

# EFFICIENT DIFFUSION MODELS UNDER NONCONVEX EQUALITY AND INEQUALITY CONSTRAINTS VIA LANDING

**Anonymous authors**

Paper under double-blind review

## ABSTRACT

The generative modeling of data in constrained sets is central to scientific and engineering applications with physical, geometric, or safety constraints (e.g., molecular generation, robotics). This article constructs constrained diffusion models on a generic nonconvex feasible sets  $\Sigma$ , by introducing a unified framework that simultaneously enforces both equality and inequality constraints throughout the diffusion process. Our theory and implementations encompass both overdamped and underdamped dynamics for the forward and backward sampling. The key algorithmic ingredient is a computationally efficient landing mechanism that replaces costly and not-always-well-defined projections onto  $\Sigma$ , maintaining feasibility without Newton solves and avoiding projection failures. Leveraging underdamped dynamics whose faster mixing reduces the steps needed to reach the prior distribution, the commonly-believed unavoidable heavy forward simulation cost in the constrained diffusion is alleviated. Empirically, this reduces function evaluations, enabling more efficient inference and training while preserving sample quality and substantially lowering memory usage. On equality-only and mixed (equality and inequality) benchmarks, our method shows reasonable sample quality, while substantially reducing computational cost and function evaluations. These results indicate that landing-based enforcement combined with underdamped dynamics provides a practical and scalable recipe for constrained diffusion on nonconvex feasible sets.

## 1 INTRODUCTION

Generative modeling is a fundamental machine learning task. In recent years, denoising diffusion models [1, 2] have become the state-of-the-art in image, audio, and video generation, matching and surpassing earlier approaches such as GAN [3, 4]. Key advantages include ease of training, high-fidelity sampling, and their flexibility in conditional generation.

The majority of these impressive results have been achieved for data in an unconstrained space, particularly  $\mathbb{R}^d$ . While this suffices for digital content creation, there are many emerging applications in science and engineering, where the generation of high-fidelity samples that adhere to constraints is crucial. Examples include molecular generation [5–7], where atoms must satisfy distance or chirality constraints and respect physical laws, robotics [8–10], where trajectories are subject to dynamics, actuation limits, safety margins, and collision-avoidance, and shape optimization for engineering design [11–13], where specifications, symmetries, and manufacturing impose constraints. In these settings, a valid sample needs to remain inside a non-trivial feasible set  $\Sigma \subset \mathbb{R}^d$ , as constraint violations would render the generation physically meaningless or unsafe. Unfortunately, enforcing constraints in diffusion models is challenging: (i) Projection-based methods demand repeated Newton-iterations, which scales poorly with dimension and may fail when  $\Sigma$  is nonconvex and projections are undefined [14]; (ii) Reparametrization encodes the constraints implicitly, which requires domain knowledge, may alter the conditioning of the score matching, and the fidelity of the sampling. Reparametrizations might be hard to find in applications with complex mixed equality-inequality constraints [15]; (iii) Penalty and barrier methods [16, 17] incorporate constraints through additional terms in the objective function, which may introduce bias (barrier functions), lead to constraint violations (penalty), and result in additional hyperparameters that are difficult to tune. Conse-

quently, constrained diffusion is computationally demanding as both training and inference rely on the simulation of diffusion processes. For example, this poses a significant bottleneck for applications including robotics - where trajectories have to be generated in real time on resource-constrained hardware.

This article addresses the need for computationally-efficient constrained diffusion. Our contributions are threefold:

- **(Landing based diffusion process)** We extend the recently developed constrained optimization technique of landing [18–21] to generative modeling, enabling inexpensive first-order updates that steer every step of the diffusion process toward  $\Sigma$  without requiring exact projections or Newton iterations. Landing incurs a negligible computational overhead and provably maintains feasibility along the path.
- **(Unified framework for constraints)** Our method works for general nonconvex feasible sets as we develop an SDE-based framework that unifies both equality and inequality constraints.
- **(Fast constrained diffusion via underdamped dynamics)** The framework encompasses constrained versions of both underdamped and overdamped Langevin dynamics. The fast mixing of the underdamped version is leveraged to substantially reduce the length of forward trajectory  $N$  to reach the prior distribution, thereby cutting the dominant sampling (up to  $47\times$ ) and training cost (up to  $5\times$ ) in constrained diffusion models.

## 2 RELATED WORKS

Recent works have extended score-based diffusion models from Euclidean spaces to non-Euclidean domains. Our setup considers a domain specified by equality and inequality constraints, and closely related is a rich collection of successful generative models for manifold data [22–27]. However, while many manifolds considered (such as  $S^d$  and  $SO(n)$ ) can also be specified using equality constraints, the geometry of the constrained set can easily become too complicated to handle when there are a large number of constraints, or when the constraints introduce manifolds with boundaries or even lower dimensional structures.

Approaches directly targeting constrained generation also exist, particularly when data lie in a bounded subset  $\Sigma \subset \mathbb{R}^d$ . Several strategies have been explored. For example, classical reflected Brownian motion [28, 29] was recently leveraged to create constrained diffusion models [17, 15, 30], and the recent development of mirror Langevin dynamics and algorithms (the version of [31, 32]) was also employed for constrained generation [33]. However, the former approach is difficult to be made simulation-free and/or tractable in the conditional score, which is essential for efficiency, while the latter only works for convex constraints. Even more recently, Riemannian Denoising Diffusion Probabilistic Models (RDDPM) [34] extends score-based models to general manifolds via per-step Newton’s projections, ensuring feasibility but incurring sizable computational cost and occasional projection failures on nonconvex sets. In parallel, Riemannian Flow Matching (RFM) [26] learns manifold flows without projections for simple manifolds, yet typically needs a long integration horizon or still projections on nontrivial geometries.

In the light of these advances, we construct an efficient diffusion process that remains on feasible sets described by equality and inequality constraints. The key is to incorporate landing, a technique developed in constrained optimization [18–21], which handles non-convex constraints and guarantees feasibility (without requiring projections, retractions, or evaluations of the exponential map that tend to be expensive).

## 3 PRELIMINARIES & NOTATIONS

**Constrained set and geometry.** We implement the diffusion model on a constrained set  $\Sigma := \{x \in \mathbb{R}^d \mid h(x) = 0, g(x) \leq 0\}$  defined by smooth equality constraints  $h : \mathbb{R}^d \rightarrow \mathbb{R}^m$  and inequality constraints  $g : \mathbb{R}^d \rightarrow \mathbb{R}^l$ . For theoretical analysis, we assume that  $\Sigma$  is a stratified manifold and constraints  $h, g$  satisfy the relaxed Constant Rank Constraint Qualification (rCRCQ) [35] on a neighborhood of  $\Sigma$ . Specifically, for each  $x \in \mathbb{R}^d$  with the index set of active inequalities  $I_x := \{j \in [l] \mid g_j(x) \geq 0\}$ , we say that rCRCQ holds if, for every index set  $J \subset I_x$ , the

set  $\{\nabla h_i(y)\}_{i=1}^m \cup \{\nabla g_j(y)\}_{j \in J}$  has the same rank for any  $y$  in the neighborhood of  $x$  (see Remark 2 for further discussion on rCRCQ). We note that the rCRCQ implies that the stacked Jacobian  $\nabla J(x) \in \mathbb{R}^{(m+|I_x|) \times d}$  (with  $J(x) := [h(x), g_{I_x}(x) + \epsilon]^T$ ) has a constant rank in the neighborhood of  $x$ , where the boundary repulsion rate  $\epsilon > 0$  is a hyperparameter to be introduced later. Due to this result, the tangent space of  $\Sigma$  is characterized by the kernel of the Jacobian, given by

$$T_x \Sigma := \{p \in \mathbb{R}^d \mid \nabla J(x)p = 0\}.$$

Accordingly, the orthogonal projector  $\Pi(x) := I - \nabla J(x)^T G(x)^\dagger \nabla J(x)$  onto  $T_x \Sigma$  is well-defined, where  $G(x)^\dagger$  is the Moore-Penrose pseudo-inverse of  $G(x) := \nabla J(x) \nabla J(x)^T$ .

For the reference measure on  $\Sigma$ , we use the induced surface (Hausdorff) measure on  $\Sigma$ , denoted as  $d\sigma_\Sigma$ . In the underdamped setting, the natural phase space is the cotangent bundle given by

$$T^* \Sigma := \{(x, p) \in \mathbb{R}^d \times \mathbb{R}^d \mid x \in \Sigma, \nabla J(x)p = 0\}$$

In this manifold, the natural reference measure is the Liouville measure  $d\sigma_{T^* \Sigma}(x, p) = d\sigma_\Sigma(x) \otimes dp(x)$  where  $dp(x)$  is Lebesgue measure on  $T_x^* \Sigma := \{p \in \mathbb{R}^d \mid \nabla J(x)p = 0\}$ . For the detailed background and notations, see subsection B.1 for the overdamped, subsection B.2 for the underdamped, and the table of key notations (Table 5).

**Time grid and schedule.** In our paper, we use continuous time  $t \in [0, T]$  and a uniform grid  $t_k := k\Delta t$  for  $k \in \{0, \dots, N\}$  with  $T = N\Delta t$  for the implementation of the diffusion model with step size  $\Delta t > 0$ . Also, the noise magnitudes used in implemented diffusion models are specified by a scheduler  $\sigma : [0, T] \rightarrow \mathbb{R}_+$  and, in our case, we use the linear scheduler given by  $\sigma(t) := \sigma_{\min} + \frac{t}{T}(\sigma_{\max} - \sigma_{\min})$  with  $\sigma_k := \sigma(t_k)$ .

## 4 MAIN RESULTS

### 4.1 CONSTRAINED LANGEVIN DYNAMICS VIA LANDING

**From projections to intrinsic landing mechanisms.** Classical constrained samplers take an unconstrained (in  $\mathbb{R}^d$ ) or tangential step (in  $T_x \Sigma$ ) and then project back to  $\Sigma$ . This is often problematic: (i) on nonconvex manifolds, nearest-point projection can be multi-valued or not globally defined; (ii) per-step projection solves are costly and may fail (e.g. Newton’s method failure); and (iii) behavior near  $\partial \Sigma$  is delicate as the active set  $I_x$  changes frequently. We therefore seek a projection-free scheme that remains well-posed when local projections struggle. Our approach embeds robustness directly in the SDE via a landing term that enforces exponential decay of constraint violation  $J$ :

$$dJ(X_t) = -\alpha \sigma(t)^2 J(X_t) dt \quad (\text{Target landing property})$$

so discretization-induced infeasibility self-corrects without explicit projection. The landing modifies only the normal component of the drift, leaving tangential drift and diffusion unchanged, so trajectories evolve intrinsically on  $\Sigma$  (or  $T^* \Sigma$  in the underdamped case). Formally, any diffusion process  $X_t$  with this property enjoys the guarantees stated next. For the detailed proofs of mathematical claims below, see Appendix B.

**Lemma 1** (Exponential decay of constraint functions). *Under the target property  $dJ(X_t) = -\alpha \sigma(t)^2 J(X_t) dt$ , the diffusion process  $X_t$  satisfies the following constraint satisfaction property almost surely:*

$$h_i(X_t) = h_i(X_0) e^{-\alpha S(t)}, \quad t \geq 0$$

and

$$\begin{cases} g_j(X_t) = -\epsilon + (g_j(X_0) + \epsilon) e^{-\alpha S(t)}, & t \leq \tau_{j, \epsilon} \\ g_j(X_t) \leq 0, & t \geq \tau_{j, \epsilon}, \end{cases}$$

where  $S(t) := \int_0^t \sigma(s)^2 ds$  and  $\tau_{j, \epsilon}$  are defined to be

$$\tau_{j, \epsilon} := \inf \left\{ t \geq 0 \mid S(t) \geq \frac{1}{\alpha} \ln \left( \frac{g_j(X_0) + \epsilon}{\epsilon} \right) \right\}, \quad \forall j \in I_{X_0}.$$

**Constrained Overdamped Langevin dynamics via Landing (OLLA).** We first derive such landing-based constrained Langevin dynamics for the overdamped case. By viewing constrained Langevin dynamics in Lagrangian form [36], we can pick a Lagrangian process  $d\lambda_t$  so that it can impose the target property  $dJ(X_t) = -\alpha\sigma(t)^2 J(X_t)dt$  and have a closed-form SDE as follows:

**Proposition 1** (Construction, stationarity and backward process of OLLA). *Consider the following Lagrangian form constrained overdamped Langevin dynamics of  $X_t \sim q_t$ :*

$$dX_t = -\frac{1}{2}\sigma(t)^2 \nabla f(X_t)dt + \sigma(t) \circ dW_t + \nabla J(X_t)^T d\lambda_t \quad (1)$$

where  $d\lambda_t$  is the adapted process such that  $dJ(X_t) = -\alpha\sigma(t)^2 J(X_t)$ . The explicit solution of  $d\lambda_t$  provides the closed form SDE of (1) as follows:

$$dX_t = - \left[ \frac{\sigma(t)^2}{2} \Pi(X_t) \nabla f(X_t) + \underbrace{\alpha\sigma(t)^2 \nabla J(X_t)^T G^\dagger(X_t) J(X_t)}_{\text{Landing term}} \right] dt + \frac{\sigma(t)^2}{2} \mathcal{H}(X_t)dt + \sigma(t) \Pi(X_t) dW_t,$$

where  $\mathcal{H}$  is the mean curvature correction term defined as

$$\mathcal{H}(x) := -\nabla J(x)^T G^\dagger(x) \left[ \text{Tr}(\nabla^2 J_1(x) \Pi(x)), \dots, \text{Tr}(\nabla^2 J_{m+|I_x|}(x) \Pi(x)) \right]^T.$$

Furthermore, the backward process  $\overleftarrow{X}_t$  of OLLA is given as:

$$d\overleftarrow{X}_t = \frac{1}{2}\sigma(T-t)^2 \Pi(\overleftarrow{X}_t) \left[ \nabla f(\overleftarrow{X}_t) + 2\nabla \ln q_{T-t}(\overleftarrow{X}_t) \right] dt + \frac{1}{2}\sigma(T-t)^2 \mathcal{H}(\overleftarrow{X}_t)dt - \underbrace{\alpha\sigma(T-t)^2 \nabla J(\overleftarrow{X}_t)^T G^\dagger(\overleftarrow{X}_t) J(\overleftarrow{X}_t)}_{\text{Landing term}} dt + \sigma(T-t) \Pi(\overleftarrow{X}_t) \circ d\overleftarrow{W}_t.$$

By assuming  $\sigma(t)$  constant and  $X_0 \in \Sigma$ , OLLA has the stationary distribution  $q_\Sigma \propto \exp(-f(x))$  with respect to  $d\sigma_\Sigma$ .

**Constrained Underdamped Langevin Dynamics via Landing (ULLA).** In the underdamped case, we are required to satisfy both the target property  $dJ(X_t) = -\alpha\sigma(t)^2 J(X_t)dt$  and also the momentum tangency constraint  $\nabla J(X_t)P_t = 0$  so that  $(X_t, P_t) \in T^*\Sigma$  for  $t \geq 0$ . Therefore, we control the two Lagrangian processes  $d\lambda_t, d\mu_t$  to impose such constraints, and the resulting solution of Lagrangian processes produces the following closed SDE for ULLA:

**Proposition 2** (Construction, stationarity, and backward process of ULLA). *Consider the following Lagrangian form constrained underdamped Langevin of  $(X_t, P_t) \sim q_t$ :*

$$\begin{cases} dX_t = \sigma(t)^2 P_t dt + \nabla J(X_t)^T d\lambda_t \\ dP_t = -\sigma(t)^2 \nabla f(X_t)dt - \sigma(t)^2 \gamma P_t dt + \sigma(t) \sqrt{2\gamma} \circ dW_t + \nabla J(X_t)^T d\mu_t, \end{cases} \quad (2)$$

where  $d\lambda_t, d\mu_t$  are the adapted processes such that  $dJ(X_t) = -\alpha\sigma(t)^2 J(X_t)$  (position constraint) and  $\nabla J(X_t)P_t = 0$  (momentum tangency constraint), respectively.

Assuming  $\nabla J(X_0)P_0 = 0$ , the explicit solution of  $d\lambda_t, d\mu_t$  provides the closed form SDE of (2) as follows:

$$\begin{cases} dX_t = \sigma(t)^2 P_t dt - \alpha\sigma(t)^2 \nabla J(X_t)^T G^\dagger(X_t) J(X_t)dt \\ dP_t = \Pi(X_t) \left[ -\sigma(t)^2 \nabla f(X_t) - \sigma(t)^2 \gamma P_t dt + \sigma(t) \sqrt{2\gamma} dW_t \right] \\ \quad - \sigma(t)^2 \nabla J(X_t)^T G^\dagger(X_t) [\mathcal{H}_1(X_t, P_t) - \alpha\mathcal{H}_2(X_t, P_t)] dt, \end{cases}$$

where  $\mathcal{H}_1 \in \mathbb{R}^{m+|I_x|}$ ,  $\mathcal{H}_2 \in \mathbb{R}^{(m+|I_x|) \times (m+|I_x|)}$  are the curvature correction terms defined as

$$[\mathcal{H}_1(x, p)]_i := p^T \nabla^2 J_i(x) p, \quad [\mathcal{H}_2(x, p)]_i := p^T \nabla^2 J_i(x) (\nabla J(x)^T G^\dagger(x) J(x))$$

with  $[\mathcal{H}_1(x, p)]_i$ ,  $[\mathcal{H}_2(x, p)]_i$  being the  $i$ th entry and column of  $\mathcal{H}_1(x, p)$ ,  $\mathcal{H}_2(x, p)$  respectively.

Furthermore, the backward process  $\overleftarrow{X}_t$  of ULLA is given as:

$$\begin{cases} d\overleftarrow{X}_t = -\sigma(T-t)^2 \overleftarrow{P}_t dt - \underbrace{\alpha \sigma(T-t)^2 \nabla J(\overleftarrow{X}_t) G^\dagger(\overleftarrow{X}_t) J(\overleftarrow{X}_t)}_{\text{Landing term}} dt \\ d\overleftarrow{P}_t = \Pi(\overleftarrow{X}_t) \left[ \sigma(T-t)^2 \left[ \nabla f(\overleftarrow{X}_t) dt + \gamma \overleftarrow{P}_t dt + 2\gamma \nabla_p \ln q_{T-t}(\overleftarrow{X}_t, \overleftarrow{P}_t) \right] dt \right] + \\ \sigma(T-t)^2 \nabla J(\overleftarrow{X}_t)^T G^\dagger(\overleftarrow{X}_t) \left[ \mathcal{H}_1(\overleftarrow{X}_t, \overleftarrow{P}_t) + \alpha \mathcal{H}_2(\overleftarrow{X}_t, \overleftarrow{P}_t) \right] dt + \sigma(T-t) \sqrt{2\gamma} \Pi(\overleftarrow{X}_t) d\overleftarrow{W}_t. \end{cases}$$

By assuming  $\sigma(t)$  constant and  $X_0 \in \Sigma$ ,  $\nabla J(X_0) P_0 = 0$ , ULLA has the stationary distribution  $q_{T^* \Sigma} \propto \exp(-f(x) - \frac{1}{2} \|p\|^2)$  with respect to  $d\sigma_{T^* \Sigma}$ .

**Error decomposition and benefits of ULLA.** Theoretically, generation error via backward process decomposes into mixing, discretization, and score estimation terms ( $W_2(q_0, p_0^\theta) \leq \mathcal{E}_{\text{mix}} + \mathcal{E}_{\text{disc}} + \mathcal{E}_{\text{score}}$ ). In this point of view, ULLA significantly reduces  $\mathcal{E}_{\text{mix}}$  via ballistic dynamics, accelerating convergence and enabling smaller trajectory lengths  $N$  compared to OLLA. Additionally, the momentum variable mitigates score singularities near  $t = 0$ , yielding a potentially smoother training objective for  $\mathcal{E}_{\text{score}}$ . We refer to Remark 3 for a detailed discussion.

