DIFFUSION SCHRÖDINGER BRIDGE MATCHING: WHEN RESAMPLING FAILS

Teodora Reu & Michael Bronstein
Department of Computer Science
University of Oxford

Francisco Vargas
Xaira Technologies

ABSTRACT

Diffusion Schrödinger Bridge Matching (DSBM) is often argued to resist error accumulation because each forward and backward update resamples from the true marginals. In this paper, we study that claim in a controlled toy model that replaces learned stochastic bridge dynamics with perturbed exact Gaussian optimal transport maps, allowing rigorous tracking of the integrated marginals.

Within this toy setting, we show that resampling alone is not a universal mechanism for robustness. Under constant shift perturbations, both the resampling and no-resampling variants drift linearly, while the resampling variant accumulates error twice as fast because forward and backward biases compound. Under variance-scaling perturbations, the behavior differs qualitatively across variants, showing that stability depends on the perturbation structure rather than on resampling alone.

We complement the theory with experiments using exact OT maps and learned neural regressors. These experiments reproduce the predicted drift phenomena and show additional shape distortion for learned maps. Overall, our results should be interpreted as a toy counterexample to the claim that bidirectional resampling by itself explains robustness to non-additive error.

1 INTRODUCTION

Iterative methods (Vargas et al., 2021; De Bortoli et al., 2021; Liu, 2022; De Bortoli et al., 2024) have become a cornerstone of generative modeling, motivated by accelerating inference in Rectified Flows (Liu, 2022; Bansal et al., 2024; Liu et al., 2024), and attaining the entropic optimal transport (eOT) plan via Diffusion Schrödinger Bridge Matching (DSBM) (De Bortoli et al., 2024).

While Rectified Flows are known to accumulate error due to the loss of marginal matching (Seong et al., 2025; Zhu et al., 2024), DSBM, leveraging its bidirectional structure and eOT objective, is frequently claimed to mitigate this issue (Peluchetti, 2023; Gushchin et al., 2024; Kholkin et al., 2024). This claimed robustness is attributed to DSBM's bidirectional resampling strategy, resampling from true marginals at each forward and backward step, rather than its stochastic (SDE) dynamics, which in error-free settings theoretically converge to eOT pairings (De Bortoli et al., 2024). However, rigorous proof of this robustness claim under perturbations is lacking. Consequently, recent work has explored bidirectional variants of Rectified Flow Zhu et al. (2024) or hybrid methods incorporating both directions Seong et al. (2025), which report empirically improved marginal matching, but provide no theoretical guarantees for why these marginals are preserved better.

Contributions Our contributions are as follows.

1. We introduce a controlled toy model of DSBM that isolates the effect of bidirectional resampling by replacing learned stochastic bridge dynamics with perturbed exact Gaussian OT maps.
2. In this setting, we prove that resampling alone does not guarantee robustness: under constant shift perturbations, the resampling variant exhibits linear drift and accumulates error twice as fast as the no-resampling baseline (in Proposition 1).
3. We also analyze variance-scaling perturbations, showing that the resulting behavior depends on the perturbation structure rather than on resampling alone (in Proposition 2).

4. Finally, we compare exact-map and learned-map implementations, showing that learned maps inherit the same drift patterns while introducing additional off-support shape distortion (in Section 4).

2 BACKGROUND

Notation. Let $\mathbb{R}^d$ denote d-dimensional Euclidean space. We use $\mathcal{P}(\mathbb{R}^d)$ for the set of probability distributions on $\mathbb{R}^d$. The source and target distributions are π_0 and π_1, with $X_0 \sim \pi_0$ and $X_1 \sim \pi_1$. A transport map is $T : \mathbb{R}^d \to \mathbb{R}^d$, and $T_\# \pi_0$ denotes the pushforward of π_0 by T. Interpolants are written $X_t = I(X_0, X_1, t)$, with $t \in [0, 1]$. We write $\mathbb{E}[\cdot]$ for expectation, $\mathrm{Var}[\cdot]$ for variance, and ∇_θ for gradients with respect to parameters θ. Bold symbols (e.g., $\mathbf{x}$) denote vectors. For a vector field $v : \mathbb{R}^d \times [0, 1] \to \mathbb{R}^d$, we write ODE-Solve$[v](x_0, t{=}1)$ for the endpoint at $t = 1$ of the ODE $dX_t = v(X_t, t)\, dt$ with initial condition $X_0 = x_0$. Similarly, for diffusion coefficient $\sigma : [0, 1] \to \mathbb{R}_{\geq 0}$, we write SDE-Solve$[v, \sigma](x_0, t{=}1)$ for the (random) endpoint of the SDE $dX_t = v(X_t, t)\, dt + \sigma(t)\, dB_t$ with $X_0 = x_0$, where B_t is standard Brownian motion.

Optimal Transport and Entropy-Regularized OT. Let π_0, π_1 be probability measures on $\mathbb{R}^d$. The Monge problem (Monge, 1781) seeks a transport map T minimizing:

$$\inf_T \int \|T(x) - x\|^2 \, d\pi_0(x) \quad \text{s.t.} \quad T_\# \pi_0 = \pi_1, \tag{1}$$

where $T_\# \pi_0$ denotes the pushforward of π_0 under T. The Kantorovich (Kantorovitch, 1958) relaxation introduces couplings $\pi \in \Pi(\pi_0, \pi_1)$ and solves:

$$\mathcal{W}_2^2(\pi_0, \pi_1) = \inf_{\pi \in \Pi(\pi_0, \pi_1)} \mathbb{E}_{(X_0, X_1) \sim \pi}[\|X_1 - X_0\|^2], \tag{2}$$

where $\Pi(\pi_0, \pi_1)$ denotes the set of joint distributions with marginals π_0 and π_1. Entropy-regularized Optimal Transport (eOT) adds a Kullback–Leibler divergence penalty $\mathcal{W}_2^\epsilon(\pi_0, \pi_1) = \inf_{\pi \in \Pi(\pi_0, \pi_1)} \mathbb{E}_\pi[\|X_1 - X_0\|^2] + \epsilon D_{\mathrm{KL}}(\pi \| \pi_0 \otimes \pi_1)$.

When π_0 is absolutely continuous, the Monge and Kantorovich problems admit the same deterministic optimal plan. A dynamic formulation describes OT as evolving a path $\{X_t\}_{t \in [0,1]}$ connecting $X_0 \sim \rho_0$ and $X_1 \sim \rho_1$. For convex costs, the optimal path is given by the straight-line interpolant $X_t = (1 - t)X_0 + tX_1$ (McCann, 1997).

Conditional Flow Matching. Given the interpolants $X_t = I(X_0, X_1, t) = \alpha_t X_0 + \beta_t X_1 + \sigma_t \epsilon$ where $\epsilon \sim \mathcal{N}(0, I)$, the CFM objective Lipman et al. (2022) is:

$$\mathcal{L}_{CFM}(v) = \mathbb{E}_{t \sim \mathcal{U}(0,1), (X_0, X_1) \sim \pi, \epsilon}[\|(X_1 - X_t)/(1 - t) - v_\theta(X_t, t)\|^2]. \tag{3}$$

2.1 ITERATIVE ALGORITHMS

1. Rectified Flows (or ReFlow). Liu et al. (2022) propose an iterative procedure to straighten transport paths. At each iteration k, a vector field $v^{(k)}$ is trained via the CFM loss $\mathcal{L}_{CFM}$ on the current coupling $(X_0^{(k)}, X_1^{(k)})$ with deterministic interpolation with $\sigma = 0$, $\alpha_t = 1 - t$, $\beta_t = t$. The next coupling is generated by pushing forward the source through the learned field: $X_1^{(k+1)} = T_f(X_0) :=$ ODE-Solve$[v^{(k)}](X_0^{(k)}, t = 1)$. This iterative resampling progressively straightens paths; convergence is achieved when

$$\mathbb{E}\left[X_0^{(k)} - X_1^{(k)} \mid X_t^{(k)} = x\right] = X_0^{(k)} - X_1^{(k)}, \tag{4}$$

which characterizes straight transport.

