\hat{\mu}, \mu) \le \epsilon\}$ admits strong duality (Wang et al., 2025), and the corresponding dual problem (11) specializes to*

$$\min_u \inf_{\lambda > 0} \lambda\varepsilon + \frac{1}{N}\sum_{i=1}^N \lambda\kappa \log \int e^{\left(V(u,X_0) - \lambda\|X_0 - \hat{X}_0^i\|^2\right)/\lambda\kappa} dX_0, \quad \textit{s.t.} \quad \textit{(5b),} \tag{21}$$

The $\lambda = 0$ boundary is interpreted through a limit. For $\lambda > 0$ and a finite normalizing integral, the worst-case conditional distribution $\gamma^*(X_0|\hat{X}_0^i)$ is obtained as

$$\gamma^*(X_0|\hat{X}_0^i) \propto e^{\left(V(u,X_0) - \lambda\|X_0 - \hat{X}_0^i\|^2\right)/\lambda\kappa}. \tag{22}$$

In contrast to the Wasserstein case, the worst-case distribution in Sinkhorn DRSB is supported on all of $\mathbb{R}^d$, even when the nominal distribution $\hat{\mu}$ is discrete. We approximate this distribution by linearizing the fixed-g cost.

Corollary 3.7 (Gaussian worst-case distribution)**.** *Fix (u, g) and assume the SOC optimality condition in Equation* (17) *holds. Suppose $\varepsilon + \frac{\kappa d}{2}\log(\pi\kappa) > 0$. Under the first-order approximation $V(X_0) \approx V(\hat{X}_0^i) + \nabla V(\hat{X}_0^i)^\top (X_0 - \hat{X}_0^i)$, where $V(x) = V(u, x; g)$, the linearized inner dual in Equation* (21) *is*

$$\inf_{\lambda>0} \lambda\left(\varepsilon + \frac{\kappa d}{2}\log \pi\kappa\right) + \frac{1}{N}\sum_{i=1}^{N}\left[V(\hat{X}_0^i) + \frac{\|u_0^\theta(\hat{X}_0^i)\|^2}{4\lambda}\right], \tag{23}$$

where the worst-case conditional distribution is

$$\gamma^*(X_0|\hat{X}_0^i) = \mathcal{N}\left(\hat{X}_0^i - \frac{u_0^\theta(\hat{X}_0^i)}{2\lambda}, \frac{\kappa}{2}I\right), \tag{24}$$

and the optimal dual variable admits the closed form:

$$\lambda^* = \sqrt{\left(\varepsilon + \frac{\kappa d}{2}\log \pi\kappa\right)^{-1} \frac{1}{N}\sum_{i=1}^{N}\left[\frac{\|u_0^\theta(\hat{X}_0^i)\|^2}{4}\right]}. \tag{25}$$

Corollary 3.7 gives closed-form sampling for the linearized inner problem.

4 Experiments

4.1 Robustness against noisy and shifted distributions on 2D transport tasks

We demonstrate the robustness of our DRSB algorithm on the 2D Gaussian-to-Gaussian and GMM-to-GMM transport tasks. We use the sliced Wasserstein distance (SWD) between the terminal and target distributions as our metric. We use a standard SB model trained with the Diffusion Schrödinger Bridge Matching (DSBM) algorithm (Shi et al., 2023) as the baseline. We test each trained model against three perturbation types, Gaussian noise (Noisy), directional shift (Shifted), and the DRSB worst-case distribution (Worst), to the initial distribution at test time.

The results are summarized in Table 1. On the original distribution, DSBM achieves lower SWD in both tasks. This is because DRSB optimizes for worst-case performance over the ambiguity set, which allocates transport capacity across a wider region of the input space at the cost of nominal accuracy. Under all three perturbation types, DRSB outperforms DSBM on both Gaussian and GMM tasks. DSBM degrades under perturbations because the learned transport map is fitted to the training distribution, and inputs that deviate from the nominal support produce suboptimal trajectories. The gap between DRSB and DSBM is largest on the worst-case distribution trained on DRSB.

Table 1: SWD (↓) under input perturbations (mean ± std, 10 seeds). Org denotes the original input; the other conditions are defined in the text.

	Gaussian-to-Gaussian		*GMM-to-GMM*	
Input	DSBM	DRSB	DSBM	DRSB
Org	**0.038**±0.007	0.123±0.012	**0.902**±0.242	1.165±0.138
Noisy	0.684±0.013	**0.638**±0.016	2.407±0.243	**2.212**±0.413
Shifted	0.666±0.011	**0.501**±0.014	5.917±0.127	**5.840**±0.095
Worst	0.951±0.013	**0.710**±0.013	1.988±0.100	**1.478**±0.100

Table 2: SWD (↓), mean ± std over 10 seeds. W-wst and S-wst are the worst-case inputs of DRSB-W and DRSB-S.

	Gaussian-to-Gaussian		*GMM-to-GMM*	
Input	DRSB-W	DRSB-S	DRSB-W	DRSB-S
Org	0.144±0.007	**0.123**±0.012	**0.885**±0.226	1.165±0.138
Noisy	0.678±0.013	**0.638**±0.016	2.315±0.236	**2.212**±0.413
Shifted	0.517±0.014	**0.501**±0.014	5.967±0.153	**5.840**±0.095
W-wst	0.972±0.007	**0.881**±0.018	**1.503**±0.052	1.791±0.102
S-wst	0.812±0.012	**0.710**±0.013	1.946±0.100	**1.478**±0.100

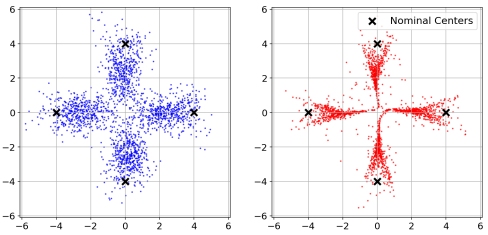


Figure 2: GMM adversarial samples from DRSB-S (left) and DRSB-W (right). Black crosses mark nominal centers.

4.2 Comparison between Wasserstein and Sinkhorn DRSB

We compare the two DRSB variants across perturbation types (Table 2 and Figure 2). DRSB-S achieves lower mean SWD in eight of the ten reported conditions. These results indicate greater robustness of DRSB-S under most of the tested perturbation conditions. Figure 2 shows the computed worst case samples for the GMM task. DRSB-S samples are more dispersed due to the entropy regularization, whereas DRSB-W samples are more concentrated. One possible interpretation is that the broader spread exposes the DRSB-S controller to a wider range of inputs during training, which may help explain its performance under the tested perturbations.

4.3 Noise Augmentation and Generalization to Unseen Noise Levels

We compare DRSB-S with a noise-augmented DSBM baseline on GMM-to-GMM transport to determine whether DRSB provides benefits beyond noise augmentation. Each augmentation baseline uses Gaussian noise at $\delta_{\text{train}} = 1$ or 2, with the number of perturbed samples equal to 20% of the clean dataset size. This makes the number of training steps involving perturbed samples comparable to that in the additional DRSB stage. We evaluate each model at test-time noise levels $\delta \in \{1, 2, 3, 4\}$.

Table 3 shows that augmentation is effective at the noise level used in training, but its benefit does not transfer consistently to unseen levels. Augmentation at $\delta_{\text{train}} = 1$ gives the lowest mean SWD at $\delta = 1$ (1.042), yet performs worse than unaugmented DSBM at every tested $\delta \geq 2$. Augmentation at $\delta_{\text{train}} = 2$ improves performance at the matched level and at some other levels, but is worse than DSBM at the unseen level $\delta = 3$. DRSB-S achieves lower mean SWD than both augmentation baselines at every tested $\delta \geq 2$. DRSB trains against an adaptive adversarial distribution within an ambiguity set, without specifying a test noise level. The results support broader generalization across the tested noise magnitudes than fixed-level augmentation. This robustness comes at the cost of higher Org input SWD (1.165, compared with 0.850 and 0.907 for augmentation).

Table 3: GMM-to-GMM SWD (↓) with noise augmentation (mean ± std, 10 seeds). Test levels $\delta = 3, 4$ are unseen by both augmentation baselines.

	DSBM	DSBM + Aug		DRSB-S
δ		$\delta_{\text{train}} = 1$	$\delta_{\text{train}} = 2$	
Org	0.902±0.242	**0.850**±0.178	0.907±0.153	1.165±0.138
1.0	1.287±0.112	**1.042**±0.246	1.119±0.196	1.150±0.206
2.0	1.842±0.086	1.881±0.145	1.800±0.153	**1.636**±0.157
3.0	2.175±0.118	2.298±0.146	2.220±0.165	**1.953**±0.095
4.0	2.407±0.243	2.425±0.129	2.308±0.138	**2.212**±0.413

4.4 Image-to-Image Translation under Noise

Table 4: FFHQ FID (↓) under Gaussian noise δ (mean ± std, 3 seeds).

	Woman → *Man*		*Man* → *Woman*	
Noise level	DSBM	DRSB	DSBM	DRSB
Org	**25.55**±0.50	26.52±0.50	**29.32**±0.44	29.57±0.48
$\delta = 0.25$	26.59±0.48	**25.82**±0.49	30.02±0.49	**29.69**±0.42
$\delta = 0.5$	26.15±0.49	**25.98**±0.49	30.46±0.43	**29.70**±0.51
$\delta = 1.0$	26.81±0.52	**25.94**±0.47	29.63±0.50	**29.48**±0.51

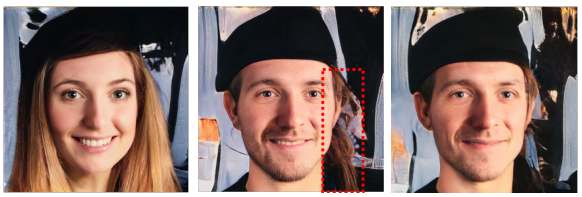


Figure 3: Woman-to-Man translation. Left to right: perturbed input, DSBM, and DRSB.

We apply DRSB to image-to-image translation on the FFHQ dataset (Karras et al., 2019), which contains images at a resolution of 1024×1024 (CC BY-NC-SA 4.0). Following the experimental settings in Theodoropoulos et al. (2024) and Gushchin et al. (2024), we transport images in the 512-dimensional latent space of the ALAE autoencoder (Pidhorskyi et al., 2020). We use FID to evaluate generation quality after decoding the latent vector. The results are shown in Table 4. DRSB outperforms DSBM across all L_2-norm noise levels. DSBM achieves a lower FID score than DRSB on nominal inputs. Figure 3 shows a qualitative example. Even when the effect of the added latent-space noise is visually imperceptible in the decoded input image, DSBM generates visible artifacts, whereas DRSB produces a clean translation.

4.5 Additional Experimental Analysis

Computation cost. Starting from pretrained SB model, the DRSB training requires approximately $0.54\times$ the DSBM pretraining cost on Gaussian transport and $0.16\times$ on FFHQ. On a single NVIDIA RTX 4090, this stage takes 404 s and 1.4 h, respectively. Most computation is spent on network gradient updates, which account for about 80% of runtime on Gaussian transport and over 90% on FFHQ. Worst-case optimization accounts for less than 1.5% in the measured runs. Thus, the adversarial optimization itself contributes little to the measured training cost. Appendix I.1 reports the training configurations and stage-wise costs.

Empirical convergence. On 2D Gaussian transport with $\rho = 10$ and $\epsilon = 12$, the variational objective stabilizes near 312 over 50 epochs. The normalized ambiguity-constraint residual remains below 0.005 at every reported epoch from epoch 25 onward. To examine whether stabilization depends on the decaying learning-rate schedule, we also run 120 iterations at a constant learning rate of 10^{-7}. The objective again settles near 312. These results provide evidence of empirical stabilization under both schedules, but do not establish stationarity of all three blocks or convergence to a saddle point (Appendix I.2).