## 4.2 TRANSITION KERNELS FOR FORWARD / BACKWARD PROCESS

In this section, we outline the discretization of the OLLA and ULLA processes. For the notations, we let  $x_k$  for  $k \in \{1, \dots, N\}$  to be the position vector at  $k$ -th discrete step of the diffusion process, and set  $q_k, p_k, p_k^\theta$  to be the marginal probability densities for forward and backward processes, and the parametrized backward process of  $x_k$ . Also, we set  $p_N, \rho(\cdot|x)$  to be the prior of position and momentum where the momentum prior is given by  $\Pi(x)\zeta \sim \rho(\cdot|x)$  with  $\zeta \sim \mathcal{N}(0, I_d)$ . For detailed derivations of the discretization schemes below, we refer to subsection B.4.

**Discretization of OLLA.** For OLLA, the discretization is straightforward. We employ a standard Euler-Maruyama scheme to integrate the corresponding SDE as follows:

$$\begin{cases} x_{k+1} = x_k + \frac{\sigma_k^2 \Delta t}{2} [-\Pi(x_k) \nabla f(x_k) - \alpha \sigma_k^2 (\nabla J^T G^\dagger J)(x_k)] + \sigma_k \sqrt{\Delta t} \Pi(x_k) \zeta_k + \kappa_k^O \\ x_k = x_{k+1} + \frac{\sigma_{k+1}^2 \Delta t}{2} \Pi(x_{k+1}) [\nabla f(x_{k+1}) + s_{k+1}(x_{k+1})] + \sigma_{k+1} \sqrt{\Delta t} \Pi(x_{k+1}) \zeta_{k+1} \\ - \alpha \sigma_{k+1}^2 \Delta t (\nabla J^T G^\dagger J)(x_{k+1}) \zeta_{k+1} + \kappa_{k+1}^O \end{cases}$$

where  $\kappa_k^O := \frac{\sigma_k^2}{2} \mathcal{H}(x_k) \Delta t$  is the mean curvature term and  $s_k(x_k) := 2\nabla \ln q_k(x_k)$ , which can be learned by a neural network  $s_{k+1}^\theta$ .

**Discretization of ULLA.** For ULLA, we adopt a specialized scheme to achieve  $2 \times$  memory efficiency, which becomes critical for storing long forward trajectories during training. The method is based on the 1st order non-symmetric OBA splitting integrator. To eliminate the need to explicitly store the momentum trajectory, we first use an approximated  $\tilde{B}$  step, which relies on the before-O step momentum on correction terms  $\mathcal{H}_1, \mathcal{H}_2$ , rather than after, and secondly, we leverage the recursive nature of the update rule to express the momentum at step  $k$  as a function of positions at previous steps, using an approximated momentum vector  $\tilde{p}_k$ . This ‘‘collapses’’ the dynamics into the 2nd order Markov chain solely depending on position variables  $x_k$  as follows:

$$\begin{cases} x_{k+1} = x_k + \sigma_k^2 \Delta t \Pi(x_k) \left[ a_k \tilde{p}_k^{\text{fwd}} - \sigma_k^2 \Delta t \nabla f(x_k) + \sqrt{1 - a_k^2} \zeta_k \right] - \alpha \sigma_k^2 \Delta t (\nabla J^T G^\dagger J)(x_k) \\ + \kappa_{k, \text{fwd}}^U \\ x_k = x_{k+1} - \sigma_{k+1}^2 \Delta t \Pi(x_{k+1}) \left[ a_{k+1} \tilde{p}_{k+1}^{\text{bwd}} + \sigma_{k+1}^2 \Delta t [\nabla f(x_{k+1}) + s_{k+1}(x_{k+1}, \tilde{p}_{k+1}^{\text{bwd}})] \right] \\ + \sigma_{k+1}^2 \Delta t \sqrt{1 - a_{k+1}^2} \Pi(x_{k+1}) \zeta_{k+1} - \alpha \sigma_{k+1}^2 \Delta t (\nabla J^T G^\dagger J)(x_{k+1}) + \kappa_{k+1, \text{bwd}}^U \end{cases}$$

where  $a_k = e^{-\gamma\sigma_k^2\Delta t} \in [0, 1]$  is a decaying factor, the approximated momentum are defined by  $\tilde{p}_k^{\text{fwd}} := \Pi(x_k) \left( (x_k - x_{k-1}) / (\sigma_{k-1}^2 \Delta t) \right)$ ,  $\tilde{p}_{k+1}^{\text{bwd}} := \Pi(x_{k+1}) \left( (x_{k+2} - x_{k+1}) / (\sigma_{k+2}^2 \Delta t) \right)$ , and the curvature correction terms are provided as:

$$\begin{cases} \kappa_{k,\text{fwd}}^U := -\sigma_k^4 \Delta t^2 \nabla J(x_k)^T G^\dagger(x_k) [\mathcal{H}_1(x, \tilde{p}_k^{\text{fwd}}) - \mathcal{H}_2(x_k, \tilde{p}_k^{\text{fwd}})] \\ \kappa_{k+1,\text{bwd}}^U := -\sigma_{k+1}^4 \Delta t^2 \nabla J(x_{k+1})^T G^\dagger(x_{k+1}) [\mathcal{H}_1(x, \tilde{p}_{k+1}^{\text{bwd}}) + \alpha \mathcal{H}_2(x_{k+1}, \tilde{p}_{k+1}^{\text{bwd}})]. \end{cases}$$

Similarly, we set  $s_{k+1}(x_{k+1}, \tilde{p}_{k+1}^{\text{bwd}}) := 2\gamma (\nabla_p \ln q_{k+1}(x_{k+1}, \tilde{p}_{k+1}^{\text{bwd}}) + \tilde{p}_{k+1}^{\text{bwd}})$  and train a neural network  $s_{k+1}^\theta$  to approximate this.

**Remark 1** (Discretization of Constrained Langevin dynamics via Newton solver). Projection-based variants of proposed methods, denoted OLLA-P and ULLA-P, can be obtained by dropping all normal landing and correction terms, and instead solving a Lagrangian multiplier system at each step so that  $h(x_k) = 0$ . For the detailed derivation, we refer to [subsection B.4](#).

### 4.3 CONDITIONAL WASSERSTEIN PATH MATCHING (CWPM)

Since the proposed OLLA and ULLA do not use per-step projections, intermediate samples  $x_k$  may exhibit minor constraint violations and lie off  $\Sigma$ . This renders previously proposed training loss, such as DT-ELBO [34] or score matching [24, 25], theoretically unstable, as they rely on the assumption that  $x_k \in \Sigma$ , which can lead to singularity issues. This problem is particularly acute when measuring the NLL loss, where small violations can introduce substantial bias and undermine its reliability as a sample quality metric. To resolve these theoretical issues, we propose the CWPM framework as below, which is based on the Wasserstein distance rather than KL-divergence, eliminating such theoretical singularities. The derivation involves the relationship between Gelbrich distance and 2-Wasserstein distance [37, 38], and refer to [Appendix D](#) for detailed proofs and assumptions.

**Theorem 1** (CWPM variational bound – overdamped, informal). *Let  $T_{k+1}^\theta = p^\theta(x_k | x_{k+1})$  be the backward transition kernel of the discretized OLLA and define the circuitous density at step  $k$  as*

$$\sigma_k := q_k T_k^\theta T_{k-1}^\theta, \dots, T_1^\theta, \quad \sigma_0 := q_0.$$

Assuming existence of  $\Lambda_{k+1} > 0$  such that

$$W_2(\sigma_k, \sigma_{k+1}) \leq \Lambda_{k+1} W_2(q_k, q_{k+1} T_{k+1}^\theta) + \mathcal{O}(\sqrt{\Delta t}), \quad k \in \{0, \dots, N-1\}$$

which holds under minor regularity assumptions on the score and constraint functions ([Lemma D.1](#)), we have

$$W_2(q_0, p_0^\theta) \leq \mathbb{E}_{q(x_{0:N})} \left[ \sum_{k=0}^{N-1} \|\Pi(x_{k+1})(x_k - \mu_{k+1}^o(x_{k+1}))\|^2 \right] + C^o,$$

where  $\mu_{k+1}^o(x_{k+1})$  is the tangential part mean of the parametrized backward process of OLLA defined by

$$\mu_{k+1}^o(x_{k+1}) := x_{k+1} + \frac{\sigma_{k+1}^2 \Delta t}{2} \Pi(x_{k+1}) [\nabla f(x_{k+1}) + s_\theta^{k+1}(x_{k+1})]$$

and  $C^o$  is a constant independent of  $\theta$ .

Similarly, we have the following version for the underdamped case:

**Theorem 2** (CWPM variational bound – underdamped, informal). *Let  $y_k = (x_k, x_{k+1}) \in \mathbb{R}^{2d}$  where  $x_k \sim q_k, x_{k+1} \sim q_{k+1}$ . Define  $\bar{q}_k$  to be the law of  $y_k$  and set  $\bar{T}_{k+1}^\theta = p^\theta(y_k | y_{k+1})$  to be the associated backward transition kernel to  $y_k$ . We set the circuitous density at step  $k$  as*

$$\bar{\sigma}_k := \bar{q}_k \bar{T}_k^\theta \bar{T}_{k-1}^\theta, \dots, \bar{T}_1^\theta, \quad \bar{\sigma}_0 := \bar{q}_0.$$

Assuming existence of  $\bar{\Lambda}_{k+1} > 0$  such that

$$W_2(\bar{\sigma}_k, \bar{\sigma}_{k+1}) \leq \bar{\Lambda}_{k+1} W_2(\bar{q}_k, \bar{q}_{k+1} \bar{T}_{k+1}^\theta) + \mathcal{O}(\Delta t), \quad k \in \{0, \dots, N-1\},$$

which holds under minor regularity assumptions on the score and constraint functions (Lemma D.2), we have

$$W_2(q_0, p_0^\theta) \leq \mathbb{E}_{q(x_{0:N})} \mathbb{E}_{\rho_N(p_N|x_N)} \left[ \sum_{k=0}^{N-1} \|\Pi(x_{k+1})(x_k - \mu_{k+1}^u(x_{k+1}, x_{k+2}))\|^2 \right] + C^u$$

where  $\mu_{k+1}^u(x_{k+1})$  is the tangential part mean of the parametrized backward process of ULLA defined by

$$\mu_{k+1}^u := x_{k+1} - \sigma_{k+1}^2 \Delta t \Pi(x_{k+1}) \left[ a_{k+1} \tilde{p}_{k+1}^{bwd} + \sigma_{k+1}^2 \Delta t [\nabla f(x_{k+1}) + s_\theta^{k+1}(x_{k+1}, \tilde{p}_{k+1}^{bwd})] \right]$$

and  $C^u$  is a constant independent of  $\theta$ .

**Choice of Training Loss.** We remark that other works on diffusion models [1, 39, 40] demonstrated that choosing the training loss weight  $\lambda(k)$  proportional to the inverse of the variance up to a proportionality constant of 1/2 (in our case,  $\lambda(k) = 1/(2\sigma_{k+1}^2 \Delta t)$  for overdamped and  $\lambda(k) = 1/(2\sigma_{k+1}^4 \Delta t^2 (1 - a_{k+1}^2))$  for underdamped) is helpful for training the score network. Notably, the resulting training losses

$$\begin{cases} L_{\text{CWPM}}^{\text{over}}(\theta) = \mathbb{E}_{q(x_{0:N})} \left[ \sum_{k=0}^{N-1} \frac{\|\Pi(x_{k+1})(x_k - \mu_{k+1}^o(x_{k+1}))\|^2}{2\sigma_{k+1}^2 \Delta t} \right] \\ L_{\text{CWPM}}^{\text{under}}(\theta) = \mathbb{E}_{q(x_{0:N})} \mathbb{E}_{\rho_N(p_N|x_N)} \left[ \sum_{k=0}^{N-1} \frac{\|\Pi(x_{k+1})(x_k - \mu_{k+1}^u(x_{k+1}, x_{k+2}))\|^2}{2\sigma_{k+1}^4 \Delta t^2 (1 - a_{k+1}^2)} \right] \end{cases}$$

lead to exactly the same training loss provided in DT-ELBO ([34], or Lemma C.1 and Lemma C.2) without the requirement  $x_k \in \Sigma$ . For the complete algorithms, refer to Algorithm 1 and Algorithm 2.

## 5 EXPERIMENTS

We evaluate on benchmarks largely following RDDPM [34]—Earth/climate datasets, mesh data, the  $SO(10)$  manifold, and Alanine dipeptide—and add a 7-Degree of Freedom (DOF) robot arm trajectory task. We compare against constrained generative model algorithms such as RFM [26] and RDDPM, as well as Euclidean forward with backward variants baselines to highlight the importance of handling intrinsic geometry and learning the score function on  $\Sigma$ . We leave experimental setup, baseline descriptions, and hyperparameters in Appendix E. Also, following a similar technique in a landing-based constrained sampling [41], we set the correction terms  $\kappa = 0$  to save computational cost by avoiding Hessian-related terms.

### 5.1 EQUALITY-ONLY SCENARIO TASKS

**Earth and climate science datasets.** This benchmark lives on the 2-sphere  $S^2$ , where nearest-point projection is globally available, so landing-based dynamics are not strictly required.

Nevertheless, we use this dataset to (i) assess the intrinsic benefits of **underdamped** dynamics and (ii) quantify the sampling quality – computational cost trade-off under landing. From Figure 1, due to faster mixing of the underdamped dynamics, underdamping markedly reduce the needed forward length  $N$ : ULLA-P is stable without projection failure even at  $N = 40$ , whereas RDDPM (OLLA-P) requires at least  $N \approx 150$  to avoid failures. Thus, smaller  $N$  yields large training-time savings while preserving comparable sample quality. Although exact projections are available here, Table 1 indicate that ULLA incurs comparable sampling quality under minor constraint violations; visual comparisons of ULLA (subsection E.3) show similar generated distribution to projection-based methods, supporting the practical value of landing.

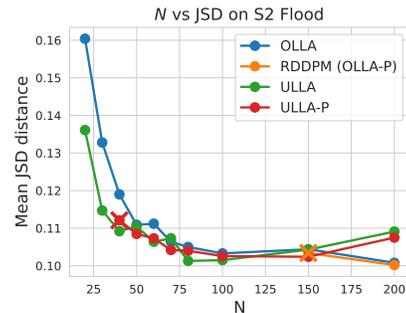


Figure 1: Mean JSD on  $S^2$  flood versus trajectory length  $N$ . Cross marker ( $\times$ ) indicates the smallest  $N$  values after which projection failures no longer occur during the forward process.

Table 1: **Generative performance on Earth & Climate distributions (Top) and Mesh histograms (Bottom).** We report the Jensen-Shannon Distance (JSD) computed on 2D spherical coordinate  $(\theta, \phi)$  histograms (Earth & Climate) and mesh face histograms (Mesh) (lower is better). The right-most columns report the average of total training wall-clock time in seconds across all experiments; values in parentheses denote the time spent on forward trajectory simulation during training (or ODE solving time for RFM). We also report the average equality violation  $|h|$  of generated samples. Mean  $\pm$  standard errors are calculated over 5 independent runs. Across both domains, our landing-based methods (OLLA, ULLA) achieve comparable JSDs while being significantly faster than Riemannian-based diffusion model baselines.

Earth and Climate Science Datasets						
Method / dataset size	Volcano (827)	Earthquake (6120)	Flood (4875)	Fire (12809)	Training time (Sampling time)	Avg. $ h $
<i>Riemannian-based</i>						
RFM ( $N = 1000$ )	0.116 $\pm$ .002	0.089 $\pm$ .001	0.108 $\pm$ .002	0.058 $\pm$ .001	4019 (0)	$5.24 \times 10^{-8}$
RDDPM ( $N = 400$ )	0.123 $\pm$ .004	0.093 $\pm$ .002	0.106 $\pm$ .002	0.051 $\pm$ .001	12388 (3302)	$1.67 \times 10^{-8}$
<i>Euclidean fwd. + bwd. variant</i>						
Euclidean ( $N = 50$ )	0.158 $\pm$ .005	0.163 $\pm$ .004	0.135 $\pm$ .016	0.140 $\pm$ .001	1130 (221)	$4.14 \times 10^{-2}$
Projected ( $N = 50$ )	0.156 $\pm$ .005	0.152 $\pm$ .003	0.133 $\pm$ .012	0.133 $\pm$ .001	1026 (221)	$2.11 \times 10^{-8}$
Lagrangian ( $N = 50$ )	0.156 $\pm$ .004	0.152 $\pm$ .003	0.137 $\pm$ .011	0.133 $\pm$ .001	1027 (220)	$8.47 \times 10^{-10}$
Guided ( $N = 50$ )	0.160 $\pm$ .006	0.174 $\pm$ .003	0.146 $\pm$ .021	0.158 $\pm$ .002	1048 (220)	$2.07 \times 10^{-2}$
<i>Ours</i>						
OLLA ( $N = 100$ )	0.128 $\pm$ .005	0.096 $\pm$ .002	0.103 $\pm$ .002	0.060 $\pm$ .001	1686 (749)	$3.88 \times 10^{-9}$
ULLA-P ( $N = 100$ )	0.122 $\pm$ .007	0.092 $\pm$ .001	0.103 $\pm$ .002	0.053 $\pm$ .001	1631 (1154)	$8.41 \times 10^{-9}$
ULLA-P ( $N = 50$ )	0.126 $\pm$ .005	0.096 $\pm$ .001	0.108 $\pm$ .001	0.064 $\pm$ .002	1018 (568)	$1.37 \times 10^{-8}$
ULLA ( $N = 50$ )	0.125 $\pm$ .005	0.099 $\pm$ .002	0.110 $\pm$ .001	0.069 $\pm$ .002	1021 (530)	$2.14 \times 10^{-9}$
Mesh Data (Grid Laplacian Eigenfunctions with index $k$ )						
Method	Stanford Bunny $k = 50$ $k = 100$		Spot the Cow $k = 50$ $k = 100$		Training time (Sampling time)	Avg. $ h $
<i>Riemannian-based</i>						
RFM ( $N = 1000$ )	0.035 $\pm$ .001	0.047 $\pm$ .001	0.043 $\pm$ .002	0.050 $\pm$ .002	145424 (112244)	$1.47 \times 10^{-4}$
RDDPM ( $N = 400$ )	0.032 $\pm$ .001	0.034 $\pm$ .000	0.046 $\pm$ .001	0.034 $\pm$ .001	1916 (126.4)	$1.56 \times 10^{-5}$
<i>Euclidean fwd. + bwd. variant</i>						
Euclidean ( $N = 30$ )	0.040 $\pm$ .001	0.047 $\pm$ .001	0.048 $\pm$ .001	0.063 $\pm$ .001	212 (0.3)	$4.20 \times 10^{-2}$
Projected ( $N = 30$ )	0.049 $\pm$ .000	0.051 $\pm$ .001	0.057 $\pm$ .001	0.068 $\pm$ .001	210 (0.3)	$6.36 \times 10^{-8}$
Lagrangian ( $N = 30$ )	0.047 $\pm$ .001	0.050 $\pm$ .001	0.056 $\pm$ .001	0.067 $\pm$ .001	233 (0.3)	$1.13 \times 10^{-4}$
Guided ( $N = 30$ )	0.068 $\pm$ .005	0.050 $\pm$ .002	0.051 $\pm$ .001	0.065 $\pm$ .001	221 (0.3)	$8.23 \times 10^{-1}$
<i>Ours</i>						
OLLA ( $N = 100$ )	0.030 $\pm$ .000	0.032 $\pm$ .001	0.047 $\pm$ .001	0.035 $\pm$ .001	642 (4.0)	$3.44 \times 10^{-6}$
ULLA-P ( $N = 50$ )	0.040 $\pm$ .001	0.038 $\pm$ .001	0.040 $\pm$ .001	0.035 $\pm$ .001	387 (10.5)	$4.20 \times 10^{-5}$
ULLA ( $N = 30$ )	0.029 $\pm$ .001	0.033 $\pm$ .001	0.044 $\pm$ .001	0.036 $\pm$ .001	360 (2.0)	$1.52 \times 10^{-7}$

**3D Mesh data on learned manifold.** Unlike the 2-sphere, meshes lie on manifolds where nearest-point projection is not globally defined and, in our benchmark, must be approximated by a Newton solver because the constraint  $h(x) = 0$  is represented by a learned neural network - making projection-based sampling computationally expensive. In this regime, landing becomes particularly effective: as shown in Table 1, ULLA and ULLA-P show comparable JSD of RDDPM and RFM with far fewer steps ( $N = 30, 50$ ), yielding  $5\times$  faster training and up to  $47\times$  faster sampling than RDDPM. The gains stem from the combinations of the following facts: (i) the underdamped dynamics permits much smaller  $N$ , and (ii) landing (particularly without curvature corrections) requires only a single constraint-gradient evaluation per step, avoiding iterative projections. These improvements indicate that, for complex learned manifolds where projection is expensive, ULLA provides a scalable and efficient alternative.