Algorithm 1 (IMF, ReFlow). *Let p_0, p_1 be endpoint laws on $\mathbb{R}^n$. Initialize the coupling $\pi^{(0)} := p_0 \otimes p_1$. For $k = 0, 1, 2, \ldots$. Iteratively, we fit a pushforward map $T_f^{(k)} : \mathbb{R}^n \to \mathbb{R}^n$ given the paired data $(X_0, X_1) \sim \pi^{(k)}$. Define a new pair by*

$$(X_0^+, X_1^+) := \big(X_0,\ T_f^{(k)}(X_0)\big), \qquad X_0 \sim p_0,$$

and set $\pi^{(k+1)} := Law(X_0^+, X_1^+)$.

This algorithm is known to accumulate error through degradation of the target marginal Seong et al. (2025). Schrödinger Bridge Matching, which we introduce next, is conjectured to mitigate this error accumulation via its bidirectional structure.

2. Schrödinger Bridge Matching The Schrödinger Bridge seeks the entropic interpolation between π_0 and π_1:

$$\inf_{v^+,v^-:\mathbb{P}_0^{v^+,v^-}=\pi_0,\mathbb{P}_1^{v^+,v^-}=\pi_1} D_{KL}(\mathbb{P}^{v^+,v^-} \| \mathbb{P}^{\text{ref}}) \tag{5}$$

where $\mathbb{P}^{\text{ref}}$ is a reference process. DSBM Shi et al. (2024) learns bidirectional vector fields v_θ after Equation 3, with $I(X_0, X_1, t) = (1-t)X_0 + tX_1 + \sqrt{(1-t)t}\epsilon$. On couplings which are computed in the way it is described in Algorithm 2. Then new couplings can be obtained the $X_1^{(k+1)} = T_f(X_0) := \text{SDE-Solve}[v_f^{(k)}, \sqrt{t(1-t)\epsilon}](X_0^{(k)}, t=1)$ for forward pass, and $X_0^{(k+1)} = T_f(X_1) := \text{SDE-Solve}[v_b^{(k)}, \sqrt{t(1-t)\epsilon}](X_1^{(k)}, t=1)$. The training procedure is described in the following algorithm.

Algorithm 2 (toy-DSBM)**.** *Let p_0, p_1 be endpoint laws on $\mathbb{R}^n$. Initialize the coupling $\pi^{(0)} := p_0 \otimes p_1$. For $k = 0, 1, 2, \ldots$, perform the two substeps:*

1. ***Forward update.*** *Fit a map $T_f^{(k)} : \mathbb{R}^n \to \mathbb{R}^n$ from paired data $(X_0, X_1) \sim \pi^{(2k)}$. Define a new pair by*

$$(X_0^+, X_1^+) := \big(X_0,\ T_f^{(k)}(X_0)\big), \qquad X_0 \sim p_0,$$

and set $\pi^{(2k+1)} := Law(X_0^+, X_1^+)$.

2. ***Backward update.*** *Fit a map $T_b^{(k)} : \mathbb{R}^n \to \mathbb{R}^n$ from reversed paired data $(X_0, X_1) \sim \pi^{(2k+1)}$. Define a new pair by*

$$(X_0^{++}, X_1^{++}) := \big(T_b^{(k)}(X_1),\ X_1\big), \qquad X_1 \sim p_1,$$

and set $\pi^{(2k+2)} := Law(X_0^{++}, X_1^{++})$.

For completeness, we also consider a non-resampling variant of the above procedure, obtained by replacing the true marginals p_0, p_1 with the current couplings at generation time.

Algorithm 3 (no-resampling toy-DSBM)**.** *The no-resampling variant of Algorithm 2 is obtained by replacing, at each forward generation step, $X_0 \sim \pi^{(0)}$ with an independent fresh draw $\bar{X}_0 \sim \pi^{(2k)}$, and at each backward generation step, $X_1 \sim \pi^{(0)}$ with an independent fresh draw $\bar{X}_1 \sim \pi^{(2k+1)}$, while keeping the fitting steps unchanged.*

Measuring Marginal Displacement. By error accumulation, we mean the drift of marginal distributions across iterations. We quantify this using standard metrics: Kullback–Leibler divergence $D_{\mathrm{KL}}(\mu\|\nu)$, total variation distance $\mathrm{TV}(\mu, \nu)$, and Wasserstein-2 distance $W_2(\mu, \nu)$. In our main results on Gaussian marginals, we focus on tracking how the variance $\mathrm{Tr}(\Sigma_k)$ of the learned distributions evolves across iterations.

Note on the algorithms. The maps $T_f^{(k)}$, and $T_b^{(k)}$ in Algorithms 1–3 can be either deterministic (via ODE integration) or stochastic (via SDE integration). The common belief is that Algorithm 2 avoids error accumulation because it resamples from the true marginals p_0, p_1 at each iteration, rather than the dynamics type it uses. In this paper, we challenge this belief by constructing failure modes where even this resampling strategy fails.

Integrated vs. True Marginals. We distinguish between *true marginals* p_0, p_1 (fixed source/-target used for resampling in Algorithm 2) and *integrated marginals* $\hat{p}_0^{(k)}, \hat{p}_1^{(k)}$ (pushforward distributions from applying maps). Specifically, $\hat{p}_1^{(k)} := (T_f^{(k)})_{\#}\hat{p}_0^{(k-1)}$ for forward steps and $\hat{p}_0^{(k)} := (T_b^{(k)})_{\#}\hat{p}_1^{(k-1)}$ for backward steps, where $T_f^{(k)}, T_b^{(k)}$ are the maps fit at iteration k.

3 DISPLACEMENT IN THEORY

To better isolate marginal displacement mechanisms in Algorithm 2 (DSBM), we analyze perturbed optimal transport (OT) maps between Gaussians rather than learned stochastic vector fields. This choice is motivated by three considerations. First, the claimed robustness of DSBM is widely attributed to its bidirectional resampling structure rather than its stochastic (SDE) dynamics (Peluchetti, 2023; Gushchin et al., 2024; Kholkin et al., 2024), suggesting that the resampling mechanism, not the noise, is the key stabilizing factor. Second, since our focus is on marginal evolution rather than coupling quality (as in entropic OT), deterministic maps suffice to reveal how errors propagate through marginals alone. Third, exact OT maps between Gaussians are globally defined on all of $\mathbb{R}^d$ (Peyré et al., 2019), allowing us to precisely characterize off-support behavior when maps trained on perturbed distributions are applied to true marginals, a scenario that would be ambiguous with learned conditional expectations $\mathbb{E}[X_1 - X_0 \mid X_t]$ which may behave unpredictably outside their training support.

This controlled setting enables rigorous theoretical analysis (Propositions 2–1) before we relax these assumptions in Section 4.2(where we examine the practically relevant case of neural network regression from finite samples).

We study two diffeomorphic perturbations of exact Gaussian OT maps (Peyré et al., 2019). Given the optimal transport map $T_{\mathrm{OT}}^{(k)}$ at iteration k, we consider perturbed maps of the form $\tilde{T}^{(k)}(x) = \tilde{t}(T_{\mathrm{OT}}^{(k)}(x))$. Our first perturbation, **constant shift** (Proposition 1), adds a fixed bias: $\tilde{t}(z) = z - \epsilon \mathbf{1}_n$ forward and $\tilde{t}(z) = z + \epsilon \mathbf{1}_n$ backward, inducing systematic mean drift. Our second perturbation, **variance scaling** (Proposition 2), rescales the map: $\tilde{t}(z) = \sqrt{\epsilon_f} z$ forward ($\sqrt{\epsilon_b} z$ backward), inflating/contracting variances ($\epsilon_f, \epsilon_b > 0$). When $\tilde{t}$ is a diffeomorphism, Lemma 1 shows that bidirectional ReFlow (Algorithm 1) and non-resampling DSBM (Algorithm 3) produce identical marginals, enabling unified analysis.

Lemma 1. *Suppose the perturbed maps $\tilde{T}_k$ (forward) and $\tilde{S}_k$ (backward) are diffeomorphisms for all k. When both algorithms use the same error perturbations, Algorithm 1 (Rectified Flow applied bidirectionally) and Algorithm 3 (no-resampling DSBM) produce identical integrated marginals $\tilde{p}_0^{(k)}, \tilde{p}_1^{(k)}$ for all $k \geq 0$.*

This result (Lemma 1) simplifies subsequent analysis: we use it to streamline proofs in Section 3 and experiments in Section 4. Importantly, Algorithm 3 has limited practical utility for generative modeling. Without resampling from true marginals p_0, p_1, one cannot generate samples at intermediate iterations, only after completing the full bidirectional cycle. However, the equivalence is valuable theoretically: it shows that non-resampling DSBM reduces to bidirectional Rectified Flow, allowing us to analyze both methods simultaneously.