SOC optimality condition. The adversarial updates rely on the relation $\nabla_{X_0} V(u, X_0; g) = -u_0(X_0)$ from Equation (17). We check this relation for the trained controller by computing the cost gradient through discrete adjoint recursion and comparing it with $-u^\theta(0, X_0)$ on the same Gaussian task. Across five evaluated epochs, the direction cosines are 0.9996–0.9998 and the gradient-to-control magnitude ratios are 0.9310–1.0761. The two quantities therefore agree closely in both direction and magnitude on the evaluated samples. This supports using the SOC relation in the adversarial update for this experiment, but does not prove exact SOC optimality (Appendix I.3).

5 Related Work

Distributionally Robust Optimization. Beginning with the work of Scarf et al. (1957), early DRO focused on moment-based ambiguity sets (Ghaoui et al., 2003; Delage & Ye, 2010). Because moments have limited expressive power, the DRO literature shifted toward distance-based ambiguity sets. Wasserstein DRO (Mohajerin Esfahani & Kuhn, 2018; Zhao & Guan, 2018; Blanchet & Murthy, 2019; Gao & Kleywegt, 2023; Zhang et al., 2025) has attracted attention because of the geometric interpretation of the Wasserstein distance. KL divergence offers convexity and a tractable inner maximization (Ben-Tal et al., 2013; Hu & Hong, 2013). However, its absolute continuity requirement and abstract interpretation are limitations. Recently, Sinkhorn DRO has been proposed to improve computational tractability while modeling realistic uncertainty (Wang et al., 2025).

Robust Optimal Transport. The OT literature studies how to reduce the sensitivity of transport maps to outliers. One approach relaxes the marginal constraints in OT (Mukherjee et al., 2021). In particular, unbalanced OT (UOT) formulations are used for this purpose (Balaji et al., 2020; Fatras et al., 2021; Choi et al., 2023). More closely related to our approach, the soft-constrained Schrödinger Bridge (Garg et al., 2024) relaxes terminal boundary constraints to prevent overfitting to noisy training samples. Wang et al. (2024) use a UOT distance within a DRO framework to handle outliers in training data. These methods and DRSB address different forms of robustness. DRSB focuses on distribution shifts at test time.

6 Conclusion and Limitations

We formulate and solve the DRSB problem, which aims to transport noisy or shifted input distributions to the target distribution. We focus on the Wasserstein and Sinkhorn variants of DRSB. A globally solved Wasserstein inner problem bounds the fixed-terminal-cost objective, while the linearized Sinkhorn inner problem permits Gaussian sampling. Exploring other distances for DRSB is a direction for future work. Minimax adversarial training can be unstable in high-dimensional settings and requires careful hyperparameter tuning and initialization. The additional experiments quantify noise robustness and training overhead. The convergence diagnostics are empirical and do not establish convergence of the full algorithm.

AI DISCLOSURE

In this work, we used generative AI tools to assist with proof writing, refine hypotheses, help design research methodology and experiments, provide feedback on them, implement methods, assist with translation, and interpret results. We also used generative AI tools to create and modify scientific figures and images, suggest experimental parameters, create and edit software code, draft parts of the paper, summarize and analyze existing literature, identify research gaps, and suggest a structure for the paper. We have not used AI to generate new research ideas. LLM-generated code was verified and tested by the author. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Yogesh Balaji, Rama Chellappa, and Soheil Feizi. Robust optimal transport with applications in generative modeling and domain adaptation. *Advances in Neural Information Processing Systems*, 33:12934–12944, 2020.

Randal W Beard and Timothy McLain. Successive Galerkin approximation algorithms for nonlinear optimal and robust control. *International Journal of Control*, 71(5):717–743, 1998.

Richard Bellman. The theory of dynamic programming. *Bulletin of the American Mathematical Society*, 60(6):503–515, 1954.

Aharon Ben-Tal, Dick Den Hertog, Anja De Waegenaere, Bertrand Melenberg, and Gijs Rennen. Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59(2):341–357, 2013.

Jose Blanchet and Karthyek Murthy. Quantifying distributional model risk via optimal transport. *Mathematics of Operations Research*, 44(2):565–600, 2019.

Yongxin Chen, Tryphon T Georgiou, and Michele Pavon. Stochastic control liaisons: Richard Sinkhorn meets Gaspard Monge on a Schrödinger bridge. *SIAM Review*, 63(2):249–313, 2021.

Jaemoo Choi, Jaewoong Choi, and Myungjoo Kang. Generative modeling through the semi-dual formulation of unbalanced optimal transport. *Advances in Neural Information Processing Systems*, 36:42433–42455, 2023.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in Neural Information Processing Systems*, 26, 2013.

Erick Delage and Yinyu Ye. Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Operations Research*, 58(3):595–612, 2010.

Carles Domingo-Enrich, Michal Drozdzal, Brian Karrer, and Ricky TQ Chen. Adjoint matching: Fine-tuning flow and diffusion generative models with memoryless stochastic optimal control. *arXiv preprint arXiv:2409.08861*, 2024.

Kilian Fatras, Thibault Séjourné, Rémi Flamary, and Nicolas Courty. Unbalanced minibatch optimal transport; applications to domain adaptation. In *International Conference on Machine Learning*, pp. 3186–3197. PMLR, 2021.

Rui Gao and Anton Kleywegt. Distributionally robust stochastic optimization with Wasserstein distance. *Mathematics of Operations Research*, 48(2):603–655, 2023.

Jhanvi Garg, Xianyang Zhang, and Quan Zhou. Soft-constrained Schrödinger bridge: a stochastic control approach. In *International Conference on Artificial Intelligence and Statistics*, pp. 4429–4437. PMLR, 2024.

Laurent El Ghaoui, Maksim Oks, and Francois Oustry. Worst-case value-at-risk and robust portfolio optimization: A conic programming approach. *Operations Research*, 51(4):543–556, 2003.

Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in Neural Information Processing Systems*, 27, 2014.

Nikita Gushchin, Sergei Kholkin, Evgeny Burnaev, and Alexander Korotin. Light and optimal Schrödinger bridge matching. In *Forty-first International Conference on Machine Learning*, 2024.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

Zhaolin Hu and L Jeff Hong. Kullback-Leibler divergence constrained distributionally robust optimization. *Available at Optimization Online*, 1(2):9, 2013.

Hilbert J Kappen. Path integrals and symmetry breaking for optimal control theory. *Journal of Statistical Mechanics: Theory and Experiment*, 2005(11):P11011, 2005.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4401–4410, 2019.

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

Guan-Horng Liu, Tianrong Chen, Oswin So, and Evangelos Theodorou. Deep generalized Schrödinger bridge. *Advances in Neural Information Processing Systems*, 35:9374–9388, 2022.

Guan-Horng Liu, Arash Vahdat, De-An Huang, Evangelos A Theodorou, Weili Nie, and Anima Anandkumar. I^2SB: Image-to-Image Schrödinger bridge. In *International Conference on Machine Learning*, 2023.

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

Peyman Mohajerin Esfahani and Daniel Kuhn. Data-driven distributionally robust optimization using the Wasserstein metric: Performance guarantees and tractable reformulations. *Mathematical Programming*, 171(1):115–166, 2018.

Debarghya Mukherjee, Aritra Guha, Justin M Solomon, Yuekai Sun, and Mikhail Yurochkin. Outlier-robust optimal transport. In *International Conference on Machine Learning*, pp. 7850–7860. PMLR, 2021.

Stanislav Pidhorskyi, Donald A Adjeroh, and Gianfranco Doretto. Adversarial latent autoencoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14104–14113, 2020.

Herbert E Scarf, KJ Arrow, and S Karlin. A min-max solution of an inventory problem. Technical report, Rand Corporation Santa Monica, 1957.

Erwin Schrödinger. *Über die Umkehrung der Naturgesetze*. Verlag der Akademie der Wissenschaften in Kommission bei Walter De Gruyter u. Company, 1931.

Yuyang Shi, Valentin De Bortoli, Andrew Campbell, and Arnaud Doucet. Diffusion Schrödinger bridge matching. *Advances in Neural Information Processing Systems*, 36:62183–62223, 2023.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

Panagiotis Theodoropoulos, Nikolaos Komianos, Vincent Pacelli, Guan-Horng Liu, and Evangelos A Theodorou. Feedback Schrödinger bridge matching. *arXiv preprint arXiv:2410.14055*, 2024.

Jie Wang, Rui Gao, and Yao Xie. Sinkhorn distributionally robust optimization. *Operations Research*, 2025.

Zifan Wang, Yi Shen, Michael M Zavlanos, and Karl H Johansson. Outlier-robust distributionally robust optimization via unbalanced optimal transport. *Advances in Neural Information Processing Systems*, 37:52189–52214, 2024.

Wolfram Wiesemann, Daniel Kuhn, and Melvyn Sim. Distributionally robust convex optimization. *Operations Research*, 62(6):1358–1376, 2014.

Luhao Zhang, Jincheng Yang, and Rui Gao. A short and general duality proof for Wasserstein distributionally robust optimization. *Operations Research*, 73(4):2146–2155, 2025.

Chaoyue Zhao and Yongpei Guan. Data-driven risk-averse stochastic optimization with Wasserstein metric. *Operations Research Letters*, 46(2):262–267, 2018.

A Proofs of Results in Section 3

A.1 Proof of Proposition 3.3

Fix an admissible pair (u, μ) and abbreviate $p_1^{u,\mu}$ as p_1^u in this subsection. Assume that p_1^u and ν have positive densities on a common support and finite KL divergence. The function class is unrestricted, and expectations are population expectations.

Lemma A.1 (Variational form of KL)**.** *Let $\Phi(g) := \mathbb{E}_{p_1^u}[g(X)] - \log \mathbb{E}_\nu[e^{g(X)}]$ be defined over measurable functions $g : \mathcal{X} \to \mathbb{R}$ with $\mathbb{E}_{p_1^u}[|g|] < \infty$ and $0 < \mathbb{E}_\nu[e^g] < \infty$. Then, for every such g,*

$$\Phi(g) \le \mathrm{KL}(p_1^u \| \nu) \tag{26}$$

and equality holds when

$$g^*(x) := \log\left(\frac{p_1^u(x)}{\nu(x)}\right). \tag{27}$$

Proof. Introduce a probability measure Q_g such that

$$\frac{dQ_g}{d\nu} = \frac{e^g}{\mathbb{E}_\nu[e^g]} \tag{28}$$

Then g can be written as

$$g(X) = \log \frac{dQ_g}{d\nu} + \log \mathbb{E}_\nu[e^{g(X)}] \tag{29}$$

and $\Phi(g)$ becomes

$$\begin{aligned}
\Phi(g) &= \int g(X)\, dp_1^u(X) - \log \mathbb{E}_\nu[e^{g(X)}] && (30)\\
&= \int \left\{\log \frac{dQ_g}{d\nu} + \log \mathbb{E}_\nu[e^{g(X)}]\right\} dp_1^u(X) - \log \mathbb{E}_\nu[e^{g(X)}] && (31)\\
&= \int \log \frac{dQ_g}{d\nu}\, dp_1^u && (32)\\
&= \int \left(\log \frac{dQ_g}{dp_1^u} + \log \frac{dp_1^u}{d\nu}\right) dp_1^u && (33)\\
&= \mathrm{KL}(p_1^u \| \nu) - \mathrm{KL}(p_1^u \| Q_g) \le \mathrm{KL}(p_1^u \| \nu) && (34)
\end{aligned}$$

Setting $g^*(X) := \log(dp_1^u/d\nu)$ gives $\log \mathbb{E}_\nu[e^{g^*}] = \log \int (dp_1^u/d\nu)\, d\nu = 0$, hence

$$\Phi(g^*) = \int \log\left(\frac{dp_1^u}{d\nu}\right) dp_1^u = \mathrm{KL}(p_1^u \| \nu). \tag{35}$$

□

Therefore, the optimal g recovers the log-density ratio $\log(p_1^u/\nu)$.