**High-dimensional special orthogonal group:  $SO(10)$ .** This experiment evaluates scalability on the high-dimensional Lie group  $SO(10) \subset \mathbb{R}^{100}$ , defined by 55 equality constraints ( $X^T X = I$ );  $\det(X) = 1$  condition is checked based on rejection. The synthetic distribution is multimodal with  $m$  modes, and sampling quality is assessed by power-trace statistics. As shown in Figure 2 and subsection E.3, landing-based methods (ULLA/ULLA-P/OLLA) remain efficient on this complex manifold, producing high-quality samples with a forward trajectory length of  $N = 50$ , whereas RDDPM requires at least  $N \approx 150$  to avoid projection failures.

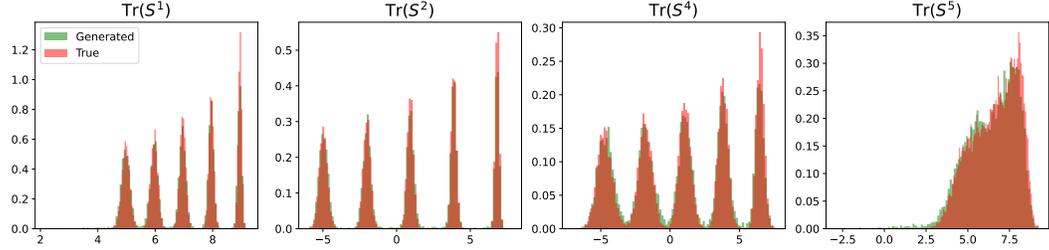


Figure 2: Histograms of the power-trace statistics  $\text{Tr}(S^k)$  for  $k \in \{1, 2, 4, 5\}$  on  $SO(10)$  with  $m = 5$ . Green: samples generated by ULLA; Red: ground truth. Forward trajectory length  $N = 50$ . The close overlap indicates that ULLA matches the target marginal on  $\text{Tr}(S^k)$ .

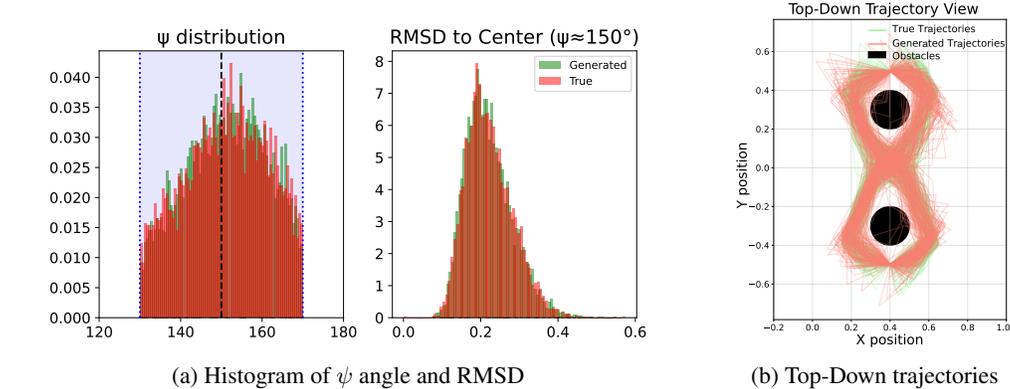


Figure 3: Comparison of constrained molecule / robot arm trajectory generation: (a) histogram of  $\psi$  angle / Root Mean Square Deviation (RMSD) to center reference point; blue shaded area is feasible region by inequality constraints  $\psi \in [130^\circ, 170^\circ]$ , (b) top-down trajectory view of generated and ground-truth trajectory samples in the 7-DOF robot arm experiment.

## 5.2 MIXED SCENARIO TASKS

**Alanine dipeptide and 7-DOF robot arm.** We further evaluate our landing algorithms under complicated mixed constraints setup.

The provided feasible set  $\Sigma$  are defined by complex intersections of equality and inequality constraints. In these settings, exact projections are often numerically unstable or computationally prohibitive. As summarized in Table 4, standard baselines encounter significant difficulties: the *Projected* Euclidean variant failed in the high-dimensional 7-DOF robot arm task due to severe projection failures, while the *Lagrangian* Euclidean variant failed to converge to a high-quality distribution in the Dipeptide task. For similar issues, OLLA, OLLA-P, ULLA-P failed in this setup.

In contrast, our proposed ULLA method demonstrates superior performance compared to the valid Euclidean forward-backward variants. ULLA not only achieves significantly lower JSDs, consistently outperforming Euclidean baselines even as the dimension scales, but also

Table 2: Effect of  $\alpha$  on JSD,  $\mathbb{E}[|h|]$

$\alpha$	JSD	$\mathbb{E}[ h(x) ]$
1.0	0.134±0.004	$5.51 \times 10^{-3}$
10.0	0.053±0.004	$1.02 \times 10^{-3}$
50.0	0.033±0.002	$7.20 \times 10^{-5}$
100.0	0.051±0.003	$3.70 \times 10^{-5}$

Table 3: Effect of  $\epsilon$  on JSD,  $\mathbb{E}[g^+]$

$\epsilon$	JSD	$\mathbb{E}[g^+(x)]$
0.01	0.048±0.002	$5.49 \times 10^{-7}$
0.05	0.033±0.002	$7.95 \times 10^{-8}$
0.1	0.052±0.007	$5.26 \times 10^{-4}$
0.5	0.081±0.009	$1.98 \times 10^{-1}$

Table 4: **Generative performance on Mixed Constraint Tasks.** The top panel presents the scalability analysis on the 7-DOF robot arm trajectory generation with increasing ambient dimensions  $d$ . For this task, JSD is measured on 2D histograms formed by projecting generated data via PCA fitted on the ground truth. The *Projected* method is excluded here due to severe projection failures during sampling. The bottom panel reports performance on Alanine Dipeptide conformation generation, measuring JSD on the marginal histograms of the dihedral angle  $\psi$  and the RMSD to a reference structure. Constraint violations ( $|h|, |g^+|$ ) are averaged over generated samples. The *Lagrangian* method is omitted in this task due to failure of convergence.

7-DOF Robot Arm Trajectory (Scalability with dimension $d$ )						
Method	JSD across dimension ( $d$ )				Avg. $ h $	Avg. $ g^+ $
	$d = 140$	$d = 280$	$d = 420$	$d = 560$		
<i>Euclidean fwd. + bwd. variant</i>						
Euclidean ( $N = 100$ )	0.498 $\pm$ .028	0.656 $\pm$ .034	0.647 $\pm$ .022	0.750 $\pm$ .025	$8.3 \times 10^{-1}$	$5.3 \times 10^{-3}$
Lagrangian ( $N = 100$ )	0.769 $\pm$ .043	0.816 $\pm$ .005	0.831 $\pm$ .001	0.831 $\pm$ .002	$4.9 \times 10^{-2}$	$1.6 \times 10^{-3}$
Guided ( $N = 100$ )	0.499 $\pm$ .028	0.655 $\pm$ .041	0.665 $\pm$ .013	0.740 $\pm$ .033	$8.0 \times 10^{-1}$	$4.9 \times 10^{-3}$
<i>Ours</i>						
ULLA ( $N = 100$ )	0.275 $\pm$ .011	0.295 $\pm$ .006	0.366 $\pm$ .005	0.391 $\pm$ .012	$1.8 \times 10^{-5}$	$1.5 \times 10^{-9}$
Alanine Dipeptide Conformation						
Method	JSD ( $\psi$ angle)		JSD (RMSD)		Avg. $ h $	Avg. $ g^+ $
<i>Euclidean fwd. + bwd. variant</i>						
Euclidean ( $N = 100$ )	0.150 $\pm$ .003		0.057 $\pm$ .001		$5.7 \times 10^{-2}$	$2.4 \times 10^{-3}$
Projected ( $N = 100$ )	0.145 $\pm$ .002		0.073 $\pm$ .005		$5.4 \times 10^{-8}$	$3.3 \times 10^{-3}$
Guided ( $N = 100$ )	0.152 $\pm$ .003		0.057 $\pm$ .002		$5.7 \times 10^{-2}$	$2.4 \times 10^{-3}$
<i>Ours</i>						
ULLA ( $N = 100$ )	0.031 $\pm$ .002		0.035 $\pm$ .002		$1.4 \times 10^{-7}$	$6.0 \times 10^{-11}$

maintains extremely low constraint violations (e.g., avg.  $|h| \approx 10^{-5}$  and  $|g^+| \approx 10^{-9}$ ), effectively respecting the complex geometry without the need for expensive multiple projection steps.

**Effect of landing rate  $\alpha$  and repulsion rate  $\epsilon$ .** We analyze how the landing rate  $\alpha$  and boundary repulsion rate  $\epsilon$  impact generation quality (JSD) and constraint satisfaction on the Alanine Dipeptide task with ULLA (fixing  $\epsilon = 0.05$  for  $\alpha$  ablation and  $\alpha = 50$  for  $\epsilon$  ablation). As shown in Table 2, increasing  $\alpha$  significantly reduces equality violations and improves JSD by strengthening the drift toward  $\Sigma$ . However, excessively large  $\alpha$  may introduce large discretization error, which can degrade sample quality (see Table 8 for full ablation). Similarly, Table 3 indicates that  $\epsilon$  requires a balanced choice: while too small  $\epsilon$  causes boundary “stickiness,” overly large  $\epsilon$  aggressively pushes trajectories inward, distorting the distribution and increasing inequality violations due to numerical instability. Visualizations in Figure 8 confirm that a moderate  $\epsilon$  yields the best trade-off.

## 6 CONCLUSION & FUTURE WORKS

We introduce a landing-based overdamped and underdamped Langevin process that avoids costly projections on general constrained sets described by both equality and inequality constraints, resulting in a unified constrained diffusion model framework. By leveraging the fast mixing property of underdamped Langevin dynamics, we shorten the forward trajectory (thus reducing function evaluations and memory), achieving reasonable sample quality while drastically cutting computational cost for training and sampling. A current limitation is that our sampling quality does not consistently and fully surpass projection-based methods. As a promising direction, we suggest training the score network to learn both tangential and normal components – so that normal components, assisted by our landing mechanism, actively reduce constraint violation – potentially improving sample quality without sacrificing efficiency.

540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593

#### REPRODUCIBILITY STATEMENT

We include anonymized code (training, evaluation, and data preprocessing scripts), exact hyperparameters in [Appendix E](#) to reproduce the provided results. We also report hardware specs, software on the [Appendix E](#) and all theoretical proofs are provided in the appendix for theoretical completeness.

#### ETHICS STATEMENT

This work uses public or synthetic data only, with no human subjects or personal data. The method may be applied to safety-critical domains (e.g., robotics); we caution against deployment without domain-specific evaluation and monitoring.

## REFERENCES

- 594  
595  
596 [1] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Ad-*  
597 *vances in Neural Information Processing Systems*, 33:6840–6851, 2020. 1, 7, 37, 42, 47
- 598 [2] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon,  
599 and Ben Poole. Score-based generative modeling through stochastic differential equations.  
600 *International Conference on Learning Representations*, 2021. 1
- 601  
602 [3] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil  
603 Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *arxiv preprint*  
604 *arXiv:1406.2661*, 2014. 1
- 605 [4] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat GANs on image synthesis.  
606 *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021. 1
- 607  
608 [5] Bowen Jing, Gabriele Corso, Jeffrey Chang, Regina Barzilay, and Tommi Jaakkola. Torsional  
609 diffusion for molecular conformer generation. *Advances in Neural Information Processing*  
610 *Systems*, 2022. 1
- 611 [6] Joseph L. Watson, David Juergens, Nathaniel R. Bennett, Brian L. Trippe and Jason Yim, He-  
612 len E. Eisenach, Woody Ahern, Andrew J. Borst, Robert J. Ragotte, Lukas F. Milles, Basile  
613 I. M. Wicky, Nikita Hanikel, Samuel J. Pellock, Alexis Courbet, William Sheffler, Jue Wang,  
614 Preetham Venkatesh, Isaac Sappington, Susana Vázquez Torres, Anna Lauko, Valentin De  
615 Bortoli, Emile Mathieu, Sergey Ovchinnikov, Regina Barzilay, Tommi S. Jaakkola, Frank Di-  
616 Maio, Minkyung Baek, and David Baker. De novo design of protein structure and function  
617 with RFdiffusion. *Nature*, 620:1089–1100, 2023.
- 618 [7] Kevin E. Wu, Kevin K. Yang, Rianne van den Berg, James Y. Zou, Alex X. Lu, and Ava P.  
619 Amini. Protein structure generation via folding diffusion. *arxiv preprint arXiv:2209.15611*,  
620 2022. 1
- 621  
622 [8] Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ  
623 Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion.  
624 *Robotics: Science and Systems*, 2023. 1
- 625 [9] Ralf Römer, Alexander von Rohr, and Angela P. Schoellig. Diffusion predictive control with  
626 constraints. *Conference on Learning for Dynamics and Control*, 2025.
- 627  
628 [10] Hao Ma, Sabrina Bodmer, Andrea Carron, Melanie Zeilinger, and Michael Muehlebach.  
629 Constraint-aware diffusion guidance for robotics: Real-time obstacle avoidance for au-  
630 tonomous racing. *Conference on Robot Learning*, 2025. 1
- 631 [11] Thomas Wagenaar, Simone Mancini, and Andrés Mateo-Gabín. Generative aerodynamic de-  
632 sign with diffusion probabilistic models. *arxiv preprint arXiv:2409.13328*, 2024. 1
- 633 [12] Alexander Htet Kyaw, Se Hwan Jeon, Miana Smith, and Neil Gershenfeld. Making physical  
634 objects with generative ai and robotic assembly: Considering fabrication constraints, sustain-  
635 ability, time, functionality and accessibility. *arxiv preprint arXiv:2504.19131v2*, 2025.
- 636  
637 [13] Lyle Regenwetter, Amin Heyrani Nobari, and Faez Ahmed. Deep generative models in engi-  
638 neering design: A review. *arxiv preprint arXiv:2110.10863*, 2022. 1
- 639 [14] Jacob K. Christopher, Stephen Baek, and Ferdinando Floretto. Constrained synthesis with  
640 projected diffusion models. *Advances in Neural Information Processing Systems*, 2024. 1
- 641  
642 [15] Aaron Lou and Stefano Ermon. Reflected diffusion models. *International Conference on*  
643 *Machine Learning*, 2023. 1, 2
- 644 [16] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. 2009. 1
- 645  
646 [17] Nic Fishman, Leo Klarner, Valentin De Bortoli, Emile Mathieu, and Michael Hutchinson.  
647 Diffusion models for constrained domains. *Transactions on Machine Learning Research*, 2023.  
1, 2

- 648 [18] Michael Muehlebach and Michael I. Jordan. On constraints in first-order optimization: A view  
649 from non-smooth dynamical systems. *Journal of Machine Learning Research*, 23(256):1–47,  
650 2022. 2
- 651 [19] Pierre Ablin and Gabriel Peyré. Fast and accurate optimization on the orthogonal manifold  
652 without retraction. *International Conference on Artificial Intelligence and Statistics*, 2022.  
653
- 654 [20] Sholom Schechtman, Daniil Tiapkin, Michael Muehlebach, and Eric Moulines. Orthogonal di-  
655 rections constrained gradient method: from non-linear equality constraints to Stiefel manifold.  
656 *Conference on Learning Theory*, 2023.
- 657 [21] Michael Muehlebach and Michael I. Jordan. Accelerated first-order optimization under non-  
658 linear constraints. *Mathematical Programming*, 2025. 2
- 659 [22] Emile Mathieu and Maximilian Nickel. Riemannian continuous normalizing flows. *Advances*  
660 *in Neural Information Processing Systems*, 2020. 2, 48
- 661 [23] Noam Rozen, Aditya Grover, Maximilian Nickel, and Yaron Lipman. Moser flow: Divergence-  
662 based generative modeling on manifolds. *Advances in Neural Information Processing Systems*,  
663 34:17669–17680, 2021. 48
- 664 [24] Valentin De Bortoli, Emile Mathieu, Michael Hutchinson, James Thornton, Yee Whye Teh,  
665 and Arnaud Doucet. Riemannian score-based generative modelling. *Advances in Neural In-*  
666 *formation Processing Systems*, 35:2406–2422, 2022. 6
- 667 [25] Chin-Wei Huang, Milad Aghajohari, Joey Bose, Prakash Panangaden, and Aaron C Courville.  
668 Riemannian diffusion models. *Advances in Neural Information Processing Systems*, 35:2750–  
669 2761, 2022. 6, 23, 25
- 670 [26] Ricky TQ Chen and Yaron Lipman. Flow matching on general geometries. In *International*  
671 *Conference on Learning Representations*, 2024. 2, 7, 47
- 672 [27] Yuchen Zhu, Tianrong Chen, Lingkai Kong, Evangelos A Theodorou, and Molei Tao. Trivi-  
673 alized momentum facilitates diffusion generative modeling on Lie groups. *International Con-*  
674 *ference on Learning Representations*, 2025. 2
- 675 [28] Ruth J. Williams. Reflected Brownian motion with skew symmetric data in a polyhedral do-  
676 main. *Probability Theory and Related Fields*, 75(4):459–485, 1987. 2
- 677 [29] Andrey Pilipenko. *An introduction to stochastic differential equations with reflection*. Univer-  
678 sitätsverlag Potsdam, 2014. 2
- 679 [30] Nic Fishman, Leo Klärner, Emile Mathieu, Michael Hutchinson, and Valentin De Bortoli.  
680 Metropolis sampling for constrained diffusion models. *Advances in Neural Information Pro-*  
681 *cessing Systems*, 36:62296–62331, 2023. 2
- 682 [31] Kelvin Shuangjian Zhang, Gabriel Peyré, Jalal Fadili, and Marcelo Pereyra. Wasserstein con-  
683 trol of mirror Langevin Monte Carlo. *Conference on Learning Theory*, 2020. 2
- 684 [32] Ruilin Li, Molei Tao, Santosh S. Vempala, , and Andre Wibisono. The mirror Langevin al-  
685 gorithm converges with vanishing bias. *International Conference on Algorithmic Learning*  
686 *Theory*, 2022. 2
- 687 [33] Guan-Hong Liu, Tianrong Chen, Evangelos A. Theodorou, and Molei Tao. Mirror diffusion  
688 models for constrained and watermarked generation. *Advances in Neural Information Process-*  
689 *ing Systems*, 2023. 2
- 690 [34] Zichen Liu, Wei Zhang, Christof Schütte, and Tiejun Li. Riemannian denoising diffusion  
691 probabilistic models. *arXiv preprint arXiv:2505.04338*, 2025. 2, 6, 7, 20, 37, 38, 47, 49, 52
- 692 [35] Leonid Minchenko and Sergey Stakhovskii. On relaxed constant rank regularity condition in  
693 mathematical programming. *Optimization*, 60(4):429–440, 2011. 2, 18
- 694 [36] Mathias Rousset, Gabriel Stoltz, and Tony Lelièvre. *Free energy computations: a mathemati-*  
695 *cal perspective*. World Scientific, 2010. 4, 22, 23, 26, 29