We will now present an example where accumulating errors will result in the total forgetting of the source and target, even in the case of the toy-DSBM Algorithm 2.

Proposition 1 (Constant shift failure mode)**.** *Let $p_0 = \mathcal{N}(0, I_n)$ and $p_1 = \mathcal{N}(\mu, I_n)$ where $\mu \in \mathbb{R}^n$. Fix $\epsilon > 0$. At each iteration, given a pairing $(X_0^{(k)}, X_1^{(k)})$ with Gaussian marginals, compute the quadratic-cost OT maps $T_{OT}^{(k)}, S_{OT}^{(k)}$ between them and define the erroneous maps via constant shift:*

$$\tilde{T}_k(x) = T_{OT}^{(k)}(x) - \epsilon \mathbf{1}_n, \quad \tilde{S}_k(y) = S_{OT}^{(k)}(y) + \epsilon \mathbf{1}_n$$

where $\mathbf{1}_n = (1, 1, \ldots, 1)^\top \in \mathbb{R}^n$. All three algorithms exhibit linear drift in marginal means, however DSBM (Algorithm 2, drifts twice faster than Algorithm 1, and 3).

Proposition 2 (Iterative OT with additive Gaussian noise)**.** *Let $p_0 = \mathcal{N}(0, I_n)$ and $p_1 = \mathcal{N}(0, I_n)$. Fix $\epsilon_f, \epsilon_b \geq 0$. At each iteration, given a pairing (X_0, X_1) with Gaussian marginals, compute the quadratic-cost OT maps between them and define the erroneous maps*

$$\tilde{T}_f^{(k)}(x) := \sqrt{\epsilon_f} T_{\mathrm{OT}}^{(k)}(x), \qquad \tilde{T}_b^{(k)}(y) := \sqrt{\epsilon_b} S_{\mathrm{OT}}^{(k)}(y),$$

For $\epsilon_f, \epsilon_b > 1$ Algorithms 1, and 3 exhibit unbounded error accumulation, while Algorithm 2 stabilizes at a finite discrepancy, and for $0 < \epsilon_f, \epsilon_b < 1$, all Algorithms exhibit unbounded error accumulation.

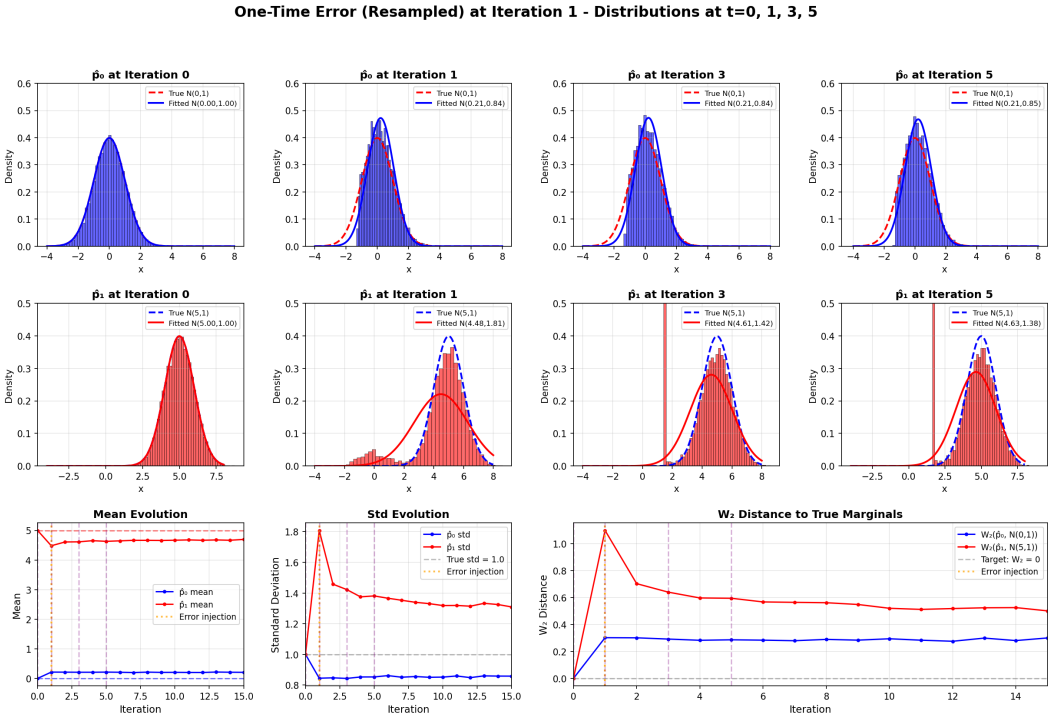


Figure 1: **One-time contamination in Algorithm 2.** At iteration 1, the forward map learns to a contaminated target $0.1\,\mathcal{N}(0,1) + 0.9\,\mathcal{N}(5,1)$ instead of $\mathcal{N}(5,1)$. Despite resampling from true p_0, p_1, the integrated marginals $\hat{p}_0, \hat{p}_1$ perpetuate the displacement. By iteration 5, $\hat{p}_1$ remains bimodal and $\hat{p}_0$ exhibits asymmetric distortion (apparent discontinuity in left tail), showing that corruption of $\hat{p}_1$ destroys $\hat{p}_0$ as well.

These two theoretical examples demonstrate that Algorithm 2 fails to alleviate accumulating error under systematic perturbations—linear drift from constant shifts (Prop. 1) and variance divergence from scaling (Prop. 2), with proofs in Appendix A .

4 Displacement in Practice

We present two classes of experiments. In Section 4.1, we work with exact optimal transport maps defined on the entire space, allowing us to precisely characterize how contaminated maps affect off-support regions. In Section 4.2, we learn transport maps via regression $\arg\min_\theta \|T_\theta(X_0) - X_1\|^2$ from finite OT couplings (X_0, X_1), mimicking practical flow matching scenarios.

Notation for Section 4. We use $\hat{p}_0^{(k)}$ and $\hat{p}_1^{(k)}$ to denote the *integrated marginals* at iteration k—that is, the pushforward distributions obtained by applying the learned (potentially erroneous) maps to samples. Specifically, $\hat{p}_1^{(k)} := T_f^{(k)} \# \hat{p}_0^{(k-1)}$ for forward steps and $\hat{p}_0^{(k)} := T_b^{(k)} \# \hat{p}_1^{(k-1)}$ for backward steps. These differ from the true marginals p_0, p_1 used for resampling in Algorithm 2.

4.1 OT-induced maps

When a map T_f is learned on $(\hat{p}_0, p_1)$ and applied to $\hat{p}_0$, its behavior is generally unknown off-support. Using exact OT maps—defined on all of $\mathbb{R}^d$—allows us to compute $T_\# \hat{p}_0$ analytically and precisely track marginal evolution.

One-time forward displacement propagation. We study how a single marginal contamination propagates through Algorithms 2 and 3. We set $p_0 = \mathcal{N}(0, I)$ and $p_1 = \mathcal{N}(5, I)$. At iteration 1 only, the forward map learns to a contaminated target $0.1\,p_0 + 0.9\,p_1 = 0.1\,\mathcal{N}(0,1) + 0.9\,\mathcal{N}(5,1)$ instead