Lemma A.2 (Binary Cross-Entropy Minimizer)**.** *The minimizer of the binary cross-entropy loss*

$$\mathcal{L}(g) = \mathbb{E}_{x\sim p_1^u}\left[\log\left(1 + e^{-g(x)}\right)\right] + \mathbb{E}_{x\sim\nu}\left[\log\left(1 + e^{g(x)}\right)\right] \tag{36}$$

is $g^*(x) = \log\left(p_1^u(x)/\nu(x)\right)$.

Proof. For each x, minimize the population-loss integrand over the scalar $z = g(x)$:

$$\begin{aligned}
\ell_x(z) &= p_1^u(x) \log\left(1 + e^{-z}\right) + \nu(x) \log\left(1 + e^z\right), && (37)\\
\ell_x'(z) &= \frac{-p_1^u(x) + e^z \nu(x)}{1 + e^z}, && (38)\\
\ell_x''(z) &= \frac{\left(p_1^u(x) + \nu(x)\right) e^z}{\left(1 + e^z\right)^2} > 0. && (39)
\end{aligned}$$

Thus the pointwise minimum satisfies $e^{g^*(x)}\nu(x) = p_1^u(x)$, giving

$$g^*(x) = \log \frac{p_1^u(x)}{\nu(x)}. \tag{40}$$

Integrating the pointwise inequality proves the result over unrestricted measurable functions. A stationary point in neural-network parameters need not satisfy this pointwise condition. □

Proposition (3.3) follows from Lemma (A.1) and Lemma (A.2). By Lemma (A.2), the minimizer of the binary cross-entropy loss recovers the log-density ratio $g^*(x) = \log(p_1^u(x)/\nu(x))$. Substituting g^* into the variational form in Lemma (A.1) achieves $\Phi(g^*) = \mathrm{KL}(p_1^u \| \nu)$, which gives Equation (7).

A.2 Proof of Theorem 3.4

Proof. For every admissible pair (u, μ), Lemma A.1 gives

$$\mathrm{KL}\left(p_1^{u,\mu} \| \nu\right) = \sup_g \left\{ \mathbb{E}_{p_1^{u,\mu}}[g(X_1)] - \log \mathbb{E}_\nu[e^{g(X_1)}] \right\}. \tag{41}$$

Substitution into Equation (5a), followed by conditioning on X_0, yields

$$\min_u \max_{\mu \in \mathfrak{M}_\epsilon(\hat{\mu})} \sup_g \left\{ \mathbb{E}_{p^{u,\mu}} \left[\int_0^1 \frac{1}{2} \|u_t^\theta(X_t)\|^2 dt + \rho g(X_1) \right] - \rho \log \mathbb{E}_\nu[e^{g(X_1)}] \right\} \tag{42}$$

$$= \min_u \max_{\mu \in \mathfrak{M}_\epsilon(\hat{\mu})} \sup_g \left\{ \mathbb{E}_{X_0 \sim \mu}[V(u, X_0; g)] - \rho \log \mathbb{E}_\nu[e^{g(X_1)}] \right\}, \tag{43}$$

which is Equation (8). This substitution is pointwise in (u, μ) and does not interchange the minimization over u with an inner maximization. For a fixed pair, the normalized maximizer is

$$g^*(x; u, \mu) = \log \frac{p_1^{u,\mu}(x)}{\nu(x)}, \qquad \log \mathbb{E}_\nu[e^{g^*(X_1; u, \mu)}] = 0. \tag{44}$$

Because this maximizer depends on both u and μ, it cannot be fixed while optimizing them. For a separately fixed g, the logarithmic term is constant with respect to u and μ and may be omitted from their optimization. This gives the fixed-g subproblem in Equation (10), rather than an equivalent formulation with a single fixed g^* for all pairs. □

B Details on the Derivation of DRSB Variants

Throughout this section, g is fixed and $V(u, x)$ abbreviates $V(u, x; g)$; in an inner distribution update, u is also fixed. The dual identities below concern this fixed-cost subproblem, under the finiteness and regularity assumptions of the cited DRO duality results.

B.1 Details on Wasserstein DRSB

We start from the fixed-g Wasserstein DRSB subproblem

$$\min_u \max_{\mu \in \mathfrak{M}_\epsilon(\hat{\mu})} \mathbb{E}_{p^u} \left[\int_0^1 \frac{1}{2} \|u_t^\theta(X_t)\|^2 dt + \rho g(X_1) \right] \tag{45a}$$

$$\text{s.t. } dX_t = u_t^\theta(X_t)dt + \sigma dW_t,\ X_0 \sim \mu \tag{45b}$$

$$\mathfrak{M}_\epsilon(\hat{\mu}) = \{\mu : W_2^2(\mu, \hat{\mu}) \le \epsilon\} \tag{45c}$$

where the squared Wasserstein-2 distance is defined as

$$W_2^2(\mu, \hat{\mu}) := \min_{\gamma \in \Gamma(\mu, \hat{\mu})} \int \|X_0 - \hat{X}_0\|^2 \gamma(dX_0, d\hat{X}_0). \tag{46}$$

Consider the inner maximization

$$\max_{\mu \in \mathfrak{M}_\epsilon(\hat{\mu})} \mathbb{E}_{p^u}\left[\int_0^1 \frac{1}{2}\|u_t^\theta(X_t)\|^2 dt + \rho g(X_1)\right] \tag{47a}$$

$$\text{s.t. } dX_t = u_t^\theta(X_t)dt + \sigma dW_t,\ X_0 \sim \mu \tag{47b}$$

$$\mathfrak{M}_\epsilon(\hat{\mu}) = \{\mu : W_2^2(\mu, \hat{\mu}) \le \epsilon\} \tag{47c}$$

Define the Lagrangian as

$$\mathcal{L}(\mu, \lambda) := \mathbb{E}_{\mathbf{X}\sim p^u}\left[\int_0^1 \frac{1}{2}\|u_t^\theta(X_t)\|^2 dt + \rho g(X_1)\right] - \lambda\left(\int \|X_0 - \hat{X}_0\|^2 \pi(dX_0, d\hat{X}_0) - \varepsilon\right) \tag{48}$$

Conditioning on X_0,

$$\mathcal{L}(\mu, \lambda) = \mathbb{E}_{X_0\sim\mu}\left[V(u, X_0)\right] - \lambda\left(\int \|X_0 - \hat{X}_0\|^2 \pi(dX_0, d\hat{X}_0) - \varepsilon\right) \tag{49}$$

where $V(u, X_0) := \mathbb{E}_{\mathbf{X}_{\cdot|0}\sim p^u_{\cdot|0}}\left[\int_0^1 \frac{1}{2}\|u_t^\theta(X_t)\|^2 dt + \rho g(X_1)\middle| X_0\right]$ is the cost function at $t = 0$. Then we have

$$\mathcal{L}(\mu, \lambda) = \lambda\varepsilon + \int V(u, X_0)\pi(X_0, \hat{X}_0)\, dX_0\, d\hat{X}_0 - \lambda \int \|X_0 - \hat{X}_0\|^2 \pi(X_0, \hat{X}_0)\, dX_0\, d\hat{X}_0 \tag{50}$$

The dual problem is

$$\min_{\lambda\ge 0}\ \lambda\varepsilon + \max_{\mu}\ \mathbb{E}_{(X_0,\hat{X}_0)\sim\pi}\left[V(X_0) - \lambda\|X_0 - \hat{X}_0\|^2\right] \tag{51}$$

By the interchangeability principle (Zhang et al., 2025), the dual problem can be written as

$$\min_{\lambda\ge 0}\ \lambda\varepsilon + \int \max_{X_0}\left\{V(X_0) - \lambda\|X_0 - \hat{X}_0\|^2\right\}\hat{\mu}(\hat{X}_0)\, d\hat{X}_0 \tag{52}$$

$$= \min_{\lambda\ge 0}\ \lambda\varepsilon - \int \min_{X_0}\left\{\lambda\|X_0 - \hat{X}_0\|^2 - V(X_0)\right\}\hat{\mu}(\hat{X}_0)\, d\hat{X}_0 \tag{53}$$

For the empirical distribution $\hat{\mu} = \frac{1}{N}\sum_{i=1}^N \delta_{\hat{X}_0^i}$, the expectation reduces to a finite sum, giving

$$\min_{\lambda\ge 0}\ \lambda\varepsilon - \frac{1}{N}\sum_{i=1}^N \min_{X_0}\left\{\lambda\|X_0 - \hat{X}_0^i\|^2 - V(X_0)\right\}. \tag{54}$$

B.1.1 Derivation of Update Rules (Equation (18) and Equation (19))

Equivalently, the samplewise maximization objective is

$$\max_{X_0} \mathcal{L}_1(X_0) := V(X_0) - \lambda\|X_0 - \hat{X}_0^i\|^2. \tag{55}$$

Taking the gradient of $\mathcal{L}_1(X_0)$ with respect to X_0 gives

$$\nabla_x \mathcal{L}_1(X_0) = \nabla_x V(X_0) - 2\lambda(X_0 - \hat{X}_0^i) \tag{56}$$

If $u = u^*$ solves the SOC problem for this fixed g and its value function is differentiable, Equation (17) gives

$$\nabla_x \mathcal{L}_1(X_0) = -u_0^*(X_0) - 2\lambda(X_0 - \hat{X}_0^i) \tag{57}$$

Replacing u^* by the trained drift gives the implemented ascent update. This substitution is approximate unless the fixed-g SOC optimality condition is satisfied:

$$X_0 \leftarrow X_0 + \eta\left[-u_0^\theta(X_0) - 2\lambda(X_0 - \hat{X}_0^i)\right]$$

where η is the gradient ascent step size. Each sample update in Equation (18) requires one evaluation of u_0^θ; total overhead also depends on the number of sample and dual updates.

Similarly, we can derive the update rule in Equation (19) from the optimality condition for the worst-case optimization. For Equation (55), the first-order optimality condition is

$$\nabla_x \mathcal{L}_1(X_0^{(i)*}) = \nabla_x V(X_0^{(i)*}) - 2\lambda(X_0^{(i)*} - \hat{X}_0^i) = 0 \tag{58}$$

Therefore, the worst-case sample satisfies

$$\nabla_x V(X_0^{(i)*}) = 2\lambda(X_0^{(i)*} - \hat{X}_0^i) \tag{59}$$

Define the objective for the dual variable as

$$\inf_{\lambda \geq 0} f(\lambda) := \lambda\varepsilon - \frac{1}{N}\sum_{i=1}^{N}\left[\lambda\|X_0^{(i)*} - \hat{X}_0^i\|^2 - V(X_0^{(i)*})\right] \tag{60}$$

By taking the gradient with respect to λ, we have

$$\begin{aligned}\frac{df(\lambda)}{d\lambda} = \varepsilon + \frac{1}{N}\sum_{i=1}^{N}\Bigg[&\nabla_x V(X_0^{(i)*})\frac{dX_0^{(i)*}}{d\lambda} - \|X_0^{(i)*} - \hat{X}_0^i\|^2 \\ &-2\lambda(X_0^{(i)*} - \hat{X}_0^i)\frac{dX_0^{(i)*}}{d\lambda}\Bigg]\end{aligned} \tag{61}$$

From Equation (59), we can cancel out the $\frac{dX_0^{(i)*}}{d\lambda}$ terms:

$$\frac{df(\lambda)}{d\lambda} = \varepsilon - \frac{1}{N}\sum_{i=1}^{N}\|X_0^{(i)*} - \hat{X}_0^i\|^2 \tag{62}$$

Consequently, we can perform gradient descent on λ:

$$\lambda \leftarrow \lambda - \zeta\left(\varepsilon - \frac{1}{N}\sum_{i=1}^{N}\|X_0^{(i)*} - \hat{X}_0^i\|^2\right)$$

where ζ is the gradient descent step size. This update is valid only while λ remains nonnegative; otherwise feasibility requires projection onto $[0, \infty)$.