- 702 [37] Matthieu Borelle, Teodoro Alamo, Cristina Stoica, Sylvain Bertrand, and Eduardo F Camacho.  
703 Minimal gelbrich distance to uncorrelation. *IEEE Control Systems Letters*, 8:61–66, 2023. 6,  
704 41
- 705 [38] Matthias Gelbrich. On a formula for the L2-Wasserstein metric between measures on Eu-  
706 clidean and Hilbert spaces. *Mathematische Nachrichten*, 147(1):185–203, 1990. 6, 41  
707
- 708 [39] Yuqing Wang, Ye He, and Molei Tao. Evaluating the design space of diffusion-based generative  
709 models. *Advances in Neural Information Processing Systems*, 37:19307–19352, 2024. 7, 42  
710
- 711 [40] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of  
712 diffusion-based generative models. *Advances in Neural Information Processing Systems*, 35:  
713 26565–26577, 2022. 7, 42
- 714 [41] Ruqi Zhang, Qiang Liu, and Xin Tong. Sampling in constrained domains with orthogonal-  
715 space variational gradient descent. *Advances in Neural Information Processing Systems*, 35:  
716 37108–37120, 2022. 7
- 717 [42] Robert Janin. Directional derivative of the marginal function in nonlinear programming. In  
718 *Sensitivity, Stability and Parametric Analysis*, pages 110–126. Springer, 2009. 18  
719
- 720 [43] Mikhail V Solodov et al. Constraint qualifications. *Wiley Encyclopedia of Operations Research  
721 and Management Science*. Wiley, New York, 2010. 18  
722
- 723 [44] R Tyrrell Rockafellar and Roger JB Wets. *Variational analysis*. Springer, 1998. 18  
724
- 725 [45] Gilbert W Stewart. On the continuity of the generalized inverse. *SIAM Journal on Applied  
726 Mathematics*, 17(1):33–45, 1969. 18
- 727 [46] Sitan Chen, Sinho Chewi, Jerry Li, Yuanzhi Li, Adil Salim, and Anru R Zhang. Sampling is as  
728 easy as learning the score: theory for diffusion models with minimal data assumptions. *arXiv  
729 preprint arXiv:2209.11215*, 2022. 19  
730
- 731 [47] Stanislas Strasman, Sobihan Surendran, Claire Boyer, Sylvain Le Corff, Vincent Lemaire, and  
732 Antonio Ocello. Wasserstein convergence of critically damped langevin diffusions. *arXiv  
733 preprint arXiv:2511.02419*, 2025. 19
- 734 [48] Xiang Cheng, Niladri S Chatterji, Peter L Bartlett, and Michael I Jordan. Underdamped  
735 Langevin MCMC: A non-asymptotic analysis. In *Conference on Learning Theory*, pages 300–  
736 323, 2018. 19
- 737 [49] Yi-An Ma, Niladri S Chatterji, Xiang Cheng, Nicolas Flammarion, Peter L Bartlett, and  
738 Michael I Jordan. Is there an analog of Nesterov acceleration for gradient-based MCMC?  
739 *arXiv preprint, arXiv:1902.00996*, 2021. 19  
740
- 741 [50] Zichen Liu, Wei Zhang, and Tiejun Li. Improving the euclidean diffusion generation of man-  
742 ifold data by mitigating score function singularity. *arXiv preprint arXiv:2505.09922*, 2025.  
743 19
- 744 [51] Tim Dockhorn, Arash Vahdat, and Karsten Kreis. Score-based generative modeling with  
745 critically-damped Langevin diffusion. In *International Conference on Learning Representations*,  
746 2022. 19, 35  
747
- 748 [52] Gregory S Chirikjian. *Stochastic Models, Information Theory, and Lie Groups, Volume 1:  
749 Classical Results and Geometric Methods*. Springer Science & Business Media, 2009. 23  
750
- 751 [53] Shinzo Watanabe and Nobuyuki Ikeda. *Stochastic differential equations and diffusion pro-  
752 cesses*. Elsevier, 2011. 23
- 753 [54] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan,  
754 Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative  
755 style, high-performance deep learning library. *Advances in neural information processing  
systems*, 32, 2019. 47

- 756 [55] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary  
757 differential equations. *Advances in neural information processing systems*, 31, 2018. 47  
758
- 759 [56] Andreas Wächter and Lorenz T Biegler. On the implementation of an interior-point filter line-  
760 search algorithm for large-scale nonlinear programming. *Mathematical programming*, 106(1):  
761 25–57, 2006. 48
- 762 [57] Jacob K Christopher, Stephen Baek, and Nando Fioretto. Constrained synthesis with projected  
763 diffusion models. *Advances in Neural Information Processing Systems*, 37:89307–89333,  
764 2024. 48
- 765 [58] Jinhao Liang, Jacob K Christopher, Sven Koenig, and Ferdinando Fioretto. Simul-  
766 taneous multi-robot motion planning with projected diffusion models. *arXiv preprint*  
767 *arXiv:2502.03607*, 2025. 48  
768
- 769 [59] National Geophysical Data Center / World Data Service (NGDC/WDS). NCEI/WDS Global  
770 Significant Volcanic Eruptions Database, 2020. URL [https://data.nodc.noaa.gov/  
771 cgi-bin/iso?id=gov.noaa.ngdc.mgg.hazards:G10147](https://data.nodc.noaa.gov/cgi-bin/iso?id=gov.noaa.ngdc.mgg.hazards:G10147). 48  
772
- 773 [60] National Geophysical Data Center / World Data Service (NGDC/WDS). NCEI/WDS Global  
774 Significant Earthquake Database. URL [https://www.ngdc.noaa.gov/hazard/  
775 earthqk.shtml](https://www.ngdc.noaa.gov/hazard/earthqk.shtml). 48
- 776 [61] G. R. Brakenridge. Global active archive of large flood events, 2017. URL [http://  
777 floodobservatory.colorado.edu/Archives/index.html](http://floodobservatory.colorado.edu/Archives/index.html). 48  
778
- 779 [62] NASA Earth Observing System Data and Information System (EOSDIS). Active fire  
780 data, 2020. URL [https://earthdata.nasa.gov/earth-observation-data/  
781 near-real-time/firms/active-fire-data](https://earthdata.nasa.gov/earth-observation-data/near-real-time/firms/active-fire-data). Data from the Land, Atmosphere  
782 Near real-time Capability for EOS (LANCE) system. 48
- 783 [63] Greg Turk and Marc Levoy. Zippered polygon meshes from range images. In *Annual confer-  
784 ence on Computer Graphics and Interactive Techniques*, pages 311–318, 1994. 48
- 785 [64] Keenan Crane, Ulrich Pinkall, and Peter Schröder. Robust fairing via conformal curvature  
786 flow. *ACM Transactions on Graphics*, 32(4):1–10, 2013. 48  
787
- 788 [65] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric  
789 regularization for learning shapes. *arXiv preprint arXiv:2002.10099*, 2020. 48
- 790 [66] Mark James Abraham, Teemu Murtola, Roland Schulz, Szilárd Páll, Jeremy C Smith, Berk  
791 Hess, and Erik Lindahl. Gromacs: High performance molecular simulations through multi-  
792 level parallelism from laptops to supercomputers. *SoftwareX*, 1:19–25, 2015. 49  
793
- 794 [67] Giacomo Fiorin, Michael L Klein, and Jérôme Hémin. Using collective variables to drive  
795 molecular dynamics simulations. *Molecular Physics*, 111(22-23):3345–3362, 2013. 49
- 796 [68] Tony Lelièvre, Thomas Pigeon, Gabriel Stoltz, and Wei Zhang. Analyzing multimodal proba-  
797 bility measures with autoencoders. *The Journal of Physical Chemistry B*, 128(11):2607–2631,  
798 2024. 49  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809

## TABLE OF CONTENTS

810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Related Works</b>	<b>2</b>
<b>3</b>	<b>Preliminaries &amp; Notations</b>	<b>2</b>
<b>4</b>	<b>Main results</b>	<b>3</b>
4.1	Constrained Langevin Dynamics via Landing . . . . .	3
4.2	Transition Kernels for Forward / Backward Process . . . . .	5
4.3	Conditional Wasserstein Path Matching (CWPM) . . . . .	6
<b>5</b>	<b>Experiments</b>	<b>7</b>
5.1	Equality-only Scenario Tasks . . . . .	7
5.2	Mixed Scenario Tasks . . . . .	9
<b>6</b>	<b>Conclusion &amp; Future works</b>	<b>10</b>
<b>A</b>	<b>Table of Key Notation, Additional Remarks, and Algorithms</b>	<b>17</b>
<b>B</b>	<b>Constrained Langevin Dynamics</b>	<b>22</b>
B.1	Construction of OLLA . . . . .	22
B.2	Construction of ULLA . . . . .	26
B.3	Properties and Backward processes of Constrained Langevin Dynamics with Landing . . . . .	30
B.4	Discretization of Constrained Langevin Dynamics . . . . .	32
<b>C</b>	<b>DT-ELBO on Riemannian manifold</b>	<b>37</b>
<b>D</b>	<b>Conditional Wasserstein Path Matching (CWPM)</b>	<b>41</b>
<b>E</b>	<b>Experiment settings and Supplementary Results</b>	<b>47</b>
E.1	Description of baseline algorithms . . . . .	47
E.2	Experiment settings and descriptions . . . . .	48
E.3	Generated samples from tasks. . . . .	52
E.4	Supplementary Results - Effect of Hyperparameters $\alpha$ and $\epsilon$ . . . . .	54

## A TABLE OF KEY NOTATION, ADDITIONAL REMARKS, AND ALGORITHMS

Table 5: Table of Key Notations

Symbol	Definition	Descriptions
$h$	$h(x) = [h_1(x), \dots, h_m(x)]^T$	Equality constraints
$g$	$g(x) = [g_1(x), \dots, g_l(x)]^T$	Inequality constraints
$\Sigma$	$\{x \in \mathbb{R}^d \mid h(x) = 0, g(x) \leq 0\}$	Constraint manifold
$I_x$	$\{i \in [l] \mid g_i(x) \geq 0\} = \{i_1, \dots, i_{ I_x }\}$	Active index set of inequalities
$g_{I_x}$	$g_{I_x}(x) = [g_{i_1}(x), \dots, g_{i_{ I_x }}(x)]^T$	Active inequality constraints
$J(x)$	$\{h(x)^T, g_{i_1}(x) + \epsilon, \dots, g_{i_{ I_x }}(x) + \epsilon\}^T$	Constraint-correction vector
$\Pi(x)$	$I - \nabla J(x)^T G(x)^\dagger \nabla J(x)$	Orthogonal projector onto $T_x \Sigma$
$T_x \Sigma$	$\{p \in \mathbb{R}^d \mid \nabla h(x)v = 0, \nabla g_{I_x}(x)v = 0\}$	Tangent space of $\Sigma$ at $x$
$T^* \Sigma$	$\{(x, p) \in \mathbb{R}^{2d} \mid x \in \Sigma, p \in T_x \Sigma\} \simeq T \Sigma$	Cotangent bundle of $\Sigma$
$\nabla_{\Sigma} f$	$\Pi(x) \nabla f(x)$	Intrinsic gradient on $\Sigma$
$\text{div}_{\Sigma} X$	$\text{Tr}(\Pi(x) \nabla X(x))$	Intrinsic divergence on $\Sigma$
$d\sigma_{\Sigma}$	Surface (Hausdorff) measure of $\Sigma$	Natural measure on $\Sigma$
$d\sigma_{T^* \Sigma}$	Liouville measure of $T^* \Sigma$	Natural measure on $T^* \Sigma$
$G(x)$	$\nabla J(x) \nabla J(x)^T$	Gram matrix, full rank assumed
$\epsilon$	Boundary repulsion rate	Controls effect of repulsion.
$\alpha$	Landing rate	Controls constraint decay
$\gamma$	Friction coefficient	Used in ULLA, ULLA-P
$\rho_{\Sigma}$	Target (stationary) density on $\Sigma$	Proportional to $\exp(-f) d\sigma_{\Sigma}$
$\text{KL}^{\Sigma}(\rho \parallel \pi)$	$\int_{\Sigma} \rho \ln \frac{\rho}{\pi} d\sigma_{\Sigma}$	KL-divergence on $\Sigma$
$T, N$	Continuous and discrete terminal time	Relationship: $T = N \Delta t$
$\sigma(t), \sigma_k$	$\sigma_{\min} + \frac{t}{T}(\sigma_{\max} - \sigma_{\min}), \sigma_k = \sigma(k \Delta t)$	Noise schedule function
$q_t, p_t^{\theta}$	Continuous time marginal densities at $t$	Forward $q_t$ , Backward $p_t^{\theta}$
$q_k, p_k^{\theta}$	Discrete time marginal densities at $k$	Forward $q_k$ , Backward $p_k^{\theta}$
$p_N, \rho(\cdot   x)$	Prior distribution of $x$ and $p$ ( $p_N$ varies)	$\rho(\cdot   x) \sim \Pi(x)\zeta, \zeta \sim \mathcal{N}(0, I)$
$\tilde{p}_k^{\text{fwd}}$	$\Pi(x_k) \left( \frac{x_k - x_{k-1}}{\sigma_{k-1}^2 \Delta t} \right)$	Forward approximated momentum
$\tilde{p}_k^{\text{bwd}}$	$\Pi(x_{k+1}) \left( \frac{x_{k+2} - x_{k+1}}{\sigma_{k+2}^2 \Delta t} \right)$	Backward approximated momentum

**Remark 2** (Comments on relaxed Constant Rank Constraint Qualification (rCRCQ)). In this remark, we further clarify the definition of the relaxed Constant Rank Constraint Qualification (rCRCQ) and its relationship with other constraint qualifications.

We first recall the definitions of the Linear Independence Constraint Qualification (LICQ), Constant Rank Constraint Qualification (CRCQ) [42], and its relaxed version (rCRCQ) [35].

**Definition A.1** (LICQ, CRCQ, and rCRCQ; [43]). Let  $\Sigma := \{x \in \mathbb{R}^d \mid h(x) = 0, g(x) \leq 0\}$  be the feasible set, and denote  $I_x = \{i \in [l] \mid g_i(x) \geq 0\}$  to be the active index set of inequalities.

- **LICQ** [44]: LICQ holds at  $x \in \Sigma$  if the set  $\{\nabla h_i(x)\}_{i=1}^m \cup \{\nabla g_j(x)\}_{j \in I_x}$  is linearly independent.
- **CRCQ** [42]: CRCQ holds at  $x \in \Sigma$  if there exists a neighborhood  $U \subset \mathbb{R}^d$  of  $x$  such that for any subsets of indices  $I \subset [m]$  and  $J \subset I_x$ , the family of gradients  $\{\nabla h_i(y)\}_{i \in I} \cup \{\nabla g_j(y)\}_{j \in J}$  has a constant rank for all  $y \in U$ .
- **rCRCQ** [35]: rCRCQ holds at  $x \in \Sigma$  if there exists a neighborhood  $U \subset \mathbb{R}^d$  of  $x$  such that for any subset of active inequalities  $J \subset I_x$ , the family of gradients  $\{\nabla h_i(y)\}_{i=1}^m \cup \{\nabla g_j(y)\}_{j \in J}$  has a constant rank for all  $y \in U$ .

The core reason for assuming rCRCQ lies in the stability of the SDE coefficients. It is a fundamental result in matrix analysis [45] that the Moore-Penrose pseudo-inverse  $A(x)^\dagger$  is continuous at a point  $x_0$  if and only if the rank of  $A(x)$  is constant in a neighborhood of  $x_0$ . By assuming rCRCQ, we guarantee that the Jacobian  $\nabla J(x)$  maintains locally constant rank (even as the active set changes across strata), which ensures that the pseudo-inverse  $G(x)^\dagger$  and the resulting projection operator  $\Pi(x)$  are continuous and well-defined. Therefore, this guarantees the drift vector and diffusion matrix of the OLLA and ULLA dynamics to be well defined.

**Hierarchy of Constraint Qualifications.** We remark that, from the variational analysis and optimization literature (e.g., [43]), rCRCQ is a strictly weaker condition than CRCQ, and CRCQ is strictly weaker than LICQ, therefore, their logical implication is as follows:

$$\text{LICQ} \implies \text{CRCQ} \implies \text{rCRCQ}.$$

In particular, rCRCQ can relax the gradient degeneracy problem appearing in LICQ.

To illustrate a case where LICQ fails due to gradient degeneracy while rCRCQ holds, consider a feasible set  $\Sigma \subset \mathbb{R}^3$  representing the  $z$ -axis. It is defined by two equality constraints and one redundant inequality constraint with a nonlinear term:

$$\begin{aligned} h_1(x) &= x_1 = 0, & h_2(x) &= x_2 = 0, \\ g(x) &= x_1 + x_2 + x_1^2 \leq 0. \end{aligned}$$

On the manifold  $\Sigma$  (where  $x_1 = x_2 = 0$ ), the inequality is active since  $g(0) = 0$ .

- **LICQ fails:** The gradients at the origin  $x = 0$  are  $\nabla h_1 = (1, 0, 0)^T$ ,  $\nabla h_2 = (0, 1, 0)^T$ , and  $\nabla g = (1, 1, 0)^T$ . We observe that  $\nabla g = \nabla h_1 + \nabla h_2$ , meaning the gradients are linearly dependent. Thus, the Gram matrix is singular, and LICQ is violated.
- **rCRCQ holds:** Now consider the Jacobian matrix of the active constraints for an arbitrary point  $x \in \mathbb{R}^3$ :

$$J(x) = \begin{bmatrix} \nabla h_1(x)^T \\ \nabla h_2(x)^T \\ \nabla g(x)^T \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 + 2x_1 & 1 & 0 \end{bmatrix}.$$

Regardless of the location  $x$ , the rank of  $J(x)$  is constant and equal to two in the entire neighborhood, satisfying rCRCQ and ensuring that the projection operator  $\Pi(x)$  via the pseudo-inverse  $G(x)^\dagger$  remains well-defined and continuous.

**Extended Usage in Our Framework.** While the standard definition of rCRCQ is checking the condition at a point “ $x \in \Sigma$ ”, we appropriately extend this usage in our diffusion model context.

In particular, since our landing-based discretized sampling algorithms (OLLA, ULLA) involves noise that may push particles slightly off the manifold, we implicitly assume that this constant rank property extends to an sufficiently large neighborhood of  $\Sigma$  which contains all discretized samples  $\{X_k\}_{k=0}^N$ , or to the entire ambient space  $\mathbb{R}^d$ . This ensures that the projection operator  $\Pi(x)$  and the drift terms are well-defined not just on  $\Sigma$ , but in the surrounding ambient space  $\mathbb{R}^d$  where the landing mechanism operates.

**Remark 3** (Error decomposition and probable benefit of ULLA). Recent theoretical progress on diffusion models (e.g., [46, 47]) suggests that the total generation error can be naturally decomposed into three distinct components. In the 2-Wasserstein distance, this can be viewed as:

$$W_2(q_0, p_0^\theta) \leq \underbrace{\mathcal{E}_{\text{mix}}}_{\text{Mixing}} + \underbrace{\mathcal{E}_{\text{disc}}}_{\text{Discretization}} + \underbrace{\mathcal{E}_{\text{score}}}_{\text{Score estimation}}$$

- 1. Discretization error ( $\mathcal{E}_{\text{disc}}$ ) & Mixing error ( $\mathcal{E}_{\text{mix}}$ ):** Regarding discretization, our ULLA implementation employs a memory-efficient first-order splitting scheme; thus, both ULLA and the baseline OLLA share the same convergence order with respect to the step size. However, ULLA gains a significant advantage in the *mixing error* due to the ballistic behavior of underdamped dynamics, which theoretically accelerates convergence to  $\mathcal{O}(\sqrt{d}/\epsilon)$  compared to the diffusive  $\mathcal{O}(d/\epsilon^2)$  of overdamped dynamics [48, 49]. This allows for a significantly smaller trajectory length  $N$  to reach the stationary prior, thereby reducing the computational cost for training and storage.
- 2. Score estimation error ( $\mathcal{E}_{\text{score}}$ ):** Employing a constrained forward process with the proposed landing mechanism allows the model to faithfully capture the intrinsic geometry of  $\Sigma$ . Crucially, because the landing mechanism analytically handles the ill-conditioned normal component, the score network  $s_\theta^t$  is only required to learn the smoother tangential component  $\Pi(x)s_{\text{true}}^t$  [50]. Adopting underdamped dynamics introduces a trade-off: learning on the extended phase space potentially increases regression complexity compared to position-only models. However, since empirical data distributions are usually supported on some data manifold  $\Sigma_{\text{data}} \subset \Sigma$ , standard overdamped models suffer from score singularities where  $\|s_{\text{true}}^t\|_2 \propto \mathcal{O}(1/t)$  near  $t = 0$  [50]. In contrast, as highlighted in Dockhorn et al. [51], underdamped dynamics yield a smoother training objective that bypasses this singularity problem due to the existence of momentum variable.