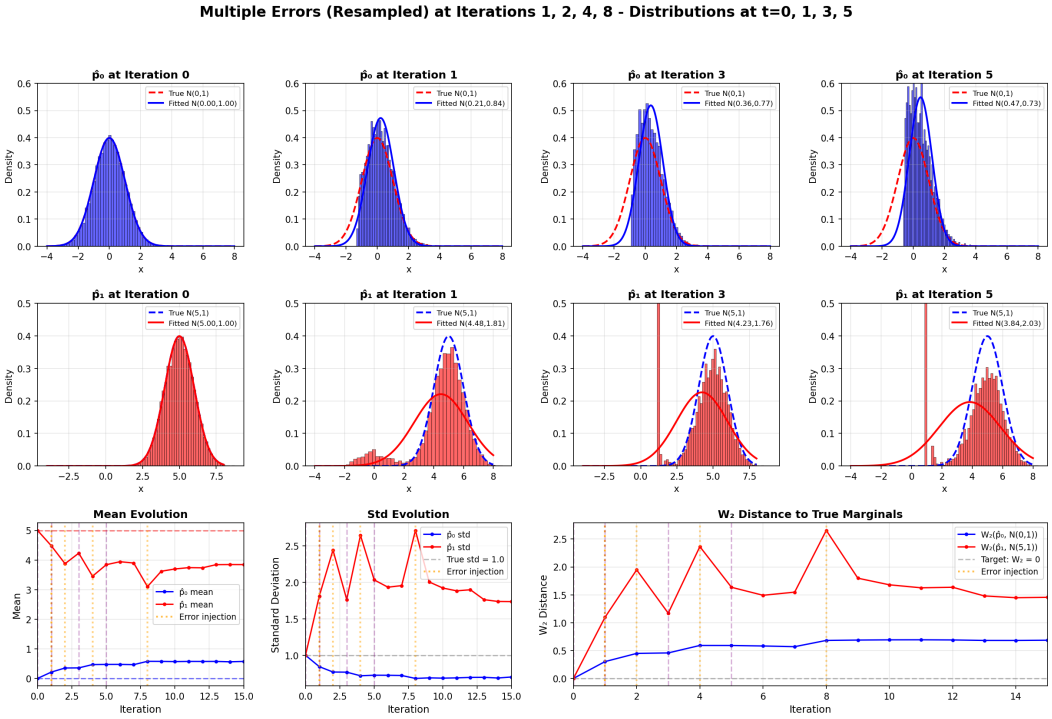
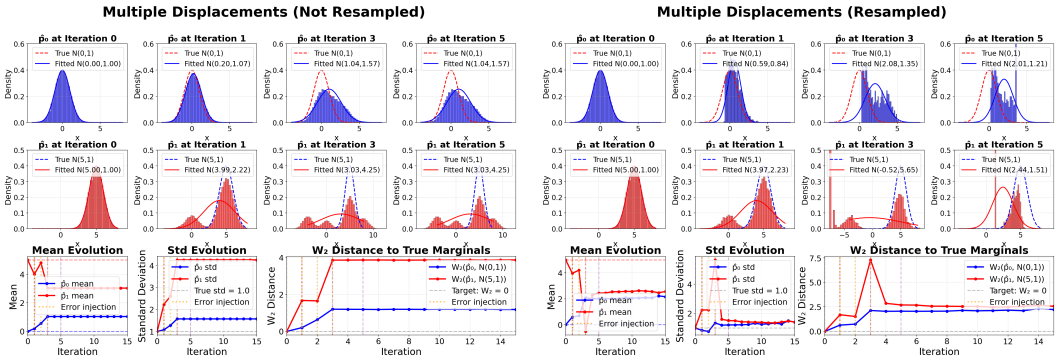


Figure 2: **Repeated contamination in Algorithm 2.** The forward map learns to $\frac{1}{10}\hat{p}_0 + \frac{9}{10}p_1$ at iterations 1, 2, 4, and 8 . Each error compounds existing corruption: $\hat{p}_1$ progressively degrades from mean 4.49 (iteration 1) to 3.66 (iteration 9) with increasing variance, while $\hat{p}_0$ drifts from 0.21 to 0.55. Unlike the one-time case (Figure 1), multiple contaminations cause cumulative degradation that resampling cannot prevent.

of the true $p_1 = \mathcal{N}(5, 1)$. In 1D, we compute the exact quadratic-cost OT maps for both forward and backward steps. Figure 1 shows that despite resampling from the true marginals p_0 and p_1, the integrated marginals $\hat{p}_0$ and $\hat{p}_1$ both accumulate displacement from the initial contamination. $\hat{p}_1$ develops persistent bimodality, and by iteration 5, $\hat{p}_0$ exhibits asymmetric distortion with apparent loss of smoothness on the left tail, demonstrating that corrupting $\hat{p}_1$ inevitably corrupts $\hat{p}_0$. In contrast, Algorithm 3 preserves $\hat{p}_0 = p_0$ while $\hat{p}_1$ remains unchanged in later iterations (see Appendix B, Figure 7).

Multiple forward displacement propagation. Having demonstrated that a single contamination at iteration 1 causes persistent displacement, we now examine cumulative effects when contamination occurs repeatedly. We use the same setup: $p_0 = \mathcal{N}(0, I)$ and $p_1 = \mathcal{N}(5, I)$. At iterations 1, 2, 4, and 8, we introduce contamination in the forward step: instead of learning $\hat{p}_0 \to p_1$, we learn $\hat{p}_0 \to 1/10\hat{p}_0 + 9/10p_1$. All other iterations follow the standard Algorithm 2: forward steps learn from the corrupted $\hat{p}_0^{(k)}$ to p_1 and apply to fresh p_0 samples, while backward steps learn from $\hat{p}_1^{(k)}$ to p_0 and apply to fresh p_1 samples. Figure 2 shows that repeated contamination amplifies marginal displacement. Both $\hat{p}_0$ and $\hat{p}_1$ exhibit progressive degradation: by iteration 9, $\hat{p}_1$ has mean 3.66 and variance 2.02 (versus true mean 5 and variance 1), while $\hat{p}_0$ drifts to mean 0.55 with compressed variance 0.66. Unlike the one-time case where displacement partially stabilizes between errors, each new contamination compounds the existing corruption, demonstrating that resampling alone cannot mitigate accumulated systematic failures.

Forward and backward displacement propagation In this experiment, we demonstrate that Algorithm 2 can be subject to drastic changes in the integrated marginals $\hat{p}_0$ and $\hat{p}_1$. We investigate the compounding effect when errors occur in both the forward and backward steps simultaneously. We use the same setup: $p_0 = \mathcal{N}(0, 1)$ and $p_1 = \mathcal{N}(5, 1)$, with Algorithm 2 (DSBM) and Algorithm 3. At iterations $k \in \{1, 2, 3\}$, we inject displacement errors in both directions. In the forward step,



(a) Not resampled (b) DSBM (Resampled)

Figure 3: **Bidirectional displacement propagation in Algorithms 2 and 3.** At iterations 1, 2, and 3, both forward and backward maps displace targets. We track marginal evolution in subsequent iterations. While both methods show similar W_2 error (on average from their true distributions), Algorithm 2 exhibits qualitative degradation: $\hat{p}_0$, and $\hat{p}_1$ transforms from a smooth Gaussian to a distribution with discrete components and discontinuous support, revealing that resampling prevents metric divergence but not structural collapse.

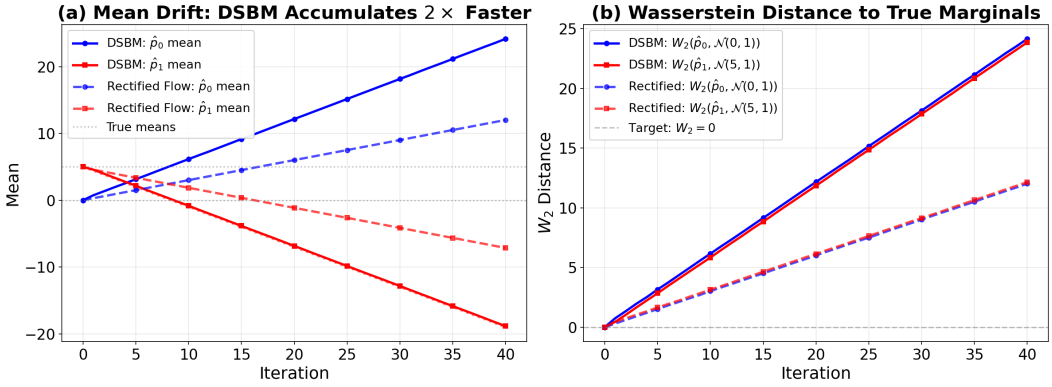


Figure 4: **Constant shift failure mode.** DSBM (solid) vs Rectified Flow (dashed) with shift $\epsilon = 0.3$. DSBM accumulates error at rate $2\epsilon k$ while Rectified Flow drifts at rate ϵk.

instead of learning $\hat{p}_0 \to p_1$, we learn $\hat{p}_0 \to \frac{1}{5}\mathcal{N}(e_f^{(k)}, 1) + \frac{4}{5}p_1$, where the displacement centers are $e_f^{(1)} = 0$, $e_f^{(2)} = 8$, and $e_f^{(3)} = -4$. Simultaneously, in the backward step, we learn $\hat{p}_1 \to \frac{1}{5}\mathcal{N}(e_b^{(k)}, 1) + \frac{4}{5}p_0$ with $e_b^{(1)} = 1$, $e_b^{(2)} = 2$, and $e_b^{(3)} = 3$. This bidirectional perturbation creates a realistic failure scenario where both transport directions are corrupted. Figure 3 demonstrates that despite resampling from the true marginals p_0 and p_1 after each step, bidirectional errors cause severe marginal degradation.