B.2 Details on Sinkhorn DRSB

We start from the fixed-g Sinkhorn DRSB subproblem

$$\min_{u} \max_{\mu \in \mathfrak{M}_\epsilon^S(\hat{\mu})} \quad \mathbb{E}_{p^u}\left[\int_0^1 \frac{1}{2}\|u_t^\theta(X_t)\|^2 dt + \rho g(X_1)\right] \tag{63a}$$

$$\text{subject to} \quad dX_t = u_t^\theta(X_t)dt + \sigma dW_t, \quad X_0 \sim \mu \tag{63b}$$

$$\mathfrak{M}_\epsilon^S(\hat{\mu}) = \{\mu : \mathcal{W}_\kappa(\hat{\mu}, \mu) \leq \epsilon\} \tag{63c}$$

where $\mathcal{W}_\kappa(\hat{\mu}, \mu)$ is the Sinkhorn distance with entropy regularization parameter κ:

$$\mathcal{W}_\kappa(\hat{\mu}, \mu) = \min_{\gamma \in \Gamma(\hat{\mu}, \mu)}\left\{\mathbb{E}_{(\hat{X}_0, X_0)\sim\gamma}\left[\|\hat{X}_0 - X_0\|^2 + \kappa \log\left(\frac{d\gamma(\hat{X}_0, X_0)}{d\hat{\mu}(\hat{X}_0)dX_0}\right)\right]\right\}. \tag{64}$$

Denote the value of the inner maximization by $\mathcal{V}$.

$$\mathcal{V} := \max_{\mu \in \mathfrak{M}_\epsilon^S(\hat{\mu})} \mathbb{E}_{X_0\sim\mu}\left[V(X_0)\right] \tag{65}$$

$$\begin{aligned}&= \max_{\gamma}\min_{\lambda \geq 0} \lambda\varepsilon + \mathbb{E}_{\hat{X}_0\sim\hat{\mu}}\mathbb{E}_{X_0\sim\gamma(X_0|\hat{X}_0)} \\ &\qquad\left[V(X_0) - \lambda\|\hat{X}_0 - X_0\|^2 - \lambda\kappa\log\left(\frac{d\gamma(X_0|\hat{X}_0)}{dX_0}\right)\right]\end{aligned} \tag{66}$$

Under the assumptions for strong duality in Sinkhorn DRO (Wang et al., 2025), we switch the order:

$$\mathcal{V} = \min_{\lambda \geq 0} \lambda\varepsilon + \mathbb{E}_{\hat{X}_0 \sim \hat{\mu}} \Bigg[\max_{\gamma(X_0|\hat{X}_0)} \mathbb{E}_{X_0 \sim \gamma(X_0|\hat{X}_0)} \Bigg[V(X_0) - \lambda \|\hat{X}_0 - X_0\|^2 - \lambda\kappa \log \left(\frac{d\gamma(X_0|\hat{X}_0)}{dX_0} \right) \Bigg] \Bigg] \tag{67}$$

For $\lambda > 0$, $\kappa > 0$, and a finite normalizing integral, the entropy-regularized inner maximization admits the following closed-form solution. Boundary cases at $\lambda = 0$ are understood by limits. The dual form is

$$\mathcal{V} = \inf_{\lambda > 0} \lambda\varepsilon + \mathbb{E}_{\hat{X}_0 \sim \hat{\mu}} \left[\lambda\kappa \log \int e^{\left(V(X_0) - \lambda \|X_0 - \hat{X}_0\|^2\right)/\lambda\kappa} dX_0 \right] \tag{68}$$

and the optimal conditional distribution is

$$\gamma^*(X_0|\hat{X}_0) \propto e^{\left(V(X_0) - \lambda \|X_0 - \hat{X}_0\|^2\right)/\lambda\kappa} \tag{69}$$

B.2.1 Second-order approximation

We approximate $V(X_0)$ up to second order:

$$V(X_0) \approx V(\hat{X}_0) + \nabla V(\hat{X}_0)^\top \delta X_0 + \frac{1}{2} \delta X_0^\top \nabla_{xx} V(\hat{X}_0)\, \delta X_0 \tag{70}$$

where $\delta X_0 = X_0 - \hat{X}_0$. Define $g := \frac{\nabla V(\hat{X}_0)}{\lambda\kappa}$ and $A := \frac{2}{\kappa} I_d - \frac{\nabla_{xx} V(\hat{X}_0)}{\lambda\kappa}$. Here g denotes a local gradient vector; the terminal-cost function remains fixed. For the quadratic approximation and $A \succ 0$, the corresponding conditional distribution is Gaussian. The equalities below apply to the approximated cost:

$$\gamma^*(X_0|\hat{X}_0) \propto \exp\left(g^\top \delta X_0 - \frac{1}{2} \delta X_0^\top A\, \delta X_0 \right) \tag{71}$$

$$\propto \exp\left(-\frac{1}{2} (\delta X_0 - A^{-1} g)^\top A (\delta X_0 - A^{-1} g) \right) \tag{72}$$

As a result,

$$\gamma^*(X_0|\hat{X}_0) = \mathcal{N}(\hat{X}_0 + A^{-1} g,\, A^{-1}) \tag{73}$$

with the full expression

$$\gamma^*(X_0|\hat{X}_0) = \mathcal{N}\left(\hat{X}_0 + \left(\frac{2}{\kappa} I - \frac{\nabla_{xx} V}{\lambda\kappa} \right)^{-1} \frac{\nabla V}{\lambda\kappa},\, \left(\frac{2}{\kappa} I - \frac{\nabla_{xx} V}{\lambda\kappa} \right)^{-1} \right) \tag{74}$$

where V and its derivatives are evaluated at $\hat{X}_0$.

B.2.2 Simplification of the dual form

Denote $I := \int e^{\left(V(X_0) - \lambda \|X_0 - \hat{X}_0\|^2\right)/\lambda\kappa} dX_0$. For the same quadratic approximation, completing the square in terms of g and A gives the following approximate integral and dual objective:

$$I = e^{\frac{V(\hat{X}_0)}{\lambda\kappa}} \int \exp\left\{ g^\top \delta X_0 - \frac{1}{2} \delta X_0^\top A\, \delta X_0 \right\} dX_0 \tag{75}$$

$$= e^{\frac{V(\hat{X}_0)}{\lambda\kappa} + \frac{1}{2} g^\top A^{-1} g} \int \exp\left\{ -\frac{1}{2} (\delta X_0 - A^{-1} g)^\top A (\delta X_0 - A^{-1} g) \right\} dX_0 \tag{76}$$

Evaluating the Gaussian integral under the condition $A \succ 0$ gives

$$I = (2\pi)^{d/2} \det(A^{-1})^{1/2} \exp\left(\frac{V(\hat{X}_0)}{\lambda\kappa} + \frac{1}{2} g^\top A^{-1} g \right) \tag{77}$$

Therefore,

$$\lambda\kappa \log I = V(\hat{X}_0) + \frac{\lambda\kappa}{2} g^\top A^{-1} g + \lambda\kappa \log\left((2\pi)^{d/2} \det(A^{-1})^{1/2}\right) \tag{78}$$

and the dual objective becomes

$$\mathcal{V} = \inf_{\lambda>0} \lambda\left(\varepsilon + \frac{\kappa d}{2}\log 2\pi\right) + \mathbb{E}_{\hat{X}_0\sim\hat{\mu}}\left[V(\hat{X}_0) + \frac{\lambda\kappa}{2} g^\top A^{-1} g + \frac{\lambda\kappa}{2}\log\det(A^{-1})\right] \tag{79}$$

Solving for λ in this form is complicated due to the dependence of A^{-1} on λ.

B.2.3 First-order approximation

In high-dimensional cases, computing the Hessian $\nabla_{xx} V(\hat{X}_0)$ is expensive. For the linearized cost, the Hessian term is omitted, giving $A = \frac{2}{\kappa} I_d$ and $A^{-1} = \frac{\kappa}{2} I_d$. The following expressions are exact for this linearized inner problem. The conditional distribution simplifies to

$$\gamma^*(X_0|\hat{X}_0) = \mathcal{N}\left(\hat{X}_0 + \frac{1}{2\lambda}\nabla V(\hat{X}_0)\,, \frac{\kappa}{2} I\right) \tag{80}$$

The λ optimization reduces to

$$\inf_{\lambda>0} \lambda\left(\varepsilon + \frac{\kappa d}{2}\log \pi\kappa\right) + \mathbb{E}_{\hat{X}_0\sim\hat{\mu}}\left[V(\hat{X}_0) + \frac{\|\nabla V(\hat{X}_0)\|^2}{4\lambda}\right] \tag{81}$$

For $\varepsilon + \frac{\kappa d}{2}\log(\pi\kappa) > 0$ and $\mathbb{E}_{\hat{\mu}}[\|\nabla V\|^2] > 0$, this yields the positive closed-form solution

$$\lambda^* = \sqrt{\left(\varepsilon + \frac{\kappa d}{2}\log \pi\kappa\right)^{-1} \mathbb{E}_{\hat{X}_0\sim\hat{\mu}}\left[\frac{\|\nabla V(\hat{X}_0)\|^2}{4}\right]} \tag{82}$$

C Stochastic Optimal Control Theory

Here we derive Equation (17) using stochastic optimal control theory. We refer the reader to Kappen (2005) for more details. The terminal cost is fixed, and the HJB derivation assumes a sufficiently smooth value function and admissible controls for the verification argument. To apply this section to DRSB, replace the terminal cost g below by ρg. The identity holds for an optimal control and its optimal value, not for an arbitrary intermediate neural drift. Consider the stochastic optimal control (SOC) problem

$$\min_u \quad \mathbb{E}\left[\int_0^1 \frac{1}{2}\|u_t(X_t)\|^2 dt + g(X_1)\right] \tag{83a}$$

$$\text{subject to} \quad dX_t = u_t(X_t)dt + \sigma dW_t \tag{83b}$$

$$X_0 \sim \mu, \tag{83c}$$

Define the cost function as

$$c_t(X_t, u_t) := \mathbb{E}\left[\int_t^1 \frac{1}{2}\|u_s(X_s)\|^2 ds + g(X_1)\right] \tag{84}$$

The value function at time t is the optimal cost from time t to 1.

$$V_t(x) := \min_{\substack{u(s)\\ t\le s\le 1}} \mathbb{E}\left[\int_t^1 \frac{1}{2}\|u_s(X_s)\|^2 ds + g(X_1) \;\middle|\; X_t = x\right] \tag{85}$$

The cost function depends on the control u_t, whereas the value function is obtained by minimizing over the control. We split the integral into the intervals from t to $t + dt$ and from $t + dt$ to 1. Substituting $t = 1$ gives $V_1(x) = g(x)$.