Figure 4 provides empirical evidence of this effect on the volcano experiment. The underdamped model exhibits Jacobian norms that are several orders of magnitude smaller across all times and, in particular, does *not* show the sharp blow-up near  $t \approx 0$  that appears in the overdamped case. This suggests that ULLA provides a numerically better-conditioned score regression problem, which can potentially reduce  $\mathcal{E}_{\text{score}}$  in practice.

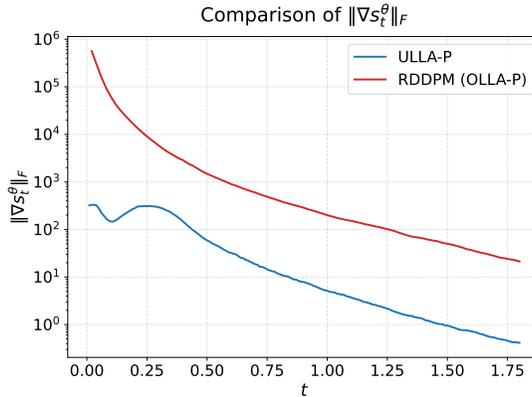


Figure 4: Comparison of the Frobenius norm of the score Jacobian  $\|\nabla s_t^\theta\|_F$  over time on the volcano experiment. The overdamped RDDPM (OLLA-P) baseline (red) exhibits very large Jacobian norms and a pronounced singular behavior as  $t \rightarrow 0$ , while the underdamped ULLA-P sampler (blue) remains several orders of magnitude smaller and shows no blow-up near  $t \approx 0$ . Jacobian is taken over to position  $x$  for the overdamped and to momentum  $p$  for the underdamped.

**Algorithm 1** Full Diffusion Pipeline for OLLA / OLLA-P (=RDDPM [34])

---

1026  
1027  
1028 1: **Input:** Data distribution  $q_{\text{data}}$ , initial score network  $s_k^\theta(x)$ , number of steps  $N$ , terminal step  $N$ ,  
1029 landing rate  $\alpha$ , boundary repulsion rate  $\epsilon$ , constraints  $h, g$ .  
1030 2: **Options:**  $\text{mode} \in \{\text{OLLA}, \text{OLLA-P}\}$ ,  $\text{use\_curvature} \in \{\text{True}, \text{False}\}$   
1031 3: **Output:** Trained score network  $s_k^\theta(x)$ , generated sample  $x_0$

---

1032 **Part 1: Forward Process (Noising)** ▷ Run Forward Process per  $l_f$  iterations  
1033 4: Sample  $x_0 \sim q_{\text{data}} = q_0$   
1034 5: **for**  $k \in \{0, \dots, N-1\}$  **do**  
1035 6:   Compute  $\nabla f(x_k), J(x_k), \nabla J(x_k), G(x_k)^\dagger, \Pi(x_k)$   
1036 7:    $\bar{\mu}_k^o(x_k) \leftarrow x_k - \frac{1}{2}\sigma_k^2 \Delta t \Pi(x_k) \nabla f(x_k)$  ▷ Prior drift term  
1037 8:   **if**  $\text{mode} = \text{OLLA-P}$  **then** ▷ Projection-based noising  
1038 9:      $x_{k+1} \leftarrow \text{Proj}_\Sigma(\bar{\mu}_k^o(x_k) + \sigma_k \sqrt{\Delta t} \Pi(x_k) \zeta_k), \quad \zeta_k \sim \mathcal{N}(0, I_d)$   
1039 10:   **else** ▷ Landing-based noising (OLLA)  
1040 11:      $\mathcal{H}(x_k) \leftarrow 0$   
1041 12:     **if**  $\text{use\_curvature}$  **then**  
1042 13:        $\text{Tr} \leftarrow [\text{Tr}(\Pi \nabla^2 J_1), \dots, \text{Tr}(\Pi \nabla^2 J_{m+|I_{x_k}|})]^T$   
1043 14:        $\mathcal{H}(x_k) \leftarrow \nabla J(x_k)^T G(x_k)^\dagger \text{Tr}$   
1044 15:     **end if**  
1045 16:      $L_k(x_k) \leftarrow -\alpha \sigma_k^2 \Delta t \nabla J(x_k)^T G(x_k)^{-1} J(x_k)$  ▷ Landing term  
1046 17:      $\kappa_k^o(x_k) \leftarrow \frac{1}{2} \sigma_k^2 \Delta t \mathcal{H}(x_k)$  ▷ Curvature term  
1047 18:      $x_{k+1} \leftarrow \bar{\mu}_k^o(x_k) + L_k(x_k) + \kappa_k^o(x_k) + \sigma_k \sqrt{\Delta t} \Pi(x_k) \zeta_k, \quad \zeta_k \sim \mathcal{N}(0, I_d)$   
1048 19:     **end if**  
1049 20: **end for**  
1050 21:  $x_N \leftarrow \text{Proj}_\Sigma(x_N)$  ▷ Terminal projection by Newton’s method  
1051 22: Store trajectory  $\{x_k\}_{k=0}^N$

---

1052 **Part 2: Score Network Training**  
1053 23:  $L_{\text{CWPM}}^{\text{over}}(\theta) \leftarrow \sum_{k=0}^{N-1} \frac{\|\Pi(x_{k+1})(x_k - \mu_{k+1}^o(x_{k+1}))\|^2}{2\sigma_{k+1}^2 \Delta t}$   
1054 24: Update network parameters:  $\theta \leftarrow \theta - \eta \nabla_\theta L_{\text{CWPM}}^{\text{over}}(\theta)$  ▷  $\eta$  is the learning rate

---

1057 **Part 3: Backward Process (Sampling)**  
1058 25: Sample  $x_N \sim p_N$  (prior)  
1059 26: **for**  $k \in \{N, \dots, 1\}$  **do**  
1060 27:   Compute  $\nabla f(x_k), J(x_k), \nabla J(x_k), G(x_k)^\dagger, \Pi(x_k)$   
1061 28:    $\mu_k^o(x_k) \leftarrow x_k + \frac{1}{2}\sigma_k^2 \Delta t \Pi(x_k) [\nabla f(x_k) + s_k^\theta(x_k)]$   
1062 29:   **if**  $\text{mode} = \text{OLLA-P}$  **then** ▷ Projection-based variant  
1063 30:      $x_{k-1} \leftarrow \text{Proj}_\Sigma(\mu_k^o(x_k) + \sigma_k \sqrt{\Delta t} \Pi(x_k) \zeta_k)$   
1064 31:   **else** ▷ Landing-based variant (OLLA)  
1065 32:      $\mathcal{H}(x_k) \leftarrow 0$   
1066 33:     **if**  $\text{use\_curvature}$  **then**  
1067 34:        $\text{Tr} \leftarrow [\text{Tr}(\Pi \nabla^2 J_1), \dots, \text{Tr}(\Pi \nabla^2 J_{m+|I_{x_k}|})]^T$   
1068 35:        $\mathcal{H}(x_k) \leftarrow \nabla J(x_k)^T G(x_k)^\dagger \text{Tr}$   
1069 36:     **end if**  
1070 37:      $L_k(x_k) \leftarrow -\alpha \sigma_k^2 \Delta t \nabla J(x_k)^T G(x_k)^\dagger J(x_k)$   
1071 38:      $\kappa_k^o(x_k) \leftarrow \frac{1}{2} \sigma_k^2 \Delta t \mathcal{H}(x_k)$   
1072 39:      $x_{k-1} \leftarrow \mu_k^o(x_k) + L_k(x_k) + \kappa_k^o(x_k) + \sigma_k \sqrt{\Delta t} \Pi(x_k) \zeta_k$   
1073 40:     **end if**  
1074 41: **end for**  
1075 42:  $x_0 \leftarrow \text{Proj}_\Sigma(x_0)$  ▷ Terminal projection by Newton’s method  
1076 43: **return**  $x_0$

---

1077  
1078  
1079

**Algorithm 2** Full Diffusion Pipeline for ULLA / ULLA-P

---

1080  
1081  
1082 1: **Input:** Data distribution  $q_{\text{data}}$ , initial score network  $s_k^\theta(x, p)$ , number of steps  $N$ , terminal time  
1083  $T$ , landing rate  $\alpha$ , boundary repulsion rate  $\epsilon$ , friction  $\gamma$ , constraints  $h, g$ .  
1084 2: **Options:** `mode`  $\in \{\text{ULLA}, \text{ULLA-P}\}$ , `use_curvature`  $\in \{\text{True}, \text{False}\}$   
1085 3: **Output:** Trained score network  $s_k^\theta(x, p)$ , generated sample  $x_0$

---

1086 **Part 1: Forward Process (Noising)** ▷ Run Forward Process per  $l_f$  iterations  
1087 4: Sample  $x_0 \sim q_{\text{data}}$ ,  $p_0 \sim \mathcal{N}(0, I_d)$  and set  $\tilde{p}_0 \leftarrow \Pi(x_0)p_0$   
1088 5:  $x_{-1} \leftarrow x_0 - \sigma_{-1}^2 \Delta t \tilde{p}_0$  ▷ Create pseudo-point for first momentum  
1089 6: **for**  $k \in \{0, \dots, N-1\}$  **do**  
1090 7: Compute  $\nabla f(x_k), J(x_k), \nabla J(x_k), G(x_k)^\dagger, \Pi(x_k)$   
1091 8:  $\tilde{p}_k^{\text{fwd}} \leftarrow \Pi(x_k) \left( \frac{x_k - x_{k-1}}{\sigma_{k-1}^2 \Delta t} \right)$  ▷ Approximate momentum from positions  
1092 9:  $a_k \leftarrow e^{-\gamma \sigma_k^2 \Delta t}$   
1093 10:  $\bar{\mu}_k^u(x_k, \tilde{p}_k^{\text{fwd}}) \leftarrow x_k + \sigma_k^2 \Delta t \Pi(x_k) [a_k \tilde{p}_k^{\text{fwd}} - \sigma_k^2 \Delta t \nabla f(x_k)]$  ▷ Prior drift term  
1094 11: **if** `mode` = ULLA-P **then** ▷ Projection-based noising  
1095 12:  $x_{k+1} \leftarrow \text{Proj}_\Sigma(\bar{\mu}_k^u(x_k, \tilde{p}_k^{\text{fwd}}) + \sigma_k^2 \Delta t \sqrt{1 - a_k^2} \Pi(x_k) \zeta_k), \quad \zeta_k \sim \mathcal{N}(0, I_d)$   
1096 13: **else** ▷ Landing-based noising (ULLA)  
1097 14:  $\mathcal{H}_1(x_k, \tilde{p}_k^{\text{fwd}}), \mathcal{H}_2(x_k, \tilde{p}_k^{\text{fwd}}) \leftarrow 0, 0$   
1098 15: **if** `use_curvature` **then**  
1099 16: Compute  $\mathcal{H}_1, \mathcal{H}_2$  using  $x_k, \tilde{p}_k^{\text{fwd}}$   
1100 17: **end if**  
1101 18:  $L_k(x_k) \leftarrow -\alpha \sigma_k^2 \Delta t \nabla J(x_k)^T G(x_k)^\dagger J(x_k)$  ▷ Landing term  
1102 19:  $\kappa_{k, \text{fwd}}^U(x_k, \tilde{p}_k^{\text{fwd}}) \leftarrow -\sigma_k^4 \Delta t^2 \nabla J(x_k)^T G^\dagger(x_k) [\mathcal{H}_1 - \alpha \mathcal{H}_2]$  ▷ Curvature term  
1103 20:  $x_{k+1} \leftarrow \bar{\mu}_k^u(x_k, \tilde{p}_k^{\text{fwd}}) + L_k(x_k) + \kappa_{k, \text{fwd}}^U(x_k, \tilde{p}_k^{\text{fwd}}) + \sigma_k^2 \Delta t \sqrt{1 - a_k^2} \Pi(x_k) \zeta_k$   
1104 21: **end if**  
1105 22: **end for**  
1106 23:  $x_N \leftarrow \text{Proj}_\Sigma(x_N)$  ▷ Terminal projection by Newton’s method  
1107 24: Store trajectory  $\{x_k\}_{k=0}^N$

---

1109 **Part 2: Score Network Training**  
1110 25:  $L_{\text{CWPM}}^{\text{under}}(\theta) \leftarrow \sum_{k=0}^{N-1} \frac{\|\Pi(x_{k+1})(x_k - \mu_{k+1}^u(x_{k+1}, x_{k+2}))\|^2}{2\sigma_{k+1}^4 \Delta t^2 (1 - a_{k+1}^2)}$   
1111 26: Update network parameters:  $\theta \leftarrow \theta - \eta \nabla_\theta L_{\text{CWPM}}^{\text{under}}(\theta)$  ▷  $\eta$  is the learning rate

---

1113 **Part 3: Backward Process (Sampling)**  
1114 27: Sample  $x_N \sim p_N$  (prior),  $p_N \sim \mathcal{N}(0, I_d)$ . Set  $\tilde{p}_N \leftarrow \Pi(x_N)p_N$ .  
1115 28:  $x_{N+1} \leftarrow x_N + \sigma_N^2 \Delta t \tilde{p}_N$  Comment Create pseudo-point for terminal momentum  
1116 29: **for**  $k \in \{N, \dots, 1\}$  **do**  
1117 30: Compute  $J(x_k), \nabla J(x_k), G(x_k)^\dagger, \Pi(x_k)$   
1118 31:  $\tilde{p}_k \leftarrow \Pi(x_k) \left( \frac{x_{k+1} - x_k}{\sigma_{k+1}^2 \Delta t} \right)$   
1119 32:  $a_k \leftarrow e^{-\gamma \sigma_k^2 \Delta t}$   
1120 33:  $\mu_k^u(x_k, \tilde{p}_k^{\text{bwd}}) \leftarrow x_k - \sigma_k^2 \Delta t \Pi(x_k) [a_k \tilde{p}_k^{\text{bwd}} + \sigma_k^2 \Delta t (\nabla f(x_k) + s_k^\theta(x_k, \tilde{p}_k^{\text{bwd}}))]$   
1121 34: **if** `mode` = ULLA-P **then** ▷ Projection-based variant  
1122 35:  $x_{k-1} \leftarrow \text{Proj}_\Sigma(\mu_k^u(x_k, \tilde{p}_k^{\text{bwd}}) + \sigma_k \sqrt{\Delta t (1 - a_k^2)} \Pi(x_k) \zeta_k)$   
1123 36: **else** ▷ Landing-based variant (ULLA)  
1124 37:  $\mathcal{H}_1(x_k, \tilde{p}_k^{\text{bwd}}), \mathcal{H}_2(x_k, \tilde{p}_k^{\text{bwd}}) \leftarrow 0, 0$   
1125 38: **if** `use_curvature` **then**  
1126 39: Compute  $\mathcal{H}_1, \mathcal{H}_2$  using  $x_k, \tilde{p}_k^{\text{bwd}}$   
1127 40: **end if**  
1128 41:  $L_k(x_k) \leftarrow -\alpha \sigma_k^2 \Delta t \nabla J(x_k)^T G(x_k)^\dagger J(x_k)$   
1129 42:  $\kappa_k^U(x_k, \tilde{p}_k^{\text{bwd}}) \leftarrow -\sigma_k^4 \Delta t^2 \nabla J(x_k)^T G^\dagger(x_k) [\mathcal{H}_1 + \alpha \mathcal{H}_2]$   
1130 43:  $x_{k-1} \leftarrow \mu_k^u(x_k, \tilde{p}_k^{\text{bwd}}) + L_k(x_k) + \kappa_k^U(x_k, \tilde{p}_k^{\text{bwd}}) + \sigma_k \sqrt{\Delta t (1 - a_k^2)} \Pi(x_k) \zeta_k$   
1131 44: **end if**  
1132 45: **end for**  
1133 46:  $x_0 \leftarrow \text{Proj}_\Sigma(x_0)$  ▷ Terminal projection by Newton’s method  
47: **return**  $x_0$

---

## B CONSTRAINED LANGEVIN DYNAMICS

In this section, we review the constrained Langevin dynamics and introduce their landing versions.

### B.1 CONSTRUCTION OF OLLA

**Notations and Background for overdamped setup.** We consider the constrained set  $\Sigma := \{x \in \mathbb{R}^d \mid h(x) = 0, g(x) \leq 0\}$ , assumed to be a stratified manifold  $\mathbb{R}^d$  with rCRCQ satisfied. We define the stacked active constraint map and its Jacobian as

$$J(x) := [h(x), g_{I_x}(x) + \epsilon] \in \mathbb{R}^{m+|I_x|}, \quad \nabla J(x) \in \mathbb{R}^{(m+|I_x|) \times d}$$

where  $I_x$  denotes the set of active inequality constraints, i.e.,  $I_x := \{i \in [l] \mid g_i(x) \geq 0\}$ . Denote the Gram matrix  $G(x) := \nabla J(x) \nabla J(x)^T \in \mathbb{R}^{(m+|I_x|) \times (m+|I_x|)}$ . The orthogonal projector onto the tangent space of  $T_x \Sigma := \{p \in \mathbb{R}^d \mid \nabla J(x)p = 0\}$  is given by  $\Pi(x) = I - \nabla J(x)^T G(x)^\dagger \nabla J(x)$ . On this manifold  $\Sigma$ , all intrinsic differential operators are defined via the projector  $\Pi$ . For a smooth scalar function  $\phi$  and smooth vector field  $X$  on  $\Sigma$ , we have

$$\nabla_\Sigma \phi := \Pi \nabla \phi, \quad \operatorname{div}_\Sigma(X) = \operatorname{Tr}(\Pi \nabla X)$$

and the Laplace-Betrami operator is  $\Delta_\Sigma \phi := \operatorname{div}_\Sigma(\nabla_\Sigma \phi)$ , where  $\Delta$  denotes ambient Euclidean gradient or Jacobian. For comprehensive backgrounds on constrained overdamped Langevin dynamics, see Chapter 3.2 in Rousset et al. [36].