Shift example Given the setting of Proposition 1, we empirically compare Algorithm 2 (DSBM) and Algorithm 3 under constant shift perturbations $\epsilon = 0.3$. With $p_0 = \mathcal{N}(0, 1)$ and $p_1 = \mathcal{N}(5, 1)$, we apply shifts $\tilde{T}(x) = T_{\text{OT}}(x) - \epsilon$ and $\tilde{S}(y) = S_{\text{OT}}(y) + \epsilon$ to the forward and backward maps respectively. Figure 4 reveals that DSBM accumulates error twice as fast as Rectified Flow: after 40 iterations, DSBM reaches mean displacements of ± 24 versus ± 12 for Rectified Flow, matching the theoretical predictions of $2\epsilon k$ and ϵk respectively. This acceleration occurs because DSBM's bidirectional updates cause errors from forward and backward steps to compound additively in each iteration. Despite resampling from true marginals p_0 and p_1 at each step, the systematic nature of the shift prevents cancellation. The Wasserstein distances in panel (b) confirm both marginals drift linearly, with DSBM degrading at double the rate. This demonstrates that bidirectional resampling amplifies rather than mitigates error accumulation under systematic perturbations.

4.2 LEARNED MAPS

In the previous section, we analyzed exact optimal transport maps defined on all of $\mathbb{R}^d$, allowing precise characterization of error propagation even in regions with no data support. We now relax this idealized setting and examine the practically relevant case where transport maps are learned via neural network regression. Specifically, we train a network to approximate the conditional expectation $\mathbb{E}[X_1 - X_0 \mid X_t]$ from finite samples, as is standard in flow matching. This introduces an additional source of error: the learned map may behave unpredictably when applied to points outside the training data support.

Shift example We extend the shift experiment to 2D ($p_0 = \mathcal{N}([0,0], I_2)$, $p_1 = \mathcal{N}([5,5], I_2)$, $\epsilon = 0.3$), replacing exact OT maps with neural network approximations as in standard flow matching. At each iteration, we train a feedforward network (2 layers, 64 units, ReLU) to regress the velocity field from paired samples, then apply the learned map (with shift $\pm\epsilon$) to fresh samples from true marginals.

Figure 5 compares learned (left) versus exact (right) maps at iterations 0, 5, and 10. Both exhibit comparable mean drift rates matching theoretical predictions. However, learned maps introduce severe distribution distortion: while exact maps preserve Gaussianity, neural networks produce elongated, non-Gaussian marginals by iteration 10. This occurs because networks trained on samples near current means extrapolate poorly when applied to increasingly distant true marginal samples. Practical implementations thus face dual error sources: systematic marginal drift from theory, plus shape distortion from off-support approximation errors.

5 CONCLUSION

We studied a controlled toy version of DSBM to test whether bidirectional resampling alone explains robustness to error accumulation. Through two tractable failure modes, we showed that DSBM with resampling (Algorithm 2) does not universally prevent error accumulation: under constant shift perturbations, it exhibits linear drift (Proposition 1), and under variance-scaling perturbations its behavior depends on the perturbation structure rather than on resampling alone (Proposition 2).

We further showed that the no-resampling variant (Algorithm 3) exhibits marginal decoupling (Lemma 1): forward steps preserve $\hat{p}_0$ while corrupting $\hat{p}_1$, demonstrating that marginal preservation does not imply transport map accuracy. Together, these results show that resampling alone is not the mechanism behind robustness to non-additive error, and can even accelerate error accumulation depending on the perturbation.

Our experiments with exact optimal transport maps and learned neural networks support the same qualitative picture. They confirm the predicted drift behavior and show that learned maps can introduce additional geometric distortion, reinforcing that the phenomenon is not eliminated by resampling alone.

Future work should investigate which aspects of these toy failure modes persist in full learned DSBM systems, high-dimensional settings, and realistic generative modeling pipelines. It would also be valuable to develop mitigation strategies, such as error detection, adaptive stopping, or hybrid updates, that reduce systematic error accumulation without losing the computational benefits of iterative refinement.

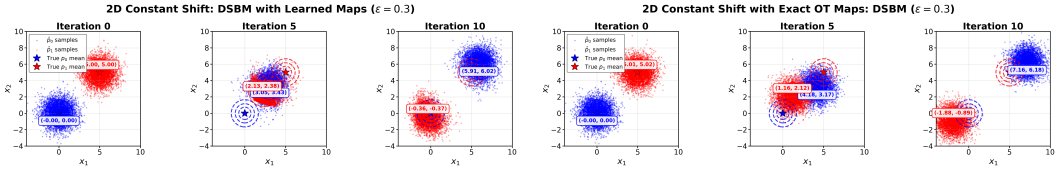


Figure 5: **Learned maps amplify distribution distortion.** Neural approximations (left) vs exact OT maps (right). Both show predicted mean drift, but learned maps produce non-Gaussian marginals.

REFERENCES

Vansh Bansal, Saptarshi Roy, Purnamrita Sarkar, and Alessandro Rinaldo. Straightness of rectified flow: A theoretical insight into wasserstein convergence. *arXiv preprint arXiv:2410.14949*, 2024.

Yann Brenier. Polar factorization and monotone rearrangement of vector-valued functions. *Communications on pure and applied mathematics*, 44(4):375–417, 1991.

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

Valentin De Bortoli, Iryna Korshunova, Andriy Mnih, and Arnaud Doucet. Schrodinger bridge flow for unpaired data translation. *Advances in Neural Information Processing Systems*, 37: 103384–103441, 2024.

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

Leonid Kantorovitch. On the translocation of masses. *Management science*, 5(1):1–4, 1958.

Sergei Kholkin, Grigoriy Ksenofontov, David Li, Nikita Kornilov, Nikita Gushchin, Alexandra Suvorikova, Alexey Kroshnin, Evgeny Burnaev, and Alexander Korotin. Diffusion & adversarial schr\" odinger bridges via iterative proportional markovian fitting. *arXiv preprint arXiv:2410.02601*, 2024.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Qiang Liu. Rectified flow: A marginal preserving approach to optimal transport. *arXiv preprint arXiv:2209.14577*, 2022.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2024.

Robert J McCann. A convexity principle for interacting gases. *Advances in mathematics*, 128(1): 153–179, 1997.

Gaspard Monge. Mémoire sur la théorie des déblais et des remblais. *Mem. Math. Phys. Acad. Royale Sci.*, pp. 666–704, 1781.

Stefano Peluchetti. Diffusion bridge mixture transports, schrödinger bridge problems and generative modeling. *Journal of Machine Learning Research*, 24(374):1–51, 2023.

Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019.

Kim Shin Seong, Mingi Kwon, Jaeseok Jeong, and Youngjung Uh. Balanced conic rectified flow. *arXiv preprint arXiv:2510.25229*, 2025.

Yuyang Shi, Valentin De Bortoli, Andrew Campbell, and Arnaud Doucet. Diffusion schrödinger bridge matching. *Advances in Neural Information Processing Systems*, 36, 2024.

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

Huminhao Zhu, Fangyikang Wang, Tianyu Ding, Qing Qu, and Zhihui Zhu. Analyzing and mitigating model collapse in rectified flow models. *arXiv preprint arXiv:2412.08175*, 2024.

A HELPFUL RESULTS

You may include other additional sections here.

Lemma 2. *Consider the non-resampling variant (Algorithm 1) with a fixed diffeomorphic perturbed map $\tilde{T}(\epsilon, x)$ applied at each forward iteration. At iteration k, the distribution $p_1^{(k)}$ can be expressed as the pushforward of p_0 under the k-fold composition:*

$$p_1^{(k)} = (\tilde{T}^{\circ k})\#p_0,$$

where $\tilde{T}^{\circ k} := \underbrace{\tilde{T} \circ \tilde{T} \circ \cdots \circ \tilde{T}}_{k \text{ times}}$.

Proof. We proceed by induction on k.