$$V_t(X_t) = \min_{\substack{u(s)\\ t\le s\le 1}} \mathbb{E}\left[\int_t^{t+dt} \frac{1}{2}\|u_s(X_s)\|^2 ds + \int_{t+dt}^1 \frac{1}{2}\|u_s(X_s)\|^2 ds + g(X_1) \;\middle|\; X_t\right] \tag{86}$$

Using Bellman's principle of optimality (Bellman, 1954), we have

$$V_t(X_t) = \min_{\substack{u(s) \\ t \le s \le 1}} \mathbb{E}\Bigg[\int_t^{t+dt} \frac{1}{2}\|u_s(X_s)\|^2 ds + \min_{\substack{u(s) \\ t+dt \le s \le 1}} \mathbb{E}\left[\int_{t+dt}^{1} \frac{1}{2}\|u_s(X_s)\|^2 ds + g(X_1) \,\middle|\, X_{t+dt} \right] \,\middle|\, X_t \Bigg] \tag{87}$$

$$= \min_{\substack{u(s) \\ t \le s \le 1}} \mathbb{E}\left[\int_t^{t+dt} \frac{1}{2}\|u_s(X_s)\|^2 ds + V_{t+dt}(X_{t+dt}) \,\middle|\, X_t \right] \tag{88}$$

Applying a Taylor expansion to $V_{t+dt}(X_{t+dt})$ at (X_t, t) gives

$$\mathbb{E}[V_{t+dt}(X_{t+dt})] \approx \mathbb{E}\left[V_t(X_t) + \frac{\partial V_t(X_t)}{\partial t} dt + (\nabla V_t)^\top dX_t + \frac{1}{2}(dX_t)^\top \nabla^2 V (dX_t) \right] \tag{89}$$

Applying stochastic calculus to Equation (83b) gives

$$\mathbb{E}[dX_t] = u_t(X_t)dt \tag{90}$$

$$\mathbb{E}[(dX_t)(dX_t)^\top] = \sigma^2 \mathbf{I} dt \tag{91}$$

Therefore,

$$\mathbb{E}[V_{t+dt}(X_{t+dt})] = \mathbb{E}\left[V_t(X_t) + \frac{\partial V_t(X_t)}{\partial t} dt + (\nabla V_t)^\top u_t(X_t) dt + \frac{\sigma^2}{2} \Delta V_t dt \right] \tag{92}$$

Taking the limit as $dt \to 0$ and substituting Equation (92) into Equation (88) gives

$$-\frac{\partial V_t(X_t)}{\partial t} = \min_u \left\{ \frac{1}{2}\|u_t(X_t)\|^2 + (\nabla V_t)^\top u_t(X_t) + \frac{\sigma^2}{2} \Delta V_t \right\} \tag{93}$$

which is the Hamilton–Jacobi–Bellman (HJB) equation. By differentiating the right-hand side, we obtain the optimal control u^*:

$$u_t^*(X_t) = -\nabla V_t(X_t) \tag{94}$$

At time $t = 0$, we have Equation (17).

D Solving the SOC with Adjoint Matching

D.1 Adjoint Matching

This section summarizes results from the original adjoint matching paper. We refer the reader to Domingo-Enrich et al. (2024), Sections 5.2 and E, for more details. The initial distribution and terminal cost are fixed throughout this section. We assume sufficient differentiability and integrability for pathwise differentiation with additive noise and interchange with expectation. Gradients are column vectors and $\nabla_x u$ denotes the Jacobian. Adjoint matching solves the optimal control problem

$$\min_{u^\theta} \ \mathbb{E}_{p^u}\left[\int_0^1 \frac{1}{2}\|u_t^\theta(X_t^u)\|^2 dt + g(X_1^u) \right] \tag{95}$$

$$\text{s.t. } dX_t^u = u_t^\theta(X_t^u)dt + \sigma_t dW_t \tag{96}$$

$$X_0 \sim \mu \tag{97}$$

Here, we explicitly include the superscript u in the state notation X^u, unlike in previous sections. This notation is useful in the derivation of the adjoint matching objective.

D.1.1 Derivation of the Adjoint System

We introduce a Lagrange multiplier process $a : \Omega \times [0,1] \to \mathbb{R}^n$ for the dynamics. The Lagrangian of the optimal control problem is

$$\begin{aligned}\mathcal{L}(u_t^\theta, a_t; \boldsymbol{X}_t^u, t) &= \int_t^1 \frac{1}{2}\|u_t^\theta(X_t^u)\|^2 dt + g(X_1^u) \\ &\quad - \int_t^1 a_t^\top \left(dX_t^u - u_t^\theta(X_t^u)dt - \sigma_t dW_t\right) \qquad (98)\\ &= \int_t^1 \frac{1}{2}\|u_t^\theta(X_t^u)\|^2 dt + g(X_1^u) - \int_t^1 a_t^\top dX_t^u \\ &\quad + \int_t^1 a_t^\top u_t^\theta(X_t^u)dt + \int_t^1 \sigma_t a_t^\top dW_t \qquad (99)\end{aligned}$$

Applying integration by parts to the second integral gives

$$\begin{aligned}\mathcal{L}(u_t^\theta, a_t, \boldsymbol{X}_t^u) &= \int_t^1 \frac{1}{2}\|u_t^\theta(X_t^u)\|^2 dt + g(X_1^u) - a_1^\top X_1 + a_t^\top X_t \\ &\quad + \int_t^1 (X_t^u)^\top \left(\frac{da_t}{dt}\right) dt + \int_t^1 a_t^\top u_t^\theta(X_t^u)dt + \int_t^1 \sigma_t a_t^\top dW_t \qquad (100)\end{aligned}$$

Taking the first variation with respect to the trajectory $\boldsymbol{X}_t$ yields

$$\begin{aligned}\frac{\delta}{\delta \boldsymbol{X}_t^u}\mathcal{L} &= (\nabla_x g(X_1^u) - a_1)^\top h_1 \\ &\quad + \int_t^1 \left(\frac{da_t}{dt} + \left(\nabla_x u_t^\theta(X_t^u)\right)^\top a_t + \nabla_x \left(\frac{1}{2}\|u_t^\theta(X_t^u)\|^2\right)\right)^\top h_t \, dt = 0, \qquad (101)\end{aligned}$$

which must hold for all h_t. Thus, the coefficients of h_t must be zero, and the adjoint system

$$\begin{cases} \dfrac{da_t}{dt} = -\left(\nabla_x u_t^\theta(X_t^u)\right)^\top a_t - \nabla_x \left(\frac{1}{2}\|u_t^\theta(X_t^u)\|^2\right) & (102a)\\ a_1 = \nabla_x g(X_1^u) & (102b)\end{cases}$$

is derived.

Next, we add the dynamics constraint to the cost-to-go function $\mathcal{J}(u, x_t)$, using the adjoint state as its multiplier.

$$\begin{aligned}\mathcal{J}(u, x_t) &:= \int_t^1 \frac{1}{2}\|u_t^\theta(X_t^u)\|^2 dt + g(X_1^u) \qquad (103)\\ &= \int_t^1 \frac{1}{2}\|u_t^\theta(X_t^u)\|^2 dt + g(X_1^u) - \int_t^1 a_t^\top \left(dX_t^u - u_t^\theta(X_t^u)dt - \sigma_t dW_t\right) \qquad (104)\\ &= \int_t^1 \frac{1}{2}\|u_t^\theta(X_t^u)\|^2 dt + g(X_1^u) - a_1^\top X_1^u + a_t^\top X_t^u \\ &\quad + \int_t^1 (X_t^u)^\top \left(\frac{da_t}{dt}\right) dt + \int_t^1 a_t^\top u_t^\theta(X_t^u)dt + \int_t^1 \sigma_t a_t^\top dW_t \qquad (105)\end{aligned}$$

Taking the gradient with respect to the initial state $X_t = x_t$ yields

$$\begin{aligned}\nabla_{x_t}\mathcal{J}(u, x_t) &= \int_t^1 (\nabla_{x_t} X_s^u)^\top \left(\frac{da_s}{ds} + \left(\nabla_x u_s^\theta(X_s^u)\right)^\top a_s + \nabla_x \left(\frac{1}{2}\|u_s^\theta(X_s^u)\|^2\right)\right) ds \\ &\quad + (\nabla_{x_t} X_1^u)^\top (\nabla_x g(X_1^u) - a_1) + a_t \qquad (106)\\ &= a_t \qquad \text{(by Equation (102))}. \qquad (107)\end{aligned}$$

As a result, the solution of the adjoint system in Equation (102) is

$$a_t(\boldsymbol{X}_t^u, u^\theta) = \nabla_{x_t}\left(\int_t^1 \frac{1}{2}\|u_s^\theta(X_s^u)\|^2 ds + g(X_1^u)\right) \qquad (108)$$

D.1.2 Derivation of the Basic Adjoint Matching Objective

Now we derive the basic adjoint matching objective

$$\mathcal{L}_{\text{basic-AM}}(u^\theta; \boldsymbol{X}^{\bar{u}}) = \frac{1}{2}\int_0^1 \|u_t^\theta(X_t^{\bar{u}}) + a_t(\boldsymbol{X}^{\bar{u}}, \bar{u})\|^2 dt, \qquad \boldsymbol{X}^{\bar{u}} \sim p^u,\ \bar{u} = \texttt{stopgrad}(u^\theta) \tag{109}$$

The trajectories are generated by the neural network with fixed parameters ($\bar{u}$), detached from gradient computation during simulation. This avoids retaining the full simulation computation graph. It does not remove the costs of trajectory generation and network updates.

To derive this objective, we start from the optimal control objective

$$\mathcal{L}(u) = \int_0^1 \frac{1}{2}\|u_t^\theta(X_t^u)\|^2 dt + g(X_1^u) \tag{110}$$

We follow a similar procedure by adding the dynamics constraint with the adjoint state $a_t(\boldsymbol{X}^{\bar{u}}, \bar{u})$ as its multiplier. Here, $a_t(\boldsymbol{X}^{\bar{u}}, \bar{u})$ is independent of θ.

$$\mathcal{L}(u) = \int_0^1 \frac{1}{2}\|u_t^\theta(X_t^u)\|^2 dt + g(X_1^u) - \int_0^1 a_t^\top \left(dX_t^u - u_t^\theta(X_t^u)dt - \sigma_t dW_t\right) \tag{111}$$

$$= \int_0^1 \frac{1}{2}\|u_t^\theta(X_t^u)\|^2 dt + g(X_1^u) - a_1^\top X_1^u + a_0^\top X_0^u + \int_0^1 (X_t^u)^\top \left(\frac{da_t}{dt}\right) dt + \int_0^1 a_t^\top u_t^\theta(X_t^u)dt + \int_0^1 \sigma_t a_t^\top dW_t \tag{112}$$

Differentiating with respect to the neural network parameter θ gives

$$\frac{d\mathcal{L}(u)}{d\theta} = \frac{d}{d\theta}\left(\int_0^1 \frac{1}{2}\|u_t^\theta(X_t^u)\|^2 dt + g(X_1^u) - a_1^\top X_1^u + a_0^\top X_0^u + \int_0^1 (X_t^u)^\top \left(\frac{da_t}{dt}\right) dt + \int_0^1 a_t^\top u_t^\theta(X_t^u)dt + \int_0^1 \sigma_t a_t^\top dW_t \right) \tag{113}$$

$$= \frac{d}{d\theta}\left(\int_0^1 \frac{1}{2}\|u_t^\theta(X_t^u)\|^2 dt + g(X_1^u) - a_1^\top X_1^u + \int_0^1 (X_t^u)^\top \left(\frac{da_t}{dt}\right) dt + \int_0^1 a_t^\top u_t^\theta(X_t^u)dt \right) \tag{114}$$