**Proposition B.1** (Construction of OLLA). *Consider the following Lagrangian-form constrained overdamped Langevin dynamics:*

$$dX_t = -\frac{1}{2}\sigma(t)^2 \nabla f(X_t) dt + \sigma(t) \circ dW_t + \nabla J(X_t)^T d\lambda_t \quad (3)$$

where  $d\lambda_t$  is the adapted process such that  $dJ(X_t) = -\alpha\sigma(t)^2 J(X_t)$ . The explicit minimum norm solution of  $d\lambda_t$  is given by

$$d\lambda_t = G^\dagger(X_t) \left[ -\alpha\sigma(t)^2 J(X_t) dt + \frac{1}{2}\sigma(t)^2 \nabla J(X_t) \nabla f(X_t) dt - \sigma(t) \nabla J(X_t) \circ dW_t \right],$$

with  $G(X_t) := \nabla J(X_t) \nabla J(X_t)^T$  defined as the Gram matrix. Therefore, the closed form SDE of (3) is as follows:

$$dX_t = - \left[ \frac{\sigma(t)^2}{2} \Pi(X_t) \nabla f(X_t) + \alpha\sigma(t)^2 \nabla J(X_t)^T G^\dagger(X_t) J(X_t) \right] dt + \frac{\sigma(t)^2}{2} \mathcal{H}(X_t) dt + \sigma(t) \Pi(X_t) dW_t,$$

where  $\mathcal{H}$  is the mean curvature correction term defined as

$$\mathcal{H}(x) := -\nabla J(x)^T G^\dagger(x) \left[ \operatorname{Tr}(\nabla^2 J_1(x) \Pi(x)), \dots, \operatorname{Tr}(\nabla^2 J_{m+|I_x|}(x) \Pi(x)) \right]^T.$$

*Proof.* From the Stratonovich chain rule, it holds that

$$-\alpha\sigma(t)^2 J(X_t) dt = \nabla J(X_t) \circ dX_t = -\frac{1}{2}\sigma(t)^2 \nabla J(X_t) \nabla f(X_t) dt + \sigma(t) \nabla J(X_t) \circ dW_t + G(X_t) d\lambda_t$$

Among the many solutions  $d\lambda_t$  satisfying the above equation, we choose the (unique) minimum norm solution of  $d\lambda_t$  process:

$$d\lambda_t = G^\dagger(X_t) \left[ -\alpha\sigma(t)^2 J(X_t) dt + \frac{1}{2}\sigma(t)^2 \nabla J(X_t) \nabla f(X_t) dt - \sigma(t) \nabla J(X_t) \circ dW_t \right].$$

We remark that  $\nabla J(X_t)^T d\lambda_t$  is unique regardless of the choice of solution  $d\lambda_t$ . Substituting back to the SDE (3) gives the following Stratonovich version of the unique closed-form SDE:

$$dX_t = - \left[ \frac{1}{2}\sigma(t)^2 \Pi(X_t) \nabla f(X_t) + \alpha\sigma(t)^2 \nabla J(X_t)^T G^\dagger(X_t) J(X_t) \right] dt + \sigma(t) \Pi(X_t) \circ dW_t.$$

To recover the Itô version of the closed-form SDE, we observe that the Itô-Stratonovich correction term coincides with the mean curvature term of a stratum  $\Sigma_{I_x} := \{x \in \mathbb{R}^d \mid J(x) = 0\}$  and its representation is given by

$$\frac{1}{2} \nabla (\sigma(t)\Pi) (\sigma(t)\Pi) = \frac{\sigma(t)^2}{2} \nabla (\Pi)\Pi = \frac{\sigma(t)^2}{2} \sum_{k=1}^d \nabla (\Pi_k)\Pi_k.$$

From the same tensor-calculus technique of Equation 3.46 in Rousset et al. [36], we observe that  $(\nabla \Pi)\Pi$  is given by

$$\nabla \Pi(x)\Pi(x) = -\nabla J(x)^T G^\dagger(x) [\text{Tr}(\nabla^2 J_1(x)\Pi(x)), \dots, \text{Tr}(\nabla^2 J_{m+|I_x|}(x)\Pi(x))]^T.$$

Therefore, this gives the following Itô version of the closed-form SDE:

$$dX_t = - \left[ \frac{\sigma(t)^2}{2} \Pi(X_t) \nabla f(X_t) + \alpha \sigma(t)^2 \nabla J(X_t)^T G^\dagger(X_t) J(X_t) \right] dt + \frac{\sigma(t)^2}{2} \mathcal{H}(X_t) dt + \sigma(t) \Pi(X_t) dW_t.$$

□

**Theorem B.1** (Fokker-Planck equation [52, 25] and the generator [53] on Riemannian manifold).  
Let  $Z_t \in \Sigma$  be a stochastic process following the SDE:

$$dZ_t = V_0 dt + \sum_{k=1}^d V_k \circ dB_t^k,$$

where  $V_0, V_k$  are smooth vector fields on  $\Sigma$  for each  $k \in [d]$  and  $B_t^k$  are  $k$ th components of Brownian motion  $B_t$ . Then, the law  $\rho_t$  of the stochastic process  $Z_t$  satisfies the following Fokker-Planck equation:

$$\partial_t \rho_t = -\text{div}_\Sigma(\rho_t V_0) + \frac{1}{2} \sum_{k=1}^d \text{div}_\Sigma(\text{div}_\Sigma(\rho_t V_k) V_k).$$

Also, the generator of  $\mathcal{L}$  of the corresponding SDE is provided as

$$\mathcal{L}\phi = V_0\phi + \frac{1}{2} \sum_{k=1}^d V_k(V_k\phi)$$

for any smooth function  $\phi$  on  $\Sigma$ .

**Lemma B.1** (Boundary condition of OLLA). Assuming  $X_0 \in \Sigma$ , OLLA (3) satisfies the following boundary condition on  $\partial\Sigma$  and property for  $t \geq 0$ :

$$(1) \langle J_t(x), n(x) \rangle = 0 \quad \text{a.e. on } \partial\Sigma, \quad (2) X_t \in \Sigma \quad \text{a.s.}$$

where  $J_t(x)$  is the probability current density defined by  $\partial_t \rho_t = -\text{div}_\Sigma(J_t)$  and  $n(x)$  is the outward unit normal vector on  $\partial\Sigma$ .

*Proof.* First, we show that  $\mathbb{P}(g_k(X_t) \leq 0) = 1$  for  $t \geq 0$  and  $k \in [l]$ . To show this, we define a convex smooth violation penalty function  $\Psi_\delta^k(x) : \mathbb{R}^d \rightarrow \mathbb{R}$  as follows:

$$\Psi_\delta^k(x) := \phi_\delta(g_k(x)), \quad \phi_\delta(r) := \begin{cases} \frac{r^2}{2\delta}, & 0 \leq r \leq \delta \\ r - \delta/2, & r \geq \delta \\ 0 & r < 0. \end{cases}$$

Then,  $\phi_\delta$  is convex,  $C^1$ , and satisfies

$$\phi_\delta \downarrow (r)_+, \quad \phi_\delta'(r) \rightarrow \mathbb{1}_{\{r>0\}}, \quad \text{as } \delta \downarrow 0$$

with  $(r)^+ := \max\{r, 0\}$ . Now, we observe that, on  $\{g_k \geq 0\}$ , the Stratonovich chain rule (as in Lemma B.4) gives

$$dg_k(X_t) = -\alpha \sigma(t)^2 (g_k(X_t) + \epsilon) dt.$$

Therefore, applying Itô's lemma on  $\phi_\delta(g_i(X_t))$  gives

$$\begin{aligned} d\phi_\delta(g_k(X_t)) &= \phi'_\delta(g_k(X_t))dg_k(X_t) + \frac{1}{2}\phi''_\delta(g_k(X_t))\underbrace{d\langle g_k(X_t), g_k(X_t) \rangle_t}_{\text{Quadratic variation}=0} \\ &= -\alpha\sigma(t)^2(g_k(X_t) + \epsilon)\phi'_\delta(g_k(X_t))dt. \end{aligned}$$

For the case  $\{g_k < 0\}$ , it trivially holds that  $\phi_\delta(g_k(X_t)) = 0$  with  $\phi'_\delta(g_k(X_t)) = 0$ . Therefore, the above observations lead to the following relation for  $t \geq 0$ :

$$\frac{d}{dt}\mathbb{E}[\Psi_\delta^k(X_t)] = \frac{d}{dt}\mathbb{E}[\phi_\delta(g_k(X_t))] = -\alpha\sigma(t)^2\mathbb{E}[(g_k(X_t) + \epsilon)\phi'_\delta(g_k(X_t))].$$

At this moment, we note that the non-decreasing property of  $\phi'_\delta(r)$  implies, for  $\forall r \geq 0$ ,

$$\phi_\delta(r) = \int_0^r \phi'_\delta(s)ds \leq \int_0^r \phi'_\delta(r)ds \leq (r + \epsilon)\phi'_\delta(r) \quad \Rightarrow \quad \Psi_\delta^k(x) \leq (g_k(x) + \epsilon)\phi'_\delta(g_k(x))$$

where the inequality  $\phi_\delta(r) \leq (r + \epsilon)\phi'_\delta(r)$  also holds trivially for  $r < 0$ . Hence, we finally have

$$\frac{d}{dt}\mathbb{E}[\Psi_\delta^k(X_t)] = -\alpha\sigma(t)^2\mathbb{E}[(g_k(X_t) + \epsilon)\phi'_\delta(g_k(X_t))] \leq -\alpha\sigma(t)^2\mathbb{E}[\Psi_\delta(X_t)]$$

and the Grönwall inequality gives

$$0 \leq \mathbb{E}[\Psi_\delta^k(X_t)] \leq e^{-\alpha \int_0^t \sigma(s)^2 ds} \mathbb{E}[\Psi_\delta^k(X_0)] = 0 \quad (\because X_0 \in \Sigma)$$

which leads to  $(g_k(X_t))_+ = 0 \Rightarrow g_k(X_t) \leq 0$  for  $k \in [l]$  by letting  $\delta \downarrow 0$  and applying the monotone convergence theorem. This proves  $\mathbb{P}(g(X_t) \leq 0) = 1$  for  $t \geq 0$  and  $X_t \in \Sigma$  a.s.

Next, we prove  $\langle J_t(x), n(x) \rangle = 0$  for  $x \in \partial\Sigma$ . We first observe that [Theorem B.1](#) gives the following Fokker-Planck equation for  $g(x) > 0$ :

$$\begin{aligned} \partial_t \rho_t &= -\text{div}_\Sigma \left( \rho_t \left[ -\frac{\sigma^2}{2} \nabla_\Sigma f - \alpha\sigma^2 \nabla J^T G^\dagger J \right] \right) + \frac{\sigma^2}{2} \sum_{k=1}^d \text{div}_\Sigma(\text{div}_\Sigma(\rho_t f_k) f_k) \\ &= -\text{div}_\Sigma \left( \rho_t \left[ -\frac{\sigma^2}{2} \nabla_\Sigma f - \alpha\sigma^2 \nabla J^T G^\dagger J \right] - \frac{\sigma^2}{2} \nabla_\Sigma \rho_t \right), \end{aligned}$$

where  $f_k := \Pi e_k$  and  $e_k$  being the  $k$ -th standard basis of  $\mathbb{R}^d$ . Also, we remark that the last equality holds using the property:

$$\sum_{k=1}^d \text{div}_\Sigma(\rho_t f_k) f_k = \sum_{k=1}^d \langle \nabla_\Sigma \rho_t, f_k \rangle f_k + \underbrace{\rho_t \sum_{k=1}^d \text{div}_\Sigma(f_k) f_k}_{=0} = \nabla_\Sigma \rho_t.$$

Therefore, the probability current density  $J_t$  is given as follows

$$J_t = -\rho_t \left[ \frac{\sigma^2}{2} \nabla_\Sigma f + \alpha\sigma^2 \nabla J^T G^\dagger J \right] - \frac{\sigma^2}{2} \nabla_\Sigma \rho_t,$$

and we have

$$\begin{aligned} 0 &= \frac{d}{dt} \int_\Sigma \rho_t(x) d\sigma_\Sigma = - \int_\Sigma \text{div}_\Sigma(J_t(x)) d\sigma_\Sigma = - \int_{\partial\Sigma} \langle J_t(x), n(x) \rangle d\sigma_{\partial\Sigma} \\ &= \int_{\partial\Sigma} \alpha\rho_t\sigma^2 \underbrace{\langle \nabla J^T G^\dagger J, n \rangle}_{>0} d\sigma_{\partial\Sigma} \geq 0. \end{aligned}$$

This implies  $\rho_t = 0$  a.e on  $\partial\Sigma$  and the following boundary condition holds almost everywhere on  $\partial\Sigma$

$$\langle J_t, n \rangle = \left\langle \rho_t \left[ -\frac{\sigma^2}{2} \nabla_\Sigma f - \alpha\sigma^2 \nabla J^T G^\dagger J \right] - \frac{\sigma^2}{2} \nabla_\Sigma \rho_t, n \right\rangle = -\alpha\sigma^2 \rho_t \langle \nabla J^T G^\dagger J, n \rangle = 0.$$

□

**Lemma B.2** (Huang et al. [25]). *Let  $\{f_k\}_{k=1}^d$  be a set of vectors defined by  $f_k = \Pi(x)e_k$ , where  $\Pi(x)$  is the orthogonal projector onto  $T_x\Sigma$  and  $e_k$  is the  $k$ th standard basis vector of  $\mathbb{R}^d$ . Then, it holds that*

$$\sum_{k=1}^d (\operatorname{div}_\Sigma f_k) f_k = 0.$$

*Proof.* Let  $r$  be the rank of the  $\nabla J(x)$  and  $\{n_1(x), \dots, n_r(x)\}$  be an orthonormal basis of  $\operatorname{Im}(\nabla J(x)^T)$ . Since  $\Pi(x)$  is the orthogonal projector onto the tangent space, it can be written using the projector onto the normal space as:

$$\Pi(x) = I - \sum_{l=1}^r n_l(x) n_l(x)^T.$$

Note that by definition,  $n_l(x) \in \operatorname{Im}(\nabla J(x)^T)$  implies  $n_l(x)^T \Pi(x) = 0$  for all  $l \in [r]$ .

Next, we define a vector field  $F(x)$  by  $F(x) = \Pi(x) \operatorname{div}_\Sigma(\Pi(x))$  where  $(\operatorname{div}_\Sigma \Pi(x))_k := \operatorname{div}_\Sigma(f_k(x))$  for the vector field  $f_k(x) = \Pi(x)^T e_k = \Pi(x) e_k$ . With this definition, we have  $\operatorname{div}_\Sigma \Pi = -\sum_{l=1}^r \operatorname{div}_\Sigma(n_l n_l^T)$  and observe that for any component index  $k \in [d]$ ,

$$\begin{aligned} (\operatorname{div}_\Sigma(n_l n_l^T))_k &= \operatorname{Tr}(\Pi \nabla(n_l n_l^T e_k)) = \sum_{i,j=1}^d \Pi_{ij} \partial_i(n_l n_{lk}) = \sum_{i,j=1}^d [\Pi_{ij} (\partial_i n_{lj}) n_{lk} + \Pi_{ij} n_{lj} (\partial_i n_{lk})] \\ &= (\operatorname{div}_\Sigma n_l) n_{lk} + \sum_{i=1}^d \underbrace{(n_l^T \Pi)_i}_{=0} \partial_i n_{lk} = (\operatorname{div}_\Sigma n_l) n_{lk}, \end{aligned}$$

where we used the property that  $n_l$  is orthogonal to the tangent space ( $n_l^T \Pi = 0$ ). From this fact, we have the following result:

$$\operatorname{div}_\Sigma \Pi = -\sum_{l=1}^r \operatorname{div}_\Sigma(n_l n_l^T) = -\sum_{l=1}^r (\operatorname{div}_\Sigma n_l) n_l \Rightarrow F = \Pi \operatorname{div}_\Sigma(\Pi) = -\sum_{l=1}^r (\operatorname{div}_\Sigma n_l) \underbrace{\Pi n_l}_{=0} = 0.$$

Finally, the definition of  $F$  gives  $\sum_{k=1}^d (\operatorname{div}_\Sigma f_k) f_k = F$ , which is zero by the argument above.  $\square$

**Theorem B.2** (Stationarity of OLLA). *Assume  $\sigma(t)$  is constant for  $\forall t \geq 0$  and  $X_0 \in \Sigma$ . Then, OLLA (3) has the following stationary distribution  $\rho_\Sigma$  with respect to measure  $d\sigma_\Sigma$ :*

$$\rho_\Sigma(x) = \frac{1}{Z_\Sigma} e^{-f(x)}, \quad x \in \Sigma$$

where  $d\sigma_\Sigma$  is the surface (or Hausdorff) measure on  $\Sigma$  and  $Z_\Sigma := \int_\Sigma e^{-f(x)} d\sigma_\Sigma$  is the normalization constant.

*Proof.* To prove stationarity, we observe that

$$\int_\Sigma \mathcal{L} \phi \rho_t d\sigma_\Sigma = \frac{d}{dt} \int_\Sigma \phi \rho_t d\sigma_\Sigma = \int_\Sigma \phi \partial_t \rho_t d\sigma_\Sigma = - \int_\Sigma \phi \operatorname{div}_\Sigma(J_t) d\sigma_\Sigma = \int_\Sigma \langle J_t, \nabla_\Sigma \phi \rangle d\sigma_\Sigma$$

where the last equality comes from the boundary condition in Lemma B.1. Since  $J_t$  is given by

$$J_t = -\frac{\sigma^2}{2} (\rho_t \nabla_\Sigma f + \nabla_\Sigma \rho_t) = -\frac{\sigma^2}{2} e^{-f} \nabla_\Sigma(\rho e^f),$$

on the interior of  $\Sigma$ , we conclude that

$$\int_\Sigma \mathcal{L} \phi \rho_\Sigma d\sigma_\Sigma = \int_\Sigma \langle J_t, \nabla_\Sigma \phi \rangle d\sigma_\Sigma = -\frac{\sigma^2}{2} \int_\Sigma \langle e^{-f} 0, \nabla_\Sigma \phi \rangle d\sigma_\Sigma = 0,$$

where  $J_t = 0$  due to the fact that  $\rho_\Sigma \propto e^{-f}$ . This proves that  $\rho_\Sigma$  is the stationary distribution of the OLLA.  $\square$

## B.2 CONSTRUCTION OF ULLA

**Notations and Background for underdamped setup.** In the constrained underdamped Langevin case, assuming  $X_0 \in \Sigma$ , the natural space of  $(X_t, P_t)$  is the cotangent bundle  $T^*\Sigma := \{(x, p) \in \mathbb{R}^d \times \mathbb{R}^d \mid x \in \Sigma, \langle \nabla J(x), p \rangle = 0\}$ , where  $\Sigma := \{x \in \mathbb{R}^d \mid h(x) = 0, g(x) \leq 0\}$  is a stratified manifold. Because there is no boundary for the cotangent space  $T_x^*\Sigma := \{p \in \mathbb{R}^d \mid \langle \nabla J(x), p \rangle = 0\}$ , the boundary of  $T^*\Sigma$  is given as  $\partial T^*\Sigma = \partial\Sigma \times T_x^*\Sigma$ . In this cotangent bundle  $T^*\Sigma$ , the canonical reference measure is the Liouville (symplectic) measure  $\sigma_{T^*\Sigma}$  defined as  $d\sigma_{T^*\Sigma}(x, p) := d\sigma_\Sigma(x) \otimes dp(x)$  with  $d\sigma_\Sigma$  the surface measure on  $\Sigma$  and  $dp(x)$  the Lebesgue measure on the cotangent space  $T_x^*\Sigma$  induced by the inner product  $\langle u, v \rangle = u^T v$ .

For the notations, we write  $\nabla_\Sigma$  and  $\text{div}_\Sigma$  for the intrinsic gradient and divergence in  $x$ , and  $\nabla_{T_x^*\Sigma}$ ,  $\text{div}_{T_x^*\Sigma}$  for the intrinsic gradient and divergence in  $p$ . Under these notations, for a smooth function  $\phi$  and smooth vector field  $X$  on  $\Sigma$ , we have

$$\nabla_\Sigma \phi = \Pi \nabla_x \phi, \quad \text{div}_\Sigma(X) = \text{Tr}(\Pi \nabla_x X)$$

Similarly, for a smooth function  $\psi$  and smooth vector  $Y$  on  $T_x^*\Sigma$ , we have

$$\nabla_{T_x^*\Sigma} \psi = \Pi \nabla_p \psi, \quad \text{div}_{T_x^*\Sigma}(Y) = \text{Tr}(\Pi \nabla_p Y),$$

where  $\nabla_x$  and  $\nabla_p$  represent ambient Euclidean partial gradient or Jacobian operators with respect to  $x$  or  $p$ . Also, the global gradient with respect to  $\sigma_{T^*\Sigma}$  is given by  $\nabla_{T^*\Sigma} \phi = [\nabla_\Sigma \phi, \nabla_{T_x^*\Sigma} \phi]^T$  for any smooth function  $\phi$  on  $T^*\Sigma$  and the global divergence with respect to  $\sigma_{T^*\Sigma}$  can be represented by  $\text{div}_{T^*\Sigma}([V^x, V^p]^T) = \text{div}_\Sigma(V^x) + \text{div}_{T_x^*\Sigma}(V^p)$  for any smooth tangent field  $[V^x, V^p]^T \in T^*\Sigma$ . For comprehensive backgrounds on constrained underdamped Langevin dynamics, see Chapter 3.3 in Rousset et al. [36].

**Proposition B.2** (Construction of ULLA). *Consider the following Lagrangian-form constrained underdamped Langevin dynamics:*

$$\begin{cases} dX_t = \sigma(t)^2 P_t dt + \nabla J(X_t)^T d\lambda_t \\ dP_t = -\sigma(t)^2 \nabla f(X_t) dt - \sigma(t)^2 \gamma P_t dt + \sigma(t) \sqrt{2\gamma} \circ dW_t + \nabla J(X_t)^T d\mu_t, \end{cases} \quad (4)$$

where  $d\lambda_t, d\mu_t$  are the adapted processes such that  $dJ(X_t) = -\alpha\sigma(t)^2 J(X_t)$  (position constraint) and  $\nabla J(X_t) P_t = 0$  (momentum tangency constraint), respectively.