Base case ($k = 0$): By definition, $p_1^{(0)} = p_1$ and $\tilde{T}^{\circ 0} = \text{id}$, so the identity map gives $p_1^{(0)} = \text{id}\#p_0 = p_0$. (Assuming $p_0 = p_1$ initially, or we can start from $k = 1$.)

Inductive step: Assume that $p_1^{(k)} = (\tilde{T}^{\circ k})\#p_0$.

In the non-resampling algorithm, at iteration $k + 1$ we sample $X_1^{(k)} \sim p_1^{(k)}$ and apply the perturbed map:

$$X_1^{(k+1)} = \tilde{T}(X_0^{(k)}).$$

By the inductive hypothesis, $X_1^{(k)} = \tilde{T}^{\circ k}(X_0)$ where $X_0 \sim p_0$. Therefore:

$$X_1^{(k+1)} = \tilde{T}(\tilde{T}^{\circ k}(X_0)) = \tilde{T}^{\circ(k+1)}(X_0),$$

which gives $p_1^{(k+1)} = (\tilde{T}^{\circ(k+1)})\#p_0$.

By induction, the result holds for all $k \geq 0$. □

Lemma 1. *Suppose the perturbed maps $\tilde{T}_k$ (forward) and $\tilde{S}_k$ (backward) are diffeomorphisms for all k. When both algorithms use the same error perturbations, Algorithm 1 (Rectified Flow applied bidirectionally) and Algorithm 3 (no-resampling DSBM) produce identical integrated marginals $\tilde{p}_0^{(k)}, \tilde{p}_1^{(k)}$ for all $k \geq 0$.*

Proof. Decompose the perturbed map as $\tilde{T}(x, \epsilon) = \tilde{t}(T_{\text{OT}}(x), \epsilon)$, where T_{OT} is the optimal transport map from $p_0^{(k)}$ to $p_1^{(k)}$ and $\tilde{t}(\cdot, \epsilon)$ is the error function applied in the target space. Similarly for the backward map: $\tilde{S}(y, \epsilon) = \tilde{s}(S_{\text{OT}}(y), \epsilon)$.

We prove by induction that the marginal sequences coincide at every iteration.

Base case ($k = 0$): Both algorithms initialize with $\pi^{(0)} = p_0 \otimes p_1$, so $\tilde{p}_0^{(0,\text{Reflow})} = \tilde{p}_0^{(0,\text{NoResampling})} = p_0$ and $\tilde{p}_1^{(0,\text{Reflow})} = \tilde{p}_1^{(0,\text{NoResampling})} = p_1$.

Inductive step: Assume $\tilde{p}_0^{(k,\text{Reflow})} = \tilde{p}_0^{(k,\text{NoResampling})}$ and $\tilde{p}_1^{(k,\text{Reflow})} = \tilde{p}_1^{(k,\text{NoResampling})}$ for some $k \geq 0$.

Forward step (computing $\tilde{p}_1^{(k+1)}$):

Algorithm 1: Fits a map on $(p_0, \tilde{p}_1^{(k)})$, perturbs it, and applies to fresh p_0 samples. By Lemma 2, this yields $\tilde{p}_1^{(k+1)} = (\tilde{T}^{\circ(k+1)})\#p_0$.

Algorithm 3: The current coupling has marginals $(\tilde{p}_0^{(k)}, \tilde{p}_1^{(k)})$. Let $T_{\text{OT}}^{(k)} : \tilde{p}_0^{(k)} \to \tilde{p}_1^{(k)}$ be the optimal transport map between these marginals (which exists by Brenier's theorem (Brenier, 1991) since diffeomorphisms preserve absolute continuity). The perturbed map is:

$$\tilde{T}_k(x) = \tilde{t}(T_{\text{OT}}^{(k)}(x), \epsilon).$$

Sampling $X_0 \sim \tilde{p}_0^{(k)}$ and applying $\tilde{T}_k$ gives:

$$\tilde{p}_1^{(k+1)} = \tilde{T}_k \# \tilde{p}_0^{(k)} = \tilde{t}(\cdot, \epsilon) \# (T_{\text{OT}}^{(k)} \# \tilde{p}_0^{(k)}) = \tilde{t}(\cdot, \epsilon) \# \tilde{p}_1^{(k)}.$$

Since $\tilde{p}_1^{(k,\text{Reflow})} = \tilde{p}_1^{(k,\text{NoResampling})}$ by the inductive hypothesis, and both algorithms apply the same error function $\tilde{t}(\cdot, \epsilon)$ to this distribution, we have $\tilde{p}_1^{(k+1,\text{Reflow})} = \tilde{p}_1^{(k+1,\text{NoResampling})}$.

Backward step: By the same argument with $\tilde{s}(\cdot, \epsilon)$ applied to $\tilde{p}_0^{(k+1)}$, we obtain $\tilde{p}_0^{(k+1,\text{Reflow})} = \tilde{p}_0^{(k+1,\text{NoResampling})}$.

By induction, the marginal sequences coincide for all $k \geq 0$. □

Lemma 3 (OT trained on a variance-inflated source)**.** *Let $\alpha \geq 0$, and define $\tilde{p} := \mathcal{N}(0, \alpha I_n)$ and $p := \mathcal{N}(0, I_n)$. Let $T := T_{\tilde{p} \to p}$ be the quadratic-cost optimal transport map from $\tilde{p}$ to p. Then $T(x) = \frac{1}{\sqrt{\alpha}} x$, and consequently*

$$T_\sharp \mathcal{N}(0, I_n) = \mathcal{N}\Big(0, \frac{1}{\alpha} I_n\Big).$$

Proof. It is a standard fact that the quadratic-cost OT map between Gaussian measures is affine. Since both $\tilde{p}$ and p are centered and isotropic, the OT map must be

$$T(x) = \frac{1}{\sqrt{\alpha}} x.$$

Let $Z \sim \mathcal{N}(0, I_n)$ and set $W := T(Z) = \frac{1}{\sqrt{\alpha}} Z$. Then $W \sim \mathcal{N}\big(0, \frac{1}{\alpha} I_n\big)$, i.e.

$$T_\sharp \mathcal{N}(0, I_n) = \mathcal{N}\Big(0, \frac{1}{\alpha} I_n\Big).$$

□

Proposition 1 (Constant shift failure mode)**.** *Let $p_0 = \mathcal{N}(0, I_n)$ and $p_1 = \mathcal{N}(\mu, I_n)$ where $\mu \in \mathbb{R}^n$. Fix $\epsilon > 0$. At each iteration, given a pairing $(X_0^{(k)}, X_1^{(k)})$ with Gaussian marginals, compute the quadratic-cost OT maps $T_{OT}^{(k)}, S_{OT}^{(k)}$ between them and define the erroneous maps via constant shift:*

$$\tilde{T}_k(x) = T_{OT}^{(k)}(x) - \epsilon \mathbf{1}_n, \quad \tilde{S}_k(y) = S_{OT}^{(k)}(y) + \epsilon \mathbf{1}_n$$

where $\mathbf{1}_n = (1, 1, \ldots, 1)^\top \in \mathbb{R}^n$. All three algorithms exhibit linear drift in marginal means.

Proof. **No-Resample.** Start with $X_0^{(0)} \sim \mathcal{N}(0, I_n)$, $X_1^{(0)} \sim \mathcal{N}(\mu, I_n)$. The OT map is $T_{\text{OT}}(x) = x + \mu$.

Iteration 1: Forward gives with error gives $X_1^{(1)} = X_0^{(0)} + \mu - \epsilon \mathbf{1}_n \sim \mathcal{N}(\mu - \epsilon \mathbf{1}_n, I_n)$. So the pairings on which the backward learns are $(X_0^{(0)} + \mu - \epsilon, X_0^{(0)})$, so $S_{\text{OT}}(y) = y - (\mu - \epsilon \mathbf{1}_n)$, we get $X_0^{(1)} = X_1^{(1)} - (\mu - \epsilon \mathbf{1}_n) + \epsilon \mathbf{1}_n = X_0^{(0)} + \mu - \epsilon \mathbf{1}_n - \mu + 2\epsilon \mathbf{1}_n = X_0^{(0)} + \epsilon \mathbf{1}_n$, so $X_0^{(1)} \sim \mathcal{N}(+\epsilon \mathbf{1}_n, I_n)$. We can see the effect of Lemma 1 in here).