Here, X_0 is fixed and therefore independent of θ. Differentiating the first term gives

$$\frac{d}{d\theta}\left(\int_0^1 \frac{1}{2}\|u_t^\theta(X_t^u)\|^2 dt\right) = \int_0^1 \left(\frac{\partial u_t^\theta}{\partial \theta}(X_t^{\bar{u}}) + \nabla_x u_t^\theta(X_t^{\bar{u}})\frac{\partial X_t^u}{\partial \theta}\right)^\top u_t^\theta(X_t^{\bar{u}})dt \tag{115}$$

$$= \int_0^1 \left(\frac{\partial u_t^\theta}{\partial \theta}(X_t^{\bar{u}})\right)^\top u_t^\theta(X_t^{\bar{u}}) + \left(\frac{\partial X_t^u}{\partial \theta}\right)^\top \nabla_x \left(\frac{1}{2}\|u_t^\theta(X_t^{\bar{u}})\|^2\right) dt. \tag{116}$$

The X_1 terms are

$$\frac{d}{d\theta}\left(g(X_1^u) - a_1^\top X_1^u\right) = \left(\frac{\partial X_1^u}{\partial \theta}\right)^\top \left(\nabla_x g(X_1^{\bar{u}}) - a_1\right) = 0 \qquad \text{(by Equation (102b))}\,. \tag{117}$$

The last two integrals are

$$\frac{d}{d\theta}\left(\int_0^1 (X_t^u)^\top \frac{da_t}{dt}dt + \int_0^1 a_t^\top u_t^\theta(X_t^u)dt\right) = \int_0^1 \left(\frac{\partial u_t^\theta}{\partial \theta}(X_t^{\bar{u}})\right)^\top a_t dt + \int_0^1 \left(\frac{\partial X_t^u}{\partial \theta}\right)^\top \left(\frac{da_t}{dt} + \left(\nabla_x u_t^\theta(X_t^{\bar{u}})\right)^\top a_t\right) dt. \tag{118}$$

Gathering all the terms, we have

$$\frac{d\mathcal{L}(u)}{d\theta} = \int_0^1 \left(\frac{\partial u_t^\theta}{\partial \theta}(X_t^{\bar{u}})\right)^\top \left(u_t^\theta(X_t^{\bar{u}}) + a_t\right) dt + \int_0^1 \left(\frac{\partial X_t^u}{\partial \theta}\right)^\top \left(\frac{da_t}{dt} + \left(\nabla_x u_t^\theta(X_t^{\bar{u}})\right)^\top a_t + \nabla_x \left(\frac{1}{2}\|u_t^\theta(X_t^{\bar{u}})\|^2\right)\right) dt \tag{119}$$

$$= \int_0^1 \left(\frac{\partial u_t^\theta}{\partial \theta}(X_t^{\bar{u}})\right)^\top \left(u_t^\theta(X_t^{\bar{u}}) + a_t\right) dt \qquad \text{(by Equation (102a))}\,. \tag{120}$$

Throughout this derivation, the $\frac{\partial X_t^u}{\partial \theta}$ terms account for the dependence of trajectories on θ. These terms cancel by the adjoint equations. The remaining terms are evaluated using trajectories generated by $\bar{u} = \texttt{stopgrad}(u)$. We can rewrite Equation (120) as

$$\frac{d\mathcal{L}(u)}{d\theta} = \int_0^1 \frac{\partial}{\partial \theta}\left(\frac{1}{2}\|u_t^\theta(X_t^{\bar{u}})\|^2\right) + \frac{\partial}{\partial \theta}\left(a_t^\top u_t^\theta(X_t^{\bar{u}})\right) dt \tag{121}$$

$$= \frac{d}{d\theta}\left(\int_0^1 \frac{1}{2}\|u_t^\theta(X_t^{\bar{u}})\|^2 + a_t^\top u_t^\theta(X_t^{\bar{u}}) dt\right) \tag{122}$$

$$= \frac{d}{d\theta}\left(\int_0^1 \frac{1}{2}\|u_t^\theta(X_t^{\bar{u}})\|^2 + a_t^\top u_t^\theta(X_t^{\bar{u}}) + \frac{1}{2}\|a_t\|^2 dt\right) \quad (\because a_t \text{ is independent of } \theta) \tag{123}$$

$$= \frac{d}{d\theta}\left(\int_0^1 \frac{1}{2}\|u_t^\theta(X_t^{\bar{u}}) + a_t(\boldsymbol{X}^{\bar{u}}, \bar{u})\|^2 dt\right) \tag{124}$$

Thus, the basic AM objective in Equation (109) can be used to optimize u_t^θ using gradient descent.

D.1.3 Connection to Optimal Control, the Lean Adjoint State, and the Adjoint Matching Objective

The pathwise adjoint system applies to any sufficiently regular admissible control. We show the consistency of the functional stationarity condition with a verified SOC optimum. During the following variation, trajectories and adjoints are held fixed, and the perturbation is a state-feedback function $h_t = h_t(X_t)$. The critical-point condition for adjoint matching follows from the first variation of $\mathbb{E}_{p^u}[\mathcal{L}_{\text{basic-AM}}]$.

$$\frac{\delta \mathbb{E}_{p^u}[\mathcal{L}_{\text{basic-AM}}]}{\delta u} = \frac{d}{d\epsilon}\left(\mathbb{E}_{p^u}\left[\frac{1}{2}\int_0^1 \|(u_t^\theta + \epsilon h_t)(X_t^{\bar{u}}) + a_t(\boldsymbol{X}^{\bar{u}})\|^2 dt\right]\right)\Big|_{\epsilon=0} \tag{125}$$

$$= \mathbb{E}_{p^u}\left[\int_0^1 h_t^\top (u_t^\theta(X_t^{\bar{u}}) + a_t(\boldsymbol{X}^{\bar{u}})) dt\right] \tag{126}$$

$$= \mathbb{E}_{p^u}\left[\int_0^1 h_t^\top (u_t^\theta(X_t^{\bar{u}}) + \mathbb{E}_{\boldsymbol{X}\sim p^u}[a_t(\boldsymbol{X}^{\bar{u}})|X_t^{\bar{u}}]) dt\right] \tag{127}$$

The last equality follows from the tower property, $\mathbb{E}_X[X] = \mathbb{E}_Y[\mathbb{E}_X[X|Y]]$. Thus, if u^* satisfies the following condition, then u^* is a critical point of the adjoint matching objective.

$$u_t^*(x_t) = -\mathbb{E}_{\boldsymbol{X}^*\sim p^{u^*}}[a_t(\boldsymbol{X}^{u^*})|X_t^{u^*} = x_t] \tag{128}$$

For a verified SOC optimum u^*, the conditional expected cost-to-go equals V_t. Using Equation (108) and interchanging differentiation and expectation gives

$$u_t^*(x_t) = -\mathbb{E}_{p^{u^*}}\left[a_t(\boldsymbol{X}^{u^*}) \mid X_t^{u^*} = x_t\right] \tag{129}$$

$$= -\mathbb{E}_{p^{u^*}}\left[\nabla_{x_t}\mathcal{J}(u^*, x_t) \mid X_t^{u^*} = x_t\right] \tag{130}$$

$$= -\nabla_{x_t}\mathbb{E}_{p^{u^*}}\left[\mathcal{J}(u^*, x_t) \mid X_t^{u^*} = x_t\right] \tag{131}$$

$$= -\nabla_{x_t} V_t(x_t). \tag{132}$$

Algorithm 2 Adjoint Matching (Domingo-Enrich et al., 2024)

Require: Parameterized drift u^θ, differentiable terminal cost $g(x)$, nominal distribution $\hat{\mu}$, target distribution ν, replay buffer $\mathcal{B}$, number of epochs E.
Initialize: u^θ

1: **for** epoch in $1, 2, \ldots, E$ **do**
2: Sample $\{\hat{X}_0^i\}_{i=1}^M \sim \hat{\mu}$
3: Sample $\{t^i\}_{i=1}^M \sim \mathcal{U}[0,1]$.
4: Simulate (5b) to get $\{(\tilde{X}_0^i, t^i, X_t^i, X_1^i)\}_{i=1}^M$.
5: Update replay buffer $\mathcal{B} \leftarrow \mathcal{B} \cup \{(t^i, X_t^i, X_1^i)\}_{i=1}^M$
6: Take gradient steps $\nabla_\theta \mathcal{L}_{\text{AM}}$ to minimize $\mathcal{L}_{\text{AM}}(\theta) = \mathbb{E}_{(t,X_t,X_1)\sim\mathcal{B}}\left[\|u^\theta(t,X_t) + \nabla g(X_1)\|^2\right]$.
7: **end for**

Thus the SOC optimum satisfies the functional stationarity condition. This calculation does not establish global optimality of an arbitrary stationary point in neural-network parameters.

We can further simplify the adjoint system to the *lean adjoint system*. Multiplying Equation (128) by $(\nabla_x u_t^*(x_t))^\top$, we have

$$(\nabla_x u_t^*(x_t))^\top u_t^*(x_t) = -\mathbb{E}_{p^{u^*}}\left[(\nabla_x u_t^*(x_t))^\top a_t(\boldsymbol{X}^{u^*}) \mid X_t^{u^*} = x_t\right] \tag{133}$$

$$\iff \mathbb{E}_{p^{u^*}}\left[-\nabla_x\left(\frac{1}{2}\|u_t^*(x_t)\|^2\right) - (\nabla_x u_t^*(x_t))^\top a_t(\boldsymbol{X}^{u^*}) \mid X_t^{u^*} = x_t\right] = 0. \tag{134}$$

The adjoint ODE has zero conditional mean drift at the SOC optimum. This motivates the lean adjoint below; it does not imply that each pathwise adjoint is constant. The functional critical-point result for lean Adjoint Matching is provided by Domingo-Enrich et al. (2024), Proposition 7, under its stated assumptions.

$$\begin{cases} \dfrac{d\tilde{a}_t}{dt} = 0 & \text{(135a)} \\ \tilde{a}_1 = \nabla_x g(X_1^u) & \text{(135b)} \end{cases}$$

where $\tilde{a}$ is called the *lean adjoint state*. The lean adjoint state is no longer equivalent to the gradient of the cost-to-go function. Since the base SDE has no drift and the state cost is zero, the lean adjoint state is constant in time, with $\tilde{a}_t = \nabla_x g(X_1^u)$. The resulting adjoint matching objective is

$$\mathcal{L}_{\text{AM}}(u^\theta; \boldsymbol{X}^{\bar{u}}) = \frac{1}{2}\int_0^1 \|u_t^\theta(X_t^{\bar{u}}) + \tilde{a}_t(\boldsymbol{X}^{\bar{u}})\|^2 dt = \frac{1}{2}\int_0^1 \|u_t^\theta(X_t^{\bar{u}}) + \nabla_x g(X_1^{\bar{u}})\|^2 dt \tag{136}$$
$$\text{where} \quad \boldsymbol{X}^{\bar{u}} \sim p^u, \quad \bar{u} = \texttt{stopgrad}(u^\theta)$$

Replacing the terminal cost by ρg and taking expectation over trajectories gives Equation (15), up to the constant factor $1/2$. The final adjoint matching algorithm is outlined in Algorithm 2.

E Full DRSB Algorithm

The main DRSB training loop is given in Algorithm 3. Each iteration consists of three steps: worst-case distribution update, terminal cost network update, and control network update. DRSB-W and DRSB-S differ only in the worst-case subroutine. Algorithm 4 details the Wasserstein worst-case optimization, which iteratively updates each sample via gradient ascent (Equation (18)) and adjusts λ via gradient descent (Equation (19)). Algorithm 5 details the Sinkhorn worst-case optimization, which computes λ^* in closed form (Equation (25)) and samples from the Gaussian worst-case distribution (Equation (24)).