Assuming  $\nabla J(X_0) P_0 = 0$ , the explicit minimum norm solution of  $d\lambda_t, d\mu_t$  are given by

$$\begin{cases} d\lambda_t = -\alpha\sigma(t)^2 G^\dagger(X_t) J(X_t) dt \\ d\mu_t = G^\dagger(X_t) \nabla J(X_t) \left[ \sigma(t)^2 \nabla f(X_t) + \sigma(t)^2 \gamma P_t dt + \sigma(t) \sqrt{2\gamma} \circ dW_t \right] + \\ G^\dagger(X_t) \left[ -\sigma(t)^2 \mathcal{H}_1(X_t, P_t) + \alpha\sigma(t)^2 \mathcal{H}_2(X_t, P_t) \right] dt, \end{cases}$$

where  $G(x) := \nabla J(x) \nabla J(x)^T$  is the Gram matrix and  $\Pi(x) = I - \nabla J(x)^T G^\dagger(x) \nabla J(x)$  is the tangential projection map.

Therefore, the closed form SDE of (4) is given as follows:

$$\begin{cases} dX_t = \sigma(t)^2 P_t dt - \alpha\sigma(t)^2 \nabla J(X_t) G^\dagger(X_t) J(X_t) dt \\ dP_t = \Pi(X_t) \left[ -\sigma(t)^2 \nabla f(X_t) - \sigma(t)^2 \gamma P_t dt + \sigma(t) \sqrt{2\gamma} \circ dW_t \right] \\ - \sigma(t)^2 \nabla J(X_t)^T G^\dagger(X_t) [\mathcal{H}_1(X_t, P_t) - \alpha\mathcal{H}_2(X_t, P_t)] dt \end{cases}$$

where  $\mathcal{H}_1 \in \mathbb{R}^{m+|I_x|}$ ,  $\mathcal{H}_2 \in \mathbb{R}^{(m+|I_x|) \times (m+|I_x|)}$  are the curvature correction terms defined as

$$\begin{aligned} [\mathcal{H}_1(x, p)]_i &:= p^T \nabla^2 J_i(x) p, \\ [\mathcal{H}_2(x, p)]_i &:= p^T \nabla^2 J_i(x) (\nabla J(x)^T G^\dagger(x) J(x)) \end{aligned}$$

with  $[\mathcal{H}_1(x, p)]_i, [\mathcal{H}_2(x, p)]_i$  being the  $i$ th entry and column of  $\mathcal{H}_1(x, p), \mathcal{H}_2(x, p)$  respectively.

*Proof.* From the the Stratonovich chain rule, we observe that

$$-\alpha\sigma(t)^2 J(X_t) dt = dJ(X_t) = \nabla J(X_t) \circ dX_t = \sigma(t)^2 \underbrace{\nabla J(X_t) P_t}_{=0} dt + G(X_t) d\lambda_t.$$

Because the initial condition gives  $\nabla J(X_0)P_0 = 0$  and  $d\mu_t$  imposes  $d(\nabla J(X_t)P_t) = 0$ , the first term becomes zero and the minimum norm solution of  $d\lambda_t$  simplifies to

$$d\lambda_t = -\alpha\sigma(t)^2 G^\dagger(X_t)J(X_t)dt.$$

To find the explicit minimum norm solution for the process  $d\mu_t$ , we consider the momentum tangency constraint  $\nabla J(X_t)^T P_t = 0$ . Using the Stratonovich chain rule again, we have

$$0 = d(\nabla J(X_t)P_t) = P_t^T \nabla^2 J(X_t)dX_t + \nabla J(X_t)dP_t.$$

By substituting  $dX_t$  and  $dP_t$ , the previous equation simplifies to

$$\begin{aligned} 0 &= P_t^T \nabla^2 J(X_t) [\sigma(t)^2 P_t dt + \nabla J(X_t)^T d\lambda_t] \\ &\quad + \nabla J(X_t) \left[ -\sigma(t)^2 \nabla f(X_t) - \sigma(t)^2 \gamma P_t dt + \sigma(t) \sqrt{2\gamma} \circ dW_t + \nabla J(X_t)^T d\mu_t \right] \\ &= [\sigma^2 \mathcal{H}_1 - \alpha \sigma^2 \mathcal{H}_2 - \sigma^2 \nabla J \nabla f - \sigma^2 \gamma \nabla J P_t] dt + \sigma \nabla J \sqrt{2\gamma} \circ dW_t + G d\mu_t. \end{aligned}$$

This gives the following minimum norm solution of  $d\mu_t$ :

$$d\mu_t = G^\dagger [\sigma^2 \nabla J \nabla f + \sigma^2 \gamma \nabla J P_t - \sigma^2 \mathcal{H}_1 + \alpha \sigma^2 \mathcal{H}_2] dt - \sigma G^\dagger \nabla J \sqrt{2\gamma} \circ dW_t.$$

Therefore, we recover the following  $dX_t, dP_t$  by plugging the adapted process  $d\lambda_t, d\mu_t$  into the previous equations:

$$\begin{cases} dX_t = \sigma(t)^2 P_t dt - \alpha \sigma(t)^2 \nabla J(X_t) G^\dagger(X_t) J(X_t) dt \\ dP_t = \Pi(X_t) \left[ -\sigma(t)^2 \nabla f(X_t) dt - \sigma(t)^2 \gamma P_t dt + \sigma(t) \sqrt{2\gamma} \circ dW_t \right] \\ \quad - \sigma(t)^2 \nabla J(X_t)^T G^\dagger(X_t) [\mathcal{H}_1(X_t, P_t) - \alpha \mathcal{H}_2(X_t, P_t)] dt. \end{cases}$$

We note that this is the unique closed form SDE because  $\nabla J(X_t)^T d\lambda_t$  and  $\nabla J(X_t)^T d\mu_t$  are unique among many solutions  $d\lambda_t, d\mu_t$  satisfying the properties.

Finally, we observe that the Itô-Stratonovich correction term  $\frac{1}{2}(\nabla B)B = 0$  where  $B = [0, \sigma\sqrt{2\gamma}\Pi]^T \in \mathbb{R}^{2d \times d}$  is the diffusion matrix. This is because the position entries of  $B$  are zero and the momentum entries of  $B$  depend only on position. Therefore, we have the same formula on the Itô version of the above Stratonovich SDE.  $\square$

**Lemma B.3** (Boundary condition of ULLA). *Assuming  $X_0 \in \Sigma$  and  $\nabla J(X_0)P_0 = 0$ , ULLA (4) satisfies the following boundary condition on  $\partial\Sigma$  and property for  $t \geq 0$ :*

$$(1) \langle J_t(x, p), n(x) \rangle = 0 \quad a.e \text{ on } \partial T^* \Sigma, \quad (2) (X_t, P_t) \in T^* \Sigma,$$

where  $J_t(x)$  is the probability current density defined by  $\partial_t \rho_t = -\text{div}_\Sigma(J_t)$  and  $n(x)$  is the outward unit normal vector on  $\partial T^* \Sigma$ .

*Proof.* First, we prove  $\mathbb{P}(g_k(X_t) > 0) = 0$  for  $t \geq 0, k \in [l]$ . In particular, this implies  $(X_t, P_t) \in T^* \Sigma$  a.s. for all  $t \geq 0$ . To show this, we observe that Lemma B.4 gives

$$dg_k(X_t) = -\alpha\sigma(t)^2 (g_k(X_t) + \epsilon)dt$$

for  $\{g_k \geq 0\}$ . Thus, the same proof introduced in Lemma B.1 gives  $\mathbb{P}(g(X_t) \leq 0) = 1$  for  $t \geq 0$  and, therefore, we have  $(X_t, P_t) \in T^* \Sigma$  a.s for  $t \geq 0$ .

Next, we show the boundary condition of this SDE. To demonstrate this, we first formulate the SDE of ULLA in the following form:

$$dZ_t = V_0 dt + \sum_{k=1}^{2d} V_k \circ dB_t^k$$

with  $Z_t := [X_t, P_t]^T \in \mathbb{R}^{2d}$ ,  $V_0(x, p) := [\sigma^2 p - \alpha \sigma^2 \nabla J G^\dagger J, -\sigma^2 \Pi \nabla f - \sigma^2 \gamma \Pi p + \kappa]^T \in \mathbb{R}^{2d}$ , and  $V_k(x, p) := [0, B e_k] \in \mathbb{R}^{2d}$ , where  $\kappa := -\sigma^2 \nabla J^T G^\dagger [\mathcal{H}_1 - \alpha \mathcal{H}_2]$  is the curvature correction

related term and  $B := \sigma \Pi \sqrt{2\gamma}$ . Now, we use the Fokker-Planck equation on the interior of  $T^*\Sigma$  (Theorem B.1) applied with  $\text{div}_{T^*\Sigma}$ , and this gives the following equation:

$$\partial_t \rho_t = -\text{div}_{T^*\Sigma}(\rho_t V_0) + \frac{1}{2} \sum_{k=1}^{2d} \text{div}_{T^*\Sigma}(\text{div}_{T^*\Sigma}(\rho_t V_k) V_k),$$

where

$$-\text{div}_{T^*\Sigma}(\rho_t V_0) = -\text{div}_{\Sigma}(\rho_t (\sigma^2 p - \alpha \sigma^2 \nabla J G^\dagger J)) - \text{div}_{T_x^*\Sigma}(\rho_t (-\sigma^2 \Pi \nabla f - \sigma^2 \gamma \Pi p + \kappa))$$

and

$$\frac{1}{2} \sum_{k=1}^{2d} \text{div}_{T^*\Sigma}(\text{div}_{T^*\Sigma}(\rho_t V_k) V_k) = \frac{1}{2} \sum_{k=1}^d \text{div}_{T_x^*\Sigma}(\langle B e_k, \nabla_{T_x^*\Sigma} \rho_t \rangle B e_k) = \frac{1}{2} \text{div}_{T_x^*\Sigma}((B B^T) \nabla_{T_x^*\Sigma} \rho_t).$$

Therefore, we recover the following equation:

$$\begin{aligned} \partial_t \rho_t &= -\text{div}_{\Sigma}(\underbrace{\rho_t (\sigma^2 p - \alpha \sigma^2 \nabla J G^\dagger J)}_{:= J_t^x(x,p)}) \\ &\quad - \text{div}_{T_x^*\Sigma} \left( \underbrace{\rho_t (-\sigma^2 \Pi \nabla f - \sigma^2 \gamma \Pi p + \kappa - \sigma^2 \gamma \Pi \Pi^T \nabla_{T_x^*\Sigma} \rho_t)}_{:= J_t^p(x,p)} \right) \\ &= -\text{div}_{T^*\Sigma}([J_t^x, J_t^p]^T) = -\text{div}_{T^*\Sigma}(J_t) \end{aligned}$$

Finally, we observe that, on the boundary  $\partial T^*\Sigma = \partial \Sigma \times T_x^*\Sigma$ , the outward normal is given by  $n = [n_x, 0]^T \in \mathbb{R}^{2d}$  with  $n_x$  being the outward unit normal vector on  $\partial \Sigma$ . Therefore, we have

$$\begin{aligned} 0 &= \frac{d}{dt} \int_{T^*\Sigma} \rho_t(x, p) d\sigma_{T^*\Sigma} = - \int_{T^*\Sigma} \text{div}_{T^*\Sigma}(J_t(x, p)) d\sigma_{T^*\Sigma} = - \int_{\partial T^*\Sigma} \langle J_t(x, p), n \rangle d\sigma_{\partial T^*\Sigma} \\ &= - \int_{\partial T^*\Sigma} \langle J_t^x(x, p), n_x \rangle d\sigma_{\partial T^*\Sigma} = - \int_{\partial T^*\Sigma} \langle \rho_t (\sigma^2 p - \alpha \sigma^2 \nabla J G^\dagger J), n_x \rangle d\sigma_{\partial T^*\Sigma} \\ &= \alpha \sigma^2 \int_{\partial T^*\Sigma} \underbrace{\rho_t \langle \nabla J G^\dagger J, n_x \rangle}_{>0} d\sigma_{\partial T^*\Sigma} \end{aligned}$$

and it implies  $\rho_t(x, p) = 0$  a.e. on  $\partial T^*\Sigma$ . Lastly, we conclude the proof by observing that the following holds a.e on  $\partial T^*\Sigma$ :

$$\langle J_t, n \rangle = \langle J_t^x, n_x \rangle = \langle \rho_t (\sigma^2 p - \alpha \sigma^2 \nabla J G^\dagger J), n_x \rangle = -\alpha \sigma^2 \rho_t \langle \nabla J^T G^\dagger J, n_x \rangle = 0.$$

□

**Theorem B.3** (Stationarity of ULLA). *Assume  $\sigma(t)$  is constant for  $\forall t \geq 0$ ,  $X_0 \in \Sigma$ , and the tangency constraint  $\nabla J(X_0)P_0 = 0$  holds. Then, the ULLA (4) has the following stationary distribution  $\rho_{T^*\Sigma}$  with respect to the measure  $d\sigma_{T^*\Sigma}$ :*

$$\rho_{T^*\Sigma}(x, p) = \frac{1}{Z_{T^*\Sigma}} \exp\left(-f(x) - \frac{1}{2}\|p\|_2^2\right), \quad (x, p) \in T^*\Sigma$$

where  $d\sigma_{T^*\Sigma}$  is the Liouville measure on  $T^*\Sigma$  and  $Z_{T^*\Sigma} := \int_{T^*\Sigma} e^{(-f(x) - \frac{1}{2}\|p\|_2^2)} d\sigma_{T^*\Sigma}$  is the normalization constant.

*Proof.* First, from Lemma B.3, we know that the SDE of ULLA can be rewritten as follows on the interior of  $T^*\Sigma$ :

$$dZ_t = V_0 dt + \sum_{k=1}^{2d} V_k \circ dB_t^k$$

with  $Z_t := [X_t, P_t]^T \in \mathbb{R}^{2d}$ ,  $V_0(x, p) := [\sigma^2 p, -\sigma^2 \Pi \nabla f - \sigma^2 \gamma \Pi p + \kappa]^T \in \mathbb{R}^{2d}$ , and  $V_k(x, p) := [0, B e_k] \in \mathbb{R}^{2d}$ , where  $\kappa := -\sigma^2 \nabla J^T G^\dagger \mathcal{H}_1$  is the curvature correction related term and  $B :=$

1512  $\sigma\Pi\sqrt{2\gamma}$ . Therefore, [Theorem B.1](#) gives the following generator  $\mathcal{L}$  for any smooth function  $\phi$  on  
 1513  $T^*\Sigma$ :

$$1514 \mathcal{L}\phi = V_0\phi + \frac{1}{2} \sum_{k=1}^{2d} V_k(V_k\phi). \quad 1515$$

1516 Because we have

$$1517 V_0\phi = \langle \nabla_\Sigma \phi, \sigma^2 p \rangle + \langle \nabla_{T_x^*\Sigma} \phi, -\sigma^2 \Pi \nabla f - \sigma^2 \gamma \Pi p + \kappa \rangle$$

1518 and

$$1519 \frac{1}{2} \sum_{k=1}^{2d} V_k(V_k\phi) = \frac{1}{2} \sum_{k=1}^d \langle Be_k, \nabla_p (\langle Be_k, \nabla_p \phi \rangle) \rangle = \frac{1}{2} \sum_{k=1}^d \langle \nabla_p^2 \phi Be_k, Be_k \rangle = \sigma^2 \gamma \text{Tr}(\Pi \nabla_p^2 \phi)$$

1520 on the interior of  $T^*\Sigma$ , we can simplify  $\mathcal{L}\phi$  as follows:

$$1521 \mathcal{L}\phi = \sigma^2 \left[ \underbrace{\langle \nabla_\Sigma \phi, p \rangle - \langle \Pi \nabla f, \nabla_{T_x^*\Sigma} \phi \rangle + \langle \kappa, \nabla_{T_x^*\Sigma} \phi \rangle}_{\mathcal{L}_H \phi} + \underbrace{\gamma (\Delta_{T_x^*\Sigma} \phi - \langle \nabla_{T_x^*\Sigma} \phi, p \rangle)}_{\mathcal{L}_{OU} \phi} \right]$$

1522 where we used  $\Pi p = p$  (tangency constraint) and  $\Delta_{T_x^*\Sigma} \phi = \text{div}_{T_x^*\Sigma}(\Pi \nabla_p \phi) = \text{Tr}(\Pi \nabla_p^2 \phi)$ . Next,  
 1523 we note the following identity:

$$1524 \text{div}_{T_x^*\Sigma} \left( e^{-\|p\|^2/2} \nabla_{T_x^*\Sigma} \phi \right) = e^{-\|p\|^2/2} (\Delta_{T_x^*\Sigma} \phi - \langle \nabla_{T_x^*\Sigma} \phi, p \rangle).$$

1525 Under this identity, we observe that

$$1526 \int_{T^*\Sigma} \mathcal{L}_{OU} \phi \rho_{T^*\Sigma} d\sigma_{T^*\Sigma} = \gamma \int_{T^*\Sigma} (\Delta_{T_x^*\Sigma} \phi - \langle \nabla_{T_x^*\Sigma} \phi, p \rangle) e^{-H} d\sigma_{T^*\Sigma} = 0$$

1527 where  $H := f(x) + \frac{1}{2}\|p\|^2$  and the last equality holds because  $T_x^*\Sigma$  does not have boundary and  
 1528  $d\sigma_{T^*\Sigma}(x, p) := d\sigma_\Sigma(x) \otimes dp(x)$  holds. Lastly, we define  $X_H := [p - \alpha \nabla J G^\dagger J, -\Pi \nabla f + \kappa]^T \in$   
 1529  $\mathbb{R}^{2d}$  so that  $X_H = [p, -\Pi \nabla f + \kappa]^T$  and  $\mathcal{L}_H \phi = \langle X_H, \nabla_{T^*\Sigma} \phi \rangle$  at the interior of  $T^*\Sigma$ . Then, we  
 1530 have

$$1531 \langle X_H, \nabla_{T^*\Sigma} H \rangle = \langle p, \nabla_\Sigma f \rangle + (-\Pi \nabla f + \kappa) \Pi p = \underbrace{\langle (\nabla_\Sigma f - \Pi \nabla f), \Pi p \rangle}_{=0} + \underbrace{\langle \kappa, \Pi p \rangle}_{=0} = 0,$$

1532 where the last equality holds due to tangency constraint of  $p$ . In addition to this, the following  
 1533 identity holds on the interior of  $T^*\Sigma$  due to Liouville's theorem, which is the preservation prop-  
 1534 erty of Liouville measure on  $T^*\Sigma$  under the constrained Hamiltonian field (see Chapter 1.2.2 and  
 1535 Proposition 3.46 in Rousset et al. [36]):

$$1536 \text{div}_{T^*\Sigma} X_H = \text{div}_\Sigma(p) + \text{div}_{T_x^*\Sigma}(-\Pi \nabla f + \kappa) = 0.$$

1537 Hence, using these properties and  $\rho_{T^*\Sigma} \propto e^{-H}$ , we have

$$1538 \int_{T^*\Sigma} \mathcal{L}_{OU} \phi \rho_{T^*\Sigma} d\sigma_{T^*\Sigma} = \int_{T^*\Sigma} \langle X_H, \nabla_{T^*\Sigma} \phi \rangle \rho_{T^*\Sigma} d\sigma_{T^*\Sigma} \stackrel{(1)}{=} - \int_{T^*\Sigma} \phi \text{div}_{T^*\Sigma}(\rho_{T^*\Sigma} X_H) d\sigma_{T^*\Sigma} \\ 1539 = - \int_{T^*\Sigma} \phi \left( \langle X_H, \nabla_{T^*\Sigma} \rho_{T^*\Sigma} \rangle + \underbrace{\text{div}_{T^*\Sigma}(X_H) \rho_{T^*\Sigma}}_{=0} \right) d\sigma_{T^*\Sigma} \\ 1540 \stackrel{(2)}{=} \int_{T^*\Sigma} \phi \rho_{T^*\Sigma} \langle X_H, \nabla_{T^*\Sigma} H \rangle d\sigma_{T^*\Sigma} = 0,$$

1541 where (1) comes from the boundary condition in [Lemma B.3](#) and (2) comes from the fact that  
 1542  $-\nabla_{T^*\Sigma} H = \nabla_{T^*\Sigma} \ln \rho_{T^*\Sigma} = \nabla_{T^*\Sigma} \rho_{T^*\Sigma} / \rho_{T^*\Sigma}$ .