The proof by induction is just applying the step above on the inductive assumption, that $X_1^{(k)} \sim \mathcal{N}(\mu - k\epsilon, I_n)$, and $X_0^{(k)} \sim \mathcal{N}(k\epsilon, I_n)$. By Lemma 1, Alg. 1 matches.

DSBM. At each step, resample from true marginals before applying maps.

Iteration 1: Forward gives with error gives $X_1^{(1)} = X_0^{(0)} + \mu - \epsilon \mathbf{1}_n \sim \mathcal{N}(\mu - \epsilon \mathbf{1}_n, I_n)$. So the pairings on which the backward learns are $(X_0^{(0)} + \mu - \epsilon, X_0^{(0)})$, so $S_{\text{OT}}(y) = y - (\mu - \epsilon \mathbf{1}_n)$, but we apply this map to $X_1 \sim N(\mu, I_n)$, instead. We get $X_0^{(1)} = X_1 - (\mu - \epsilon \mathbf{1}_n) + \epsilon \mathbf{1}_n = X_1 - \mu + 2\epsilon \mathbf{1}_n \sim \mathcal{N}(2\epsilon \mathbf{1}_n, I_n)$. We can see the effect of Lemma 1 in here).

Both marginals drift away from their targets. After k iterations: $\bar{X}_0^{(k)} \sim \mathcal{N}((2k-1)\epsilon \mathbf{1}_n, I_n)$, $\bar{X}_1^{(k)} \sim \mathcal{N}(\mu - 2k\epsilon \mathbf{1}_n, I_n)$. The toy-DSBM version drifts at rate 2ϵ because both forward and backward errors accumulate independently. □

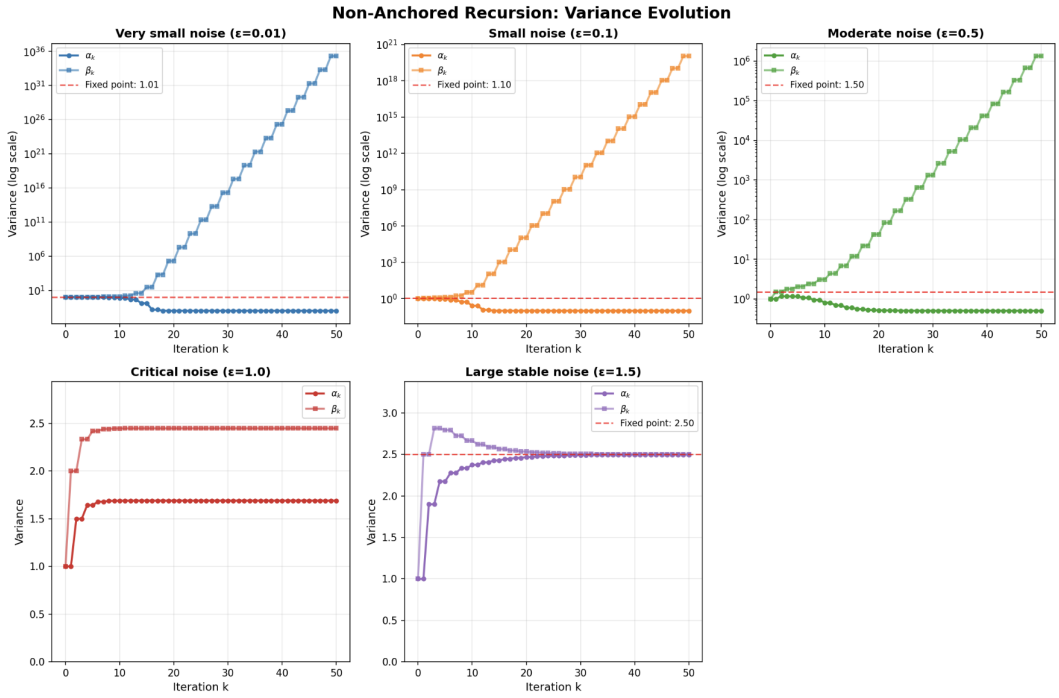


Figure 6: For $\epsilon_b = \epsilon_f < 1$, it looks like the sequence diverge.

Proposition 2 (Iterative OT with additive Gaussian noise)**.** *Let $p_0 = \mathcal{N}(0, I_n)$ and $p_1 = \mathcal{N}(0, I_n)$. Fix $\epsilon_f, \epsilon_b \geq 0$. At each iteration, given a pairing (X_0, X_1) with Gaussian marginals, compute the quadratic-cost OT maps between them and define the erroneous maps*

$$T_f(x) := T_{\mathrm{OT}}(x) + \eta_f, \qquad T_b(y) := S_{\mathrm{OT}}(y) + \eta_b,$$

where $\eta_f \sim \mathcal{N}(0, \epsilon_f I_n)$ and $\eta_b \sim \mathcal{N}(0, \epsilon_b I_n)$ are independent.

Algorithm 3 (non-resampled) exhibits unbounded error accumulation, while Algorithm 2 (resampled) stabilizes at a finite discrepancy.

Proof. **Toy-DSBM Version** At iteration 0 we start with $\pi^{(0)} = p_0 \otimes p_1$ (independent). Since $p_0 = p_1 = \mathcal{N}(0, I)$, the OT map is $T_{\mathrm{OT}}(x) = x$. For the forward, we sample fresh $\bar{X}_0 \sim p_0$ and apply $\bar{X}_1^{(1)} = \bar{X}_0 + \eta_f$ with $\eta_f \sim \mathcal{N}(0, \epsilon_f I)$. Thus $\bar{X}_1^{(1)} \sim \mathcal{N}(0, (1+\epsilon_f)I)$. The OT map from $\mathcal{N}(0, (1+\epsilon_f)I) \to \mathcal{N}(0, I)$ is $S_{\mathrm{OT}}(y) = \frac{1}{\sqrt{1+\epsilon_f}} y$ (from Lemma 3). Start fresh $\bar{X}_1 \sim p_1$ and apply $\bar{X}_0^{(2)} = \frac{1}{\sqrt{1+\epsilon_f}} \bar{X}_1 + \eta_b$. Thus

$$\bar{X}_0^{(2)} \sim \mathcal{N}\left(0, \alpha_2 I\right), \quad \alpha_2 := \frac{1}{1+\epsilon_f} + \epsilon_b.$$

At iteration 1, the OT map from $\mathcal{N}(0, \alpha_2 I) \to \mathcal{N}(0, I)$ is $T_{\mathrm{OT}}(x) = \frac{1}{\sqrt{\alpha_2}} x$. For the forward, we sample $\bar{X}_0 \sim p_0$ (not from $\alpha_2 I$!) and apply $\bar{X}_1^{(3)} = \frac{1}{\sqrt{\alpha_2}} \bar{X}_0 + \eta_f$. Thus $\bar{X}_1^{(3)} \sim \mathcal{N}(0, (\frac{1}{\alpha_2} + \epsilon_f)I)$. Similarly, $\bar{X}_0^{(4)} \sim \mathcal{N}(0, \alpha_4 I)$ with $\alpha_4 = \frac{1}{\frac{1}{\alpha_2}+\epsilon_f} + \epsilon_b$.

Fixed Point. The system converges when $\alpha^* = \frac{1}{\frac{1}{\alpha^*}+\epsilon_f} + \epsilon_b$. This quadratic has a finite solution α^*, so $|\alpha^* - 1| < \infty$—no unbounded accumulation. The positive solution of this system is $\alpha^* = \frac{\epsilon_b}{2} + \frac{\sqrt{\epsilon_f^2 \epsilon_b^2 + 4\epsilon_f \epsilon_b}}{2e_f}$, which means that the forward error will be $\alpha^* - 1 = \frac{\epsilon_b}{2} + \frac{\sqrt{\epsilon_f^2 \epsilon_b^2 + 4\epsilon_f \epsilon_b}}{2e_f} - 1 \approx \sqrt{\frac{e_b}{e_f}} + \frac{e_b}{2} - 1$.