F Implementation Details

All experiments are conducted on a single NVIDIA RTX 4090 GPU. The drift network u^θ and the terminal cost network g^ϕ are both parameterized as multi-layer perceptrons. For the 2D transport tasks (Gaussian and GMM), the baseline SB model is pretrained with the DSBM algorithm (Shi

Algorithm 3 Distributionally Robust Schrödinger Bridge

Require: Parameterized drift u^θ, parameterized terminal cost g^ϕ, nominal distribution $\hat{\mu}$, target distribution ν, replay buffer $\mathcal{B}$, number of epochs E.
Initialize: u^θ with a pretrained SB model from $\hat{\mu}$ to ν, g^ϕ, λ

for epoch in $1, 2, \ldots, E$ **do**
  $\{\tilde{X}_0^i\}_{i=1}^M, \lambda \leftarrow$ `WassersteinWorstCase`(u^θ, λ) (Algorithm 4) *or*
  $\{\tilde{X}_0^i\}_{i=1}^M \leftarrow$ `SinkhornWorstCase`(u^θ)
  (Algorithm 5)
  Sample $\{t^i\}_{i=1}^M \sim \mathcal{U}[0,1]$.
  Simulate (5b) to get $\{(\tilde{X}_0^i, t^i, X_t^i, X_1^i)\}_{i=1}^M$.
  Update replay buffer $\mathcal{B} \leftarrow \mathcal{B} \cup \{(t^i, X_t^i, X_1^i)\}_{i=1}^M$
  Take gradient steps $\nabla_\phi \mathcal{L}_D$ to minimize
  $\mathcal{L}_D(\phi) = \mathbb{E}_{X_1 \sim \mathcal{B}}\left[\log(1 + e^{-g^\phi(X_1)})\right] + \mathbb{E}_{X_1 \sim \nu}\left[\log(1 + e^{g^\phi(X_1)})\right]$
  Take gradient steps $\nabla_\theta \mathcal{L}_{\text{AM}}$ to minimize
  $\mathcal{L}_{\text{AM}}(\theta) = \mathbb{E}_{(t, X_t, X_1) \sim \mathcal{B}}\left[\|u^\theta(t, X_t) + \rho \nabla g^\phi(X_1)\|^2\right]$.
end for

Algorithm 4 Wasserstein worst-case optimization

Require: Minibatch size β, number of λ updates L, worst-case update threshold τ.
Function `WassersteinWorstCase`(u^θ, λ)

Prepare result set $\mathcal{R} = \{\}$
for λ update iteration $1, 2, \ldots, L$ **do**
  Sample $\{\hat{X}_0^j\}_{j=1}^\beta \sim \hat{\mu}$
  $\{\tilde{X}_0^j\}_{j=1}^\beta \leftarrow \{\hat{X}_0^j\}_{j=1}^\beta$
  repeat
    $\{X_0^j\}_{j=1}^\beta \leftarrow \{\tilde{X}_0^j\}_{j=1}^\beta$
    $\left\{\tilde{X}_0^j \leftarrow X_0^j + \eta\left[-u^\theta(0, X_0^j) - 2\lambda(X_0^j - \hat{X}_0^j)\right]\right\}_{j=1}^\beta$
  until $\frac{1}{\beta}\sum_{j=1}^\beta \|\tilde{X}_0^j - X_0^j\|^2 < \tau$.
  Update $\lambda \leftarrow \lambda - \zeta\left[\varepsilon - \frac{1}{\beta}\sum_{j=1}^\beta \|\tilde{X}_0^j - \hat{X}_0^j\|^2\right]$
  Save $\mathcal{R} \leftarrow \mathcal{R} \cup \{\tilde{X}_0^j\}_{j=1}^\beta$
end for
Return: $\mathcal{R} = \{\tilde{X}_0^i\}_{i=1}^{L\beta}, \lambda$

et al., 2023), and the DRSB training is initialized from this pretrained model following the successive approximation scheme in Algorithm 1. For the FFHQ image translation task, the model operates in the 512-dimensional latent space of the ALAE autoencoder (Pidhorskyi et al., 2020).

F.1 Practical Techniques for Adversarial Training

The minimax structure of DRSB can make training unstable, particularly in high-dimensional settings. The following techniques stabilize the adversarial optimization.

Gradual adversarial training. The adversary in DRSB is the worst-case optimization, and its penalty is controlled by the dual variable λ. Gradually decreasing λ from a large initial value increases the adversarial shift slowly. In DRSB-W, this occurs naturally through the λ gradient descent steps (Figure 4a). In DRSB-S, the closed-form λ^* can produce extreme worst-case shifts from the start. To mitigate this, we linearly increase the ambiguity set radius ε over the first few iterations (Figure 4b).

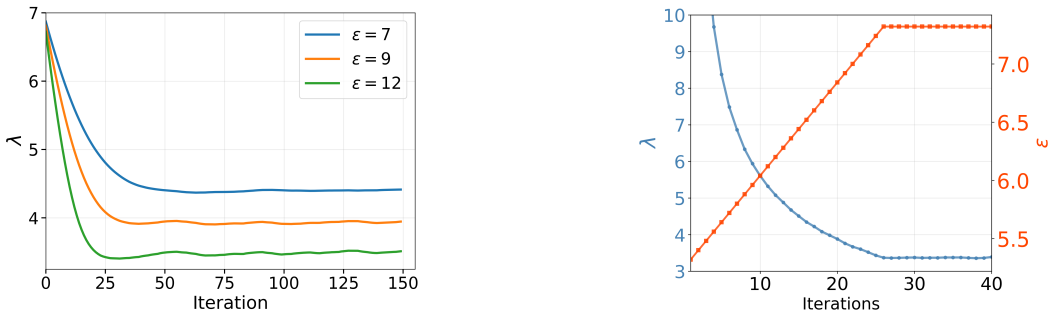
Pretraining the terminal cost network g^ϕ. Consistent with findings in robust control (Beard & McLain, 1998), proper initialization is critical for convergence in adversarial training. We pretrain g^ϕ using the pretrained SB drift $\bar{u}$ by minimizing the joint objective $\mathcal{L}_D(\phi) + \xi\mathcal{L}_{\text{AM}}(\phi)$, where ξ is a

Algorithm 5 Sinkhorn worst-case optimization

Require: Batch size B, number of worst-case samples per clean sample w.
Function `SinkhornWorstCase`(u^θ)

1: Sample $\{\hat{X}_0^j\}_{j=1}^B \sim \hat{\mu}$
2: Compute $\lambda \leftarrow \sqrt{\left(\varepsilon + \frac{\kappa d}{2}\log \pi\kappa\right)^{-1} \frac{1}{B}\sum_{j=1}^B \frac{\|u^\theta(0,\hat{X}_0^j)\|^2}{4}}$
3: Sample $\left\{\{\tilde{X}_0^l\}_{l=1}^w \sim \mathcal{N}\left(\hat{X}_0^j - \frac{u_0^\theta(\hat{X}_0^j)}{2\lambda}, \frac{\kappa}{2}I\right)\right\}_{j=1}^B$

Return: $\{\tilde{X}_0^i\}_{i=1}^{wB}$



(a) DRSB-W λ evolution.

(b) DRSB-S ε schedule.

Figure 4: Training dynamics. (a) Evolution of λ in DRSB-W. (b) ε-schedule for DRSB-S and corresponding λ evolution.

weight hyperparameter. Generated samples for $\mathcal{L}_D(\phi)$ are obtained from $\bar{u}$, and $\mathcal{L}_{\text{AM}}(\phi)$ regresses $\rho\nabla g^\phi(X_1)$ to the negative drift $-\bar{u}(t, X_t)$.

Regularization with a pretrained SB model. Worst-case samples can push the controller away from the correct data manifold, leading to semantic degradation. Since the pretrained SB control $\bar{u}$ captures the data manifold direction, we regularize u^θ to remain close to $\bar{u}$ via the loss $\mathcal{L}_{\text{AM}}(\theta) + \alpha\mathbb{E}[\|u^\theta - \bar{u}\|^2]$, where α is a weight hyperparameter.

F.2 Sensitivity to κ in DRSB-S

The entropy regularization parameter κ in the Sinkhorn distance controls the support of the worst-case distribution. As $\kappa \to 0$, the Sinkhorn distance approaches the Wasserstein distance, and the ambiguity set preserves geometric interpretability. As κ increases, higher-entropy couplings γ are favored, and the worst-case distribution has wider support.

Training stability is sensitive to κ. From Equation (24), larger κ produces a more diffuse worst-case distribution. If κ is too large, training becomes unstable due to extreme deviation from the nominal samples.

In high-dimensional settings, κ interacts with the dimension d through the optimal dual variable in Equation (25). The term $h(\kappa, \varepsilon) := \varepsilon + \frac{\kappa d}{2}\log(\pi\kappa)$ must satisfy $h(\kappa, \varepsilon) > 0$ for λ^* to be well-defined. If $h(\kappa, \varepsilon)$ is too large, λ^* becomes small and the worst-case distribution deviates excessively from the nominal in Equation (24). Setting κ near 0 or $1/\pi$ eliminates the $\log(\pi\kappa)$ contribution and reduces the sensitivity to d.

Table 5 reports FID scores on the FFHQ Woman→Man task for two values of κ and several values of the ambiguity set radius ε. Both $\kappa = 0.005$ and $\kappa = 0.32$ yield stable training and robustness to input noise. In the FFHQ task with $d = 512$, values of κ far from 0 or $1/\pi$ (e.g., $\kappa = 0.25$) or $\varepsilon > 30$ cause training to diverge.

Table 5: FID (↓) sensitivity to κ and ε on FFHQ Woman→Man under noise δ.

Model	Org	$\delta = 0.25$	0.50	1.00
DSBM	**25.55**±0.50	26.59±0.48	26.15±0.49	26.81±0.52
DRSB-S κ=0.005, ε=9	26.76±0.49	26.28±0.44	26.60±0.52	27.15±0.49
DRSB-S κ=0.005, ε=12	26.36±0.48	26.48±0.48	26.10±0.50	26.66±0.52
DRSB-S κ=0.005, ε=15	26.08±0.47	26.57±0.50	26.35±0.53	26.40±0.53
DRSB-S κ=0.32, ε=7	26.70±0.48	25.93±0.48	26.05±0.47	26.05±0.50
DRSB-S κ=0.32, ε=12	26.52±0.50	**25.82**±0.49	**25.98**±0.49	25.94±0.47
DRSB-S κ=0.32, ε=15	26.26±0.50	25.88±0.44	26.51±0.52	**25.89**±0.49

G Additional Experiments

The main text reports selected perturbation conditions in Table 1 and Table 2. This section provides the full results across all noise levels and shift directions.

G.1 Noise Robustness

Table 6 reports SWD under L_2-norm noise at eight levels ($\delta = 0.5$ to 4.0) for both tasks. DSBM performs best at low noise in the Gaussian task, while DRSB-W and DRSB-S outperform DSBM at higher noise levels. In the GMM task, DRSB-S outperforms both DSBM and DRSB-W at all noise levels except $\delta = 0.5$.

Table 6: SWD (↓) on 2D transport tasks under varying L_2-norm noise levels δ (mean ± std over 10 seeds). **Bold** indicates best performance; gray indicates outperforming the baseline DSBM.

Model	δ=0.5	1.0	1.5	2.0	2.5	3.0	3.5	4.0
(a) Gaussian-to-Gaussian								
DSBM	**0.052**±0.006	**0.072**±0.009	**0.128**±0.012	**0.214**±0.010	**0.313**±0.015	0.431±0.012	0.555±0.013	0.684±0.013
DRSB-W	0.080±0.010	0.093±0.012	0.138±0.013	0.216±0.013	0.315±0.012	0.425±0.013	0.548±0.013	0.678±0.013
DRSB-S	0.131±0.011	0.134±0.014	0.165±0.015	0.227±0.016	0.314±0.015	**0.414**±0.015	**0.522**±0.015	**0.638**±0.016
(b) GMM-to-GMM								
DSBM	1.164±0.137	1.287±0.112	1.597±0.077	1.842±0.086	2.051±0.116	2.175±0.118	2.254±0.108	2.407±0.243
DRSB-W	**1.096**±0.216	1.220±0.172	1.547±0.114	1.821±0.123	1.985±0.111	2.139±0.128	2.222±0.166	2.315±0.236
DRSB-S	1.191±0.195	**1.150**±0.206	**1.293**±0.145	**1.636**±0.157	**1.836**±0.127	**1.953**±0.095	**2.013**±0.118	**2.212**±0.413

G.2 Shift Robustness

Table 7 reports SWD under directional distribution shifts and worst-case inputs. DRSB-S achieves the lowest SWD under most shift directions in both tasks. DRSB-W outperforms DSBM on the original GMM distribution, suggesting that the worst-case update can act as a regularizer on complex manifolds.

H Worst-Case Cost Analysis

H.1 Worst-Case Cost Interpretation

Remark H.1 (Fixed-terminal-cost bound)**.** Fix a controller u and terminal cost g. If $\tilde{\mu}$ globally maximizes $\mathbb{E}_{X_0\sim\mu}[V(u, X_0; g)]$ over $\mathfrak{M}_\epsilon(\hat{\mu})$, then

$$\forall \mu \in \mathfrak{M}_\epsilon(\hat{\mu}), \qquad \mathbb{E}_{X_0\sim\mu}[V(u, X_0; g)] \le \mathbb{E}_{X_0\sim\tilde{\mu}}[V(u, X_0; g)]. \tag{137}$$

This is a direct consequence of global inner optimality. It does not require an optimal controller or imply that the ambiguity constraint is active. Approximate adversarial updates need not attain this bound. Since g is fixed, the inequality is not a certificate for the full objective with terminal KL divergence.

Table 7: SWD (↓) on 2D transport tasks under distribution shifts (mean ± std over 10 seeds). Org: original distribution; Up/Down/Left/Right: directional shift; W-wst: DRSB-W worst case; S-wst: DRSB-S worst case. **Bold** indicates best performance; gray indicates outperforming the baseline DSBM.

Model	Org	Up	Down	Left	Right	W-wst	S-wst
				(a) Gaussian-to-Gaussian			
DSBM	**0.038**±0.007	0.665±0.012	0.673±0.020	0.666±0.011	**0.651**±0.009	1.081±0.021	0.951±0.013
DRSB-W	0.144±0.007	0.691±0.020	0.698±0.020	0.517±0.014	0.792±0.014	0.972±0.007	0.812±0.012
DRSB-S	0.123±0.012	**0.647**±0.018	**0.640**±0.014	**0.501**±0.014	0.769±0.016	**0.881**±0.018	**0.710**±0.013
				(b) GMM-to-GMM			
DSBM	0.902±0.242	6.167±0.233	5.798±0.245	5.917±0.127	5.917±0.132	2.086±0.166	1.988±0.100
DRSB-W	**0.885**±0.226	5.949±0.262	5.865±0.282	5.967±0.153	5.785±0.120	**1.503**±0.052	1.946±0.100
DRSB-S	1.165±0.138	**5.887**±0.247	**5.698**±0.203	**5.840**±0.095	**5.675**±0.185	1.791±0.102	**1.478**±0.100

Applying this bound to a test distribution requires verifying its membership in the ambiguity set. For empirical training and test distributions, the Wasserstein distance is well defined and can be computed from their samples. With the squared Euclidean transport cost, the Wasserstein ambiguity set constrains the average squared displacement from the nominal distribution.

H.2 Empirical Cost Comparison

We compare the measured costs of the computed adversarial distribution, four shifted distributions (Up, Down, Left, Right), and the nominal distribution within the Wasserstein ambiguity set. For fixed u^θ and g^ϕ, we estimate $\mathbb{E}[\sum_t \frac{1}{2}\|u_t\|^2\Delta t + \rho g^\phi(X_1)]$ over 1000 uniform time steps. In Figure 5, the computed adversarial distribution has the largest measured cost among these distributions. This finite comparison does not verify global inner optimality or certify the full objective with terminal KL divergence.

The control cost in Figure 5 saturates as ρ increases, while the terminal-cost term receives greater weight.

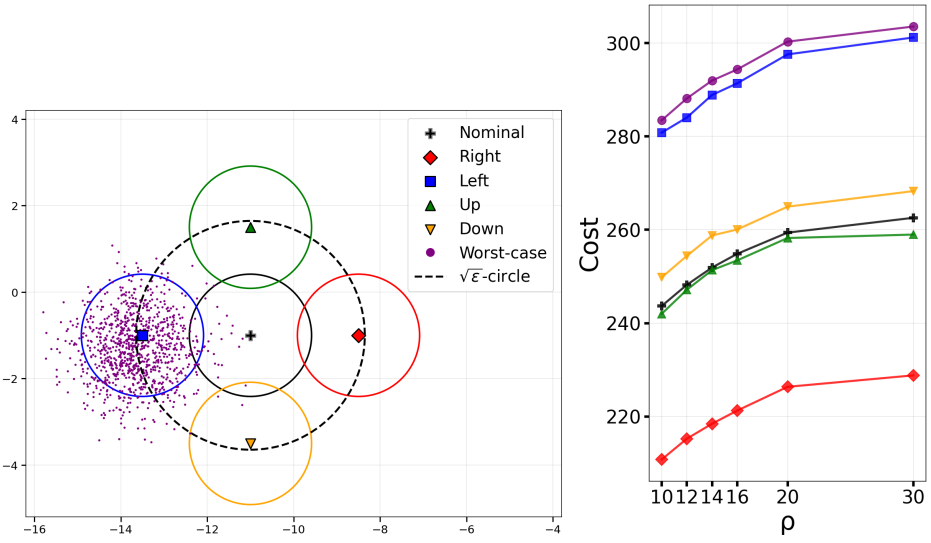


Figure 5: Cost comparison within the Wasserstein ambiguity set. **Left**: Evaluated distributions with the ambiguity-set boundary (dashed). **Right**: Measured cost versus ρ.

I Additional Experimental Analysis

I.1 Measured Computation Cost

DRSB starts from a pretrained DSBM drift. Table 8 separates the two stages. The DRSB stage costs approximately $0.54\times$ the Gaussian DSBM time and $0.16\times$ the FFHQ DSBM time. Including pretraining gives approximately $1.54\times$ and $1.16\times$, respectively. Gradient updates account for about 80% of Gaussian training time and over 90% on FFHQ, while worst-case optimization accounts for less than 1.5%. DRSB-W uses up to 90 worst-case gradient steps in these runs; DRSB-S uses Gaussian sampling with a closed-form λ update under the linear approximation. FFHQ DRSB draws four Monte Carlo samples per clean sample. These measurements describe the tested workloads and do not establish how runtime scales with dimension.

Table 8: Measured training costs. DRSB columns report the additional stage after DSBM pretraining. Trajectory counts follow the reported training configurations.

	Gaussian DSBM	Gaussian DRSB	FFHQ DSBM	FFHQ DRSB
Iterations	140	50	70	50
Gradient steps (total)	224k	101k	3.36M	320k
Trajectories	4,096	1,024	20,480	40,960
Wall-clock time	754 s	404 s	8.5 h	1.4 h
Stage time / DSBM time	$1\times$	$0.54\times$	$1\times$	$0.16\times$

I.2 Empirical Convergence

We examine the 2D Gaussian task with $\rho = 10$, $\epsilon = 12$, and 50 epochs, using a cosine learning-rate schedule from 3×10^{-7} to 10^{-7}. In Table 9, $R_k = \left|1 - \frac{1}{N\epsilon}\sum_i \|X_0^{i*} - \hat{X}_0^i\|^2\right|$ is the constraint residual, $G_k = \mathbb{E}_{p_1^{u^{(k)},\mu^{(k)}}}[g^\phi(X_1)]$ is the mean terminal critic output, and U_k is the mean squared control magnitude on a fixed clean evaluation set. At the reported epochs $k \geq 25$, $R_k < 0.005$. The final dual variable is $\lambda_{50} = 3.483$, close to the value 3.496 predicted by the active-constraint SOC relation. The terminal critic output and control magnitude change less at later epochs. These summaries measure stabilization, not gradients or stationarity residuals for all three blocks.

Table 9: Reported update diagnostics on the 2D Gaussian task.

k	5	10	15	20	25	30	35	40	45	50
R_k	0.260	0.319	0.240	0.111	0.002	0.004	0.002	0.000	0.002	0.001
λ_k	6.601	5.696	4.535	3.627	3.427	3.457	3.477	3.488	3.494	3.483
G_k	0.752	1.232	1.735	2.142	2.068	1.757	1.672	1.631	1.524	1.550
U_k	22.94	22.92	23.19	23.74	24.10	24.22	24.29	24.36	24.32	24.30

We also evaluate the variational objective

$$\mathcal{L}(u, \mu, g) = \mathbb{E}_{X_0 \sim \mu}[V(u, X_0; g)] - \rho \log \mathbb{E}_{X_1 \sim \nu}[e^{g(X_1)}], \tag{138}$$

where $V(u, x; g)$ includes the control effort and terminal cost ρg. The control effort is estimated with a Riemann sum. The sampled values in Table 10 stabilize near 312 after an observed peak of 313.44 at $k = 32$. A separate run with constant learning rate 10^{-7} for 120 iterations also settles near 312. This provides an additional check beyond the decreasing learning-rate schedule, without proving convergence of the block algorithm.

Table 10: Variational objective at the reported epochs.

k	5	10	20	30	40	50
$\mathcal{L}$	270.99	276.16	304.99	311.99	311.44	311.37

I.3 Empirical Check of the SOC Optimality Condition

The simplified adversarial update replaces $\nabla_{X_0} V(u, X_0; g)$ by $-u^\theta(0, X_0)$ using Equation (17). The identity is exact for an optimal controller of the fixed-terminal-cost SOC problem, but finite training need not satisfy it. We assess this approximation on the same 2D Gaussian experiment by computing $\nabla_{X_0} V$ with a discrete adjoint recursion requiring one vector-Jacobian product per time step. The controller and terminal cost are held fixed for this gradient evaluation.

Table 11 reports both direction cosines and gradient-to-control magnitude ratios. The cosine tests whether the surrogate points in the cost-gradient direction, while the ratio tests its scale. Across the evaluated epochs, cosines are 0.9996–0.9998 and ratios are 0.9310–1.0761. At epoch 50, the cosine is 0.9998 and the ratio is 1.0089, corresponding to a gradient magnitude approximately 0.9% larger than the control magnitude. These measurements support approximate satisfaction of the SOC relation on the evaluated samples. They do not establish exact SOC optimality or its validity across other tasks and input distributions.

Table 11: SOC consistency diagnostics on the 2D Gaussian task.

k	10	20	30	40	50
$\cos\left(\nabla_{X_0} V, -u^\theta(0, X_0)\right)$	0.9997	0.9996	0.9996	0.9997	0.9998
$\|\nabla_{X_0} V\| / \|u^\theta(0, X_0)\|$	0.9310	1.0761	1.0249	1.0278	1.0089