Non-Resampling Version At iteration 0 we start with $\pi^{(0)} = p_0 \otimes p_1$ (independent). Since $p_0 = p_1 = \mathcal{N}(0, I)$, the OT map is $T_{\mathrm{OT}}(x) = x$. For the forward, we sample fresh $\bar{X}_0 \sim p_0$ and apply $\bar{X}_1^{(1)} = \bar{X}_0 + \eta_f$ with $\eta_f \sim \mathcal{N}(0, \epsilon_f I)$. Thus $\bar{X}_1^{(1)} \sim \mathcal{N}(0, (1+\epsilon_f)I)$. The OT map from $\mathcal{N}(0, (1+\epsilon_f)I) \to \mathcal{N}(0, I)$ is $S_{\mathrm{OT}}(y) = \frac{1}{\sqrt{1+\epsilon_f}} y$. Take $X_1 \sim \mathcal{N}(0, (1+\epsilon_f)I)$, and we will arrive at $X_0^{(2)} = \frac{X_1}{\sqrt{1+\epsilon_f}} + \eta_b$. We conclude that now we have $X_0^{(2)} \sim N(0, (1+\epsilon_b)I)$, and $X_1^{(2)} \sim N(0, (1+\epsilon_f)I)$.

At iteration 1 we start from $X_0 \sim \mathcal{N}(0, (1+\epsilon_b)I)$ and $X_1 \sim \mathcal{N}(0, (1+\epsilon_f)I)$. The OT map is $T_{\mathrm{OT}}(x) = \sqrt{\frac{1+\epsilon_f}{1+\epsilon_b}} x$. Applying the noisy forward map gives $X_1^{(3)} = \sqrt{\frac{1+\epsilon_f}{1+\epsilon_b}} X_0 + \eta_f$, so

$$X_1^{(3)} \sim \mathcal{N}\left(0, \left(\frac{1+\epsilon_f}{1+\epsilon_b} + \epsilon_f\right) I\right).$$

Similarly, the backward step yields $X_0^{(4)} \sim \mathcal{N}\left(0, \left(\frac{1+\epsilon_b}{(1+\epsilon_f)/(1+\epsilon_b)+\epsilon_f} + \epsilon_b\right) I\right)$. Both variances grow unboundedly as iterations proceed, demonstrating error accumulation.

The recursive formulas for the non-resampling version are:

Let $\alpha_k := \mathrm{Var}[X_0^{(k)}]$ and $\beta_k := \mathrm{Var}[X_1^{(k)}]$ denote the variances at iteration k.

Forward step (odd k):

$$\beta_{k+1} = \frac{\beta_k}{\alpha_k} + \epsilon_f$$

Backward step (even k):

$$\alpha_{k+1} = \frac{\alpha_k}{\beta_k} + \epsilon_b$$

Initial conditions:

$$\alpha_0 = \beta_0 = 1$$

From Figure 6 we can see that divergence takes place for small error initial values.

□

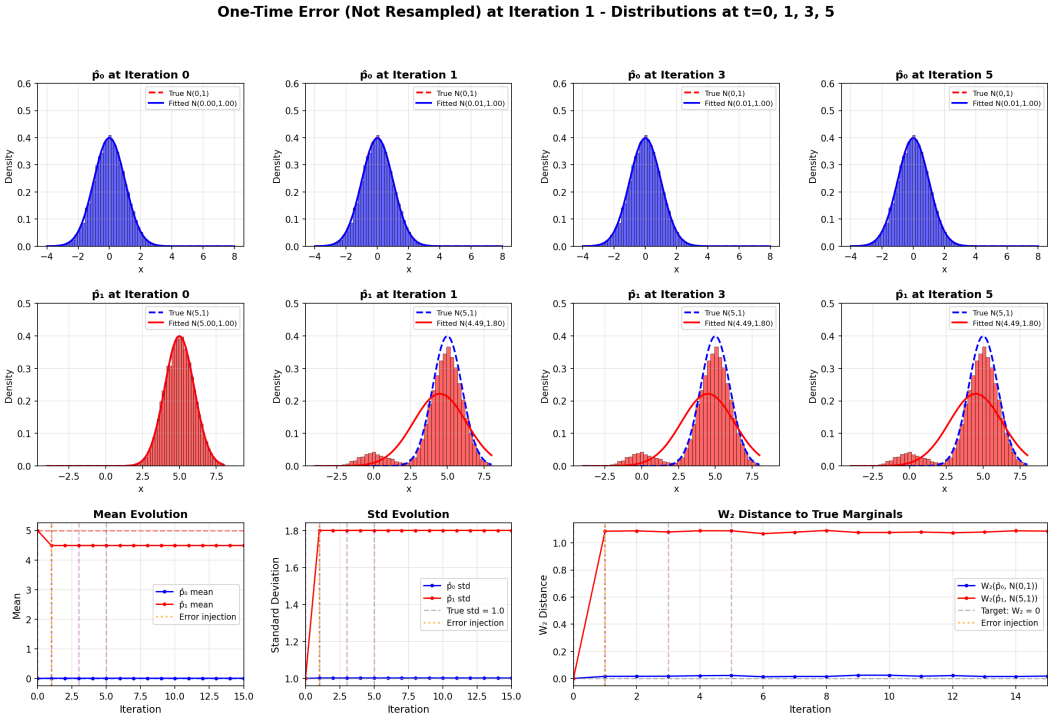


Figure 7: **One-time contamination in Algorithm 3.** At iteration 1, the forward map learns to a contaminated target $0.1\,\mathcal{N}(0,1) + 0.9\,\mathcal{N}(5,1)$ instead of $\mathcal{N}(5,1)$. No resampling the displacement remains the same but it does not affect $\hat{p}_0$.

B ADDITIONAL EXPERIMENTS

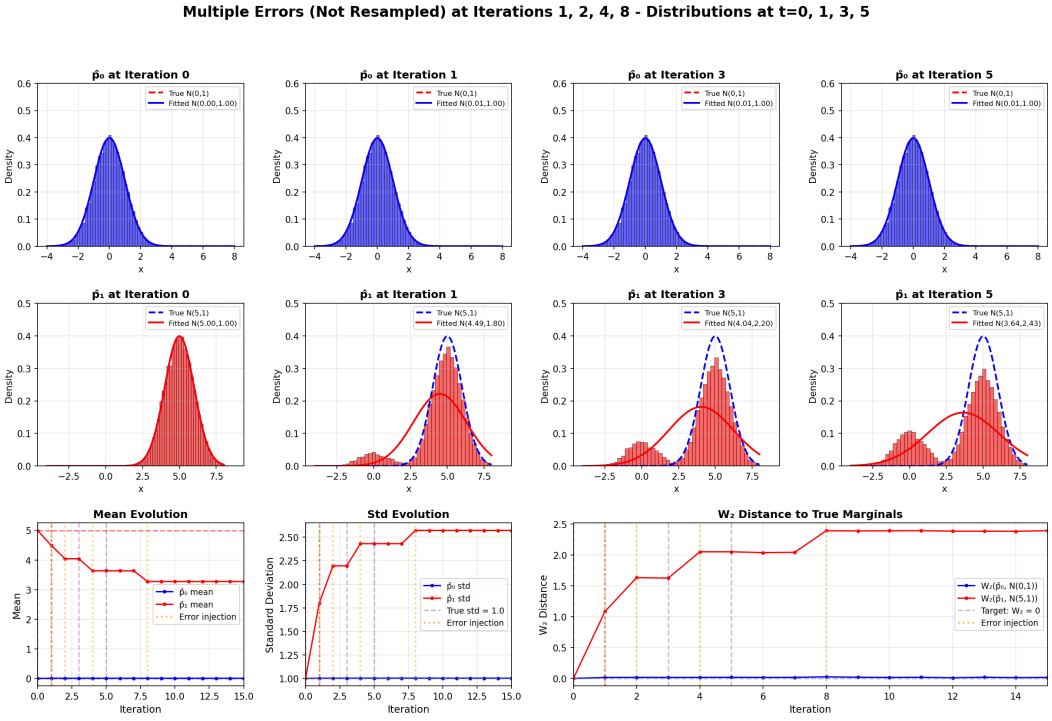


Figure 8: **Repeated contamination in Algorithm 3.** The forward map learns to $\frac{1}{10}\hat{p}_0 + \frac{9}{10}p_1$ at iterations 1, 2, 4, and 8 . Each error compounds existing corruption: $\hat{p}_1$ progressively degrades from mean 4.49 (iteration 1) to 3.66 (iteration 9) with increasing variance, while $\hat{p}_0$ drifts from 0.21 to 0.55. Unlike the one-time case (Figure 1), multiple contaminations cause cumulative degradation that resampling cannot prevent.