1543 By combining the observations above, we have

$$1544 \int_{T^*\Sigma} \mathcal{L}\phi \rho_{T^*\Sigma} d\sigma_{T^*\Sigma} = \int_{T^*\Sigma} \mathcal{L}_{OU} \phi \rho_{T^*\Sigma} d\sigma_{T^*\Sigma} + \int_{T^*\Sigma} \mathcal{L}_H \phi \rho_{T^*\Sigma} d\sigma_{T^*\Sigma} = 0,$$

1545 which proves the theorem.  $\square$

B.3 PROPERTIES AND BACKWARD PROCESSES OF CONSTRAINED LANGEVIN DYNAMICS WITH LANDING

**Exponential decaying properties of constrained Langevin dynamics with landing** Due to our previous construction, the landing term  $-\alpha \nabla J(X_t) G^T(X_t) J(X_t)$  appears on both constrained overdamped or underdamped Langevin dynamics so that the processes always satisfy  $dJ(X_t) = -\alpha \sigma(t)^2 J(X_t)$  deterministically  $\forall t \geq 0$ . Therefore, even if  $X_t \notin \Sigma$ , the process can approach  $\Sigma$  exponentially fast as illustrated in the following Lemma B.4.

**Lemma B.4** (Exponential decay of constraint functions). *Let  $J(x)$  be the constraint function vector defined as*

$$J(x) = [h_1(x), \dots, h_m(x), g_{i_1}(x) + \epsilon, \dots, g_{i_{|I_x|}}(x) + \epsilon]^T \in \mathbb{R}^{m+|I_x|} \quad (5)$$

where  $h : \mathbb{R}^d \rightarrow \mathbb{R}^m$ ,  $g : \mathbb{R}^d \rightarrow \mathbb{R}^l$  are equality and inequality constraint functions respectively, and  $I_x := \{i \in [l] : g(x) \geq 0\}$  is the active index set of inequality constraints.

Under this setup, the constrained Langevin dynamics (3 or 4) satisfy the following constraint satisfaction property almost surely:

$$h_i(X_t) = h_i(X_0) e^{-\alpha S(t)}, \quad t \geq 0$$

and

$$\begin{cases} g_j(X_t) = -\epsilon + (g_j(X_0) + \epsilon) e^{-\alpha S(t)}, & t \leq \tau_{j,\epsilon} \\ g_j(X_t) \leq 0, & t \geq \tau_{j,\epsilon}, \end{cases}$$

where  $S(t) := \int_0^t \sigma(s)^2 ds$  and  $\tau_{j,\epsilon}$  is defined to be

$$\tau_{j,\epsilon} := \inf \left\{ t \geq 0 \mid S(t) \geq \frac{1}{\alpha} \ln \left( \frac{g_j(X_0) + \epsilon}{\epsilon} \right) \right\}, \quad \forall j \in I_{x_0}.$$

*Proof.* From the Stratonovich chain rule, it holds almost surely that

$$dJ(X_t) = \nabla J(X_t) \circ dX_t = -\alpha \sigma(t)^2 J(X_t) dt.$$

For each equality constraint  $h_i$ , the component is active for  $\forall t \geq 0$ . Therefore, we have:

$$dh_i(X_t) = -\alpha \sigma(t)^2 h_i(X_t) dt$$

and solving this ODE yields:

$$h_i(X_t) = h_i(X_0) e^{-\alpha S(t)}, \quad t \geq 0.$$

For the inequality constraints, we fix  $j \in I_{X_0} := \{j \in [l] \mid g(X_0) \geq 0\}$ . While the  $j$ -th inequality is active, we have  $J_{m+j} = g_j + \epsilon$  in the constraint vector, and the same chain rule gives

$$d(g_j(X_t) + \epsilon) = -\alpha \sigma(t)^2 (g_j(X_t) + \epsilon) dt.$$

Hence, before time  $t \leq \tau_{j,\epsilon} := \inf \left\{ t \geq 0 \mid S(t) \geq \frac{1}{\alpha} \ln \left( \frac{g_j(X_0) + \epsilon}{\epsilon} \right) \right\}$ , we have

$$g_j(X_t) = -\epsilon + (g_j(X_0) + \epsilon) e^{-\alpha S(t)}.$$

After  $t \geq \tau_{j,\epsilon}$ , the particle  $X_t$  is instantaneously repelled into the interior of  $\Sigma$  whenever it hits the boundary  $\partial\Sigma$ . Therefore,  $g_j(X_t) \leq 0$  holds for  $t \geq \tau_{j,\epsilon}$ .  $\square$

**Backward process on manifold** Now, we discuss how to construct the backward process of the proposed constrained Langevin dynamics. Define  $\mathcal{M} \subset \mathbb{R}^D$  to be a smooth, compact, embedded Riemannian manifold endowed with an induced metric, and we restrict the choice of  $\mathcal{M}$  to be either  $\mathcal{M} = \Sigma$  ( $D = d$ ) or  $\mathcal{M} = T^*\Sigma$  ( $D = 2d$ ). For  $x \in \mathcal{M}$ , let  $\Pi$  be the tangential projection map on  $\mathcal{M}$ . We consider the stochastic process  $X_t \sim q_t$  driven by the following Stratonovich SDE on  $\mathcal{M}$  with  $X_0 \sim q_0 = q_{\text{data}}$ :

$$dX_t = \Pi(X_t) b(X_t, t) dt + \kappa(t) \Pi(X_t) \circ dW_t, \quad t \in [0, T] \quad (6)$$

where  $b(y, t) : \mathcal{M} \times \mathbb{R} \rightarrow \mathbb{R}^D$ ,  $\kappa(t) : \mathbb{R} \rightarrow \mathbb{R}^{D \times D}$  are the drift vector and diagonal diffusion matrix for the corresponding SDE. In practice,  $\kappa(t) : \mathbb{R} \rightarrow \mathbb{R}^{D \times D}$  is chosen to satisfy  $X_T \sim q_T \approx p_{\text{prior}}$  with  $p_{\text{prior}}$  being the easy-to-sample prior distribution on  $\mathcal{M}$  and  $T$  the terminal time for sampling.

Then, [Lemma B.5](#) shows that the stochastic process  $\overleftarrow{X}_t \sim p_t$  driven by the following Stratonovich SDE becomes the backward process of  $X_t$ :

$$d\overleftarrow{X}_t = \Pi(\overleftarrow{X}_t) \left[ -b(\overleftarrow{X}_t, T-t) + \kappa(T-t)^2 \nabla \ln q_{T-t}(\overleftarrow{X}_t) \right] dt + \kappa(T-t) \Pi(\overleftarrow{X}_t) \circ d\overleftarrow{W}_t, \quad t \in [0, T]. \quad (7)$$

It is assumed that the forward process (6) and the backward process (7) have the same boundary conditions, so that  $q_t = p_{T-t}$  holds for  $t \in [0, T]$ .

**Lemma B.5** (Backward process verification). *Let  $X_t, \overleftarrow{X}_t \in \mathcal{M}$  be the stochastic process driven by the forward process (6) and the backward process (7), respectively. If  $q_T = p_0$  and the boundary conditions of  $q_t$  and  $p_{T-t}$  appearing in the Fokker-Planck equation are the same, then the following relation holds:*

$$q_t(x) = p_{T-t}(x), \quad x \in \mathcal{M}, \quad t \in [0, T],$$

where  $q_t, p_t$  are the probability densities of each  $X_t$  and  $\overleftarrow{X}_t$ .

*Proof.* Let  $\nabla_{\mathcal{M}} := \Pi \nabla$  and  $\text{div}_{\mathcal{M}}$  be the intrinsic gradient and divergence on  $\mathcal{M}$ , and let  $\Delta_{\mathcal{M}} := \text{div}_{\mathcal{M}}(\nabla_{\mathcal{M}} \cdot)$  be the intrinsic Laplacian operator on  $\mathcal{M}$ . From [Theorem B.1](#) with  $V_0(y, t) = \Pi(y)b(y, t)$ ,  $V_k(y, t) = \kappa(t)\Pi(y)e_k$ , and  $e_k$  being  $k$ th standard basis of  $\mathbb{R}^d$ , the Fokker-Planck equation of (6) is given by

$$\partial_t q_t = -\text{div}_{\mathcal{M}}(q_t V_0) + \frac{1}{2} \sum_{k=1}^D \text{div}_{\mathcal{M}}(\text{div}_{\mathcal{M}}(q_t V_k) V_k) = -\text{div}_{\mathcal{M}}(q_t \Pi b) + \frac{1}{2} \kappa(t)^2 \Delta_{\mathcal{M}} q_t,$$

where we used the property  $\sum_{k=1}^D \text{div}_{\mathcal{M}}(\text{div}_{\mathcal{M}}(q_t \Pi e_k) (\Pi e_k)) = \Delta_{\mathcal{M}} p_t$ .

Now observe that the process  $\overleftarrow{X}_t$  driven by (7) has the following drift and diffusion term:

$$\tilde{V}_0(y, t) = \Pi(y) [-b(y, T-t) + \kappa(T-t)^2 \nabla_{\mathcal{M}} \ln q_{T-t}(x)], \quad \tilde{V}_k(y, t) = g(T-t) \Pi(y) e_k.$$

Therefore, [Theorem B.1](#) again implies the following Fokker-Planck equation:

$$\begin{aligned} \partial_t p_{T-t} &= -\partial_s p_s |_{s=T-t} = \text{div}_{\mathcal{M}}(p_{T-t} \tilde{V}_0) - \frac{1}{2} \sum_{k=1}^d \text{div}_{\mathcal{M}}(\text{div}_{\mathcal{M}}(p_{T-t} \tilde{V}_k) \tilde{V}_k) \\ &= -\text{div}_{\mathcal{M}}(p_{T-t} \Pi b) + \kappa(t)^2 \text{div}_{\mathcal{M}}(q_{T-t} \nabla_{\mathcal{M}} \ln p_{T-t}) - \frac{1}{2} \kappa(t)^2 \Delta_{\mathcal{M}} p_{T-t} \\ &= -\text{div}_{\mathcal{M}}(p_{T-t} \Pi b) + \frac{1}{2} \kappa(t)^2 \Delta_{\mathcal{M}} p_{T-t}. \end{aligned}$$

In the case of manifold with boundary, it is assumed that the boundary condition on the Fokker-Planck equation of  $q_t$  and  $p_{T-t}$  are the same and  $q_T = p_0$ . This implies that  $\overleftarrow{X}_t$  achieves the desired property as stated in the theorem.  $\square$

**Backward process of Constrained Langevin with landing** From [Proposition B.1](#), we note that the forward process  $X_t \sim q_t$  of OLLA is given as follows in the interior of  $\Sigma$ :

$$dX_t = -\frac{1}{2} \sigma(t)^2 \Pi(X_t) \nabla f(X_t) dt + \sigma(t) \Pi(X_t) \circ dW_t.$$

Therefore, applying [Lemma B.5](#) on the interior of  $\Sigma$ , the backward process of the OLLA is given as:

$$d\overleftarrow{X}_t = \frac{1}{2} \sigma(T-t)^2 \Pi(\overleftarrow{X}_t) \left[ \nabla f(\overleftarrow{X}_t) + 2 \nabla \ln q_{T-t}(\overleftarrow{X}_t) \right] dt + \sigma(t) \Pi(\overleftarrow{X}_t) \circ d\overleftarrow{W}_t.$$

Now, we note that the similar construction and proof provided in [Proposition B.1](#) and [Lemma B.1](#) can demonstrate that adding the landing term  $-\alpha\sigma^2\nabla JG^\dagger J$  to the previous SDE enforces it to have the same boundary condition  $\langle J_t, n \rangle = 0$  a.e. on  $\partial\Sigma$  imposed on the forward process. Therefore, the backward process  $\overleftarrow{X}_t$  is given as:

$$d\overleftarrow{X}_t = \frac{1}{2}\sigma(T-t)^2\Pi(\overleftarrow{X}_t) \left[ \nabla f(\overleftarrow{X}_t) + 2\nabla \ln q_{T-t}(\overleftarrow{X}_t) \right] dt - \alpha\sigma(T-t)^2\nabla J(\overleftarrow{X}_t)^T G^\dagger(\overleftarrow{X}_t) J(\overleftarrow{X}_t) dt + \frac{\sigma(t)^2}{2}\mathcal{H}(\overleftarrow{X}_t)dt + \sigma(t)\Pi(\overleftarrow{X}_t) \circ d\overleftarrow{W}_t, \quad (\text{Stratonovich-sense})$$

which is equal to the following Itô version SDE involving the Itô-Stratonovich correction term  $\mathcal{H}$ :

$$d\overleftarrow{X}_t = \frac{1}{2}\sigma(T-t)^2\Pi(\overleftarrow{X}_t) \left[ \nabla f(\overleftarrow{X}_t) + 2\nabla \ln q_{T-t}(\overleftarrow{X}_t) \right] dt + \frac{1}{2}\sigma(T-t)^2\mathcal{H}(\overleftarrow{X}_t)dt - \underbrace{\alpha\sigma(T-t)^2\nabla J(\overleftarrow{X}_t)^T G^\dagger(\overleftarrow{X}_t) J(\overleftarrow{X}_t)}_{\text{Landing term}} dt + \sigma(T-t)\Pi(\overleftarrow{X}_t) \circ d\overleftarrow{W}_t. \quad (\text{Itô-sense})$$

Similarly, from [Proposition B.2](#), the forward process  $[X_t, P_t]^T \sim \overleftarrow{P}_t$  of ULLA is given as below in the interior of  $\Sigma$ :

$$\begin{cases} dX_t = \sigma(t)^2 P_t dt \\ dP_t = \Pi(X_t) \left[ \sigma(t)^2 [-\nabla f(X_t) - \gamma P_t] dt + \sigma(t)\sqrt{2\gamma} \circ dW_t \right] - \sigma(t)^2 \nabla J(X_t)^T G^\dagger(X_t) \mathcal{H}_1(X_t, P_t) dt. \end{cases}$$

Therefore, [Lemma B.5](#) again implies that the following stochastic process  $[\overleftarrow{X}_t, \overleftarrow{P}_t]^T \sim p_t$  becomes the backward process of  $X_t$  on the interior of  $\Sigma$ :

$$\begin{cases} d\overleftarrow{X}_t = -\sigma(T-t)^2 \overleftarrow{P}_t dt \\ d\overleftarrow{P}_t = \Pi(\overleftarrow{X}_t) \left[ \sigma(T-t)^2 \left[ \nabla f(\overleftarrow{X}_t) dt + \gamma \overleftarrow{P}_t dt + 2\gamma \nabla_p \ln q_{T-t}(\overleftarrow{X}_t, \overleftarrow{P}_t) \right] dt + \sigma(T-t)\sqrt{2\gamma} \circ d\overleftarrow{W}_t \right] + \sigma(T-t)^2 \nabla J(\overleftarrow{X}_t)^T G^\dagger(\overleftarrow{X}_t) \mathcal{H}_1(\overleftarrow{X}_t, \overleftarrow{P}_t) dt. \end{cases}$$

Also, the similar construction and proof in [Proposition B.2](#) and [Lemma B.3](#) show that adding both the landing term  $-\alpha\sigma^2\nabla JG^\dagger J$  and the corresponding landing correction term  $+\alpha\sigma^2\nabla JG^\dagger \mathcal{H}_2$  to the previous SDE imposes the same boundary condition as in the forward process. Hence, the backward process  $[\overleftarrow{X}_t, \overleftarrow{P}_t]^T$  is provided as:

$$\begin{cases} d\overleftarrow{X}_t = -\sigma(T-t)^2 \overleftarrow{P}_t dt - \underbrace{\alpha\sigma(T-t)^2 \nabla J(\overleftarrow{X}_t)^T G^\dagger(\overleftarrow{X}_t) J(\overleftarrow{X}_t)}_{\text{Landing term}} dt \\ d\overleftarrow{P}_t = \Pi(\overleftarrow{X}_t) \left[ \sigma(T-t)^2 \left[ \nabla f(\overleftarrow{X}_t) dt + \gamma \overleftarrow{P}_t dt + 2\gamma \nabla_p \ln q_{T-t}(\overleftarrow{X}_t, \overleftarrow{P}_t) \right] dt + \sigma(T-t)\sqrt{2\gamma} \circ d\overleftarrow{W}_t \right] + \sigma(T-t)^2 \nabla J(\overleftarrow{X}_t)^T G^\dagger(\overleftarrow{X}_t) \left[ \mathcal{H}_1(\overleftarrow{X}_t, \overleftarrow{P}_t) + \underbrace{\alpha\mathcal{H}_2(\overleftarrow{X}_t, \overleftarrow{P}_t)}_{\text{Landing correction term}} \right] dt. \end{cases}$$

Because the Itô-Stratonovich correction term vanishes in the underdamped case [Proposition B.2](#), the Itô version SDE can be recovered from the Stratonovich version SDE by chaining  $\circ$  to  $\cdot$  in the Brownian motion term  $d\overleftarrow{W}_t$ .

#### B.4 DISCRETIZATION OF CONSTRAINED LANGEVIN DYNAMICS

In the discretization setup, we use a uniform grid  $t_k = k\Delta t$  for  $k = 0, \dots, N$  with terminal time  $T = N\Delta T$ . The forward trajectory uses state notations  $X_k := X_{t_k}$  (or  $[X_k, P_k]^T := [X_{t_k}, X_{t_k}]^T$ ) and updates in ascending index as  $X_k \leftarrow X_{k+1} + \dots$ . The backward trajectory is written in

1728 descending index, using states  $X'_k := \overleftarrow{X}_{T-t_k}$  (or  $[X'_k, P'_k]^T := [\overleftarrow{X}_{T-t_k}, \overleftarrow{P}_{T-t_k}]^T$ ) and updating  
 1729 as  $X'_k \leftarrow X'_{k+1} + \dots$  for  $k \in \{N-1, \dots, 0\}$ . We define the noise schedule in continuous time and  
 1730 evaluate the schedule on the discrete grid. This yields  
 1731

$$1732 \quad \sigma(t) := \sigma_{\min} + \frac{t}{T} (\sigma_{\max} - \sigma_{\min}), \quad \sigma_k := \sigma(t_k) = \sigma_{\min} + \frac{k}{N} (\sigma_{\max} - \sigma_{\min}).$$

1734 We also denote  $q_k, p_k$  to be the probability densities of  $X_k$  (or  $[X_k, P_k]^T$ ) and  $X'_k$  (or  $[X'_k, P'_k]^T$ ) so  
 1735 that  $q_k = q_{t_k}$  and  $p_k = p_{T-t_k}$ .

1736 **Discretization of OLLA** For the discretization of OLLA, we use straightforward Euler-  
 1737 Maruyama (EM) discretization.

1738 **Discretization of Forward OLLA** Recall that the Itô version of the forward process SDE is  
 1739 provided as follows:

$$1741 \quad dX_t = \frac{\sigma(t)^2}{2} [-\Pi(X_t)\nabla f(X_t) - \alpha\sigma(t)^2\nabla J(X_t)^T G^\dagger(X_t)J(X_t)] dt + \frac{\sigma(t)^2}{2} \mathcal{H}(X_t)dt$$

$$1742 \quad + \sigma(t)\Pi(X_t)dW_t.$$

1744 Therefore, the EM discretization of the forward process SDE becomes:

$$1746 \quad X_{k+1} = X_k + \frac{\sigma_k^2}{2} [-\Pi(X_k)\nabla f(X_k) - \alpha\sigma_k^2\nabla J(X_k)^T G^\dagger(X_k)J(X_k)] \Delta t$$

$$1747 \quad + \frac{\sigma_k^2}{2} \mathcal{H}(X_k)\Delta t + \sigma_k\sqrt{\Delta t}\Pi(X_k)\zeta_k, \quad (\text{Forward-OLLA})$$

1750 where  $\zeta_k \sim \mathcal{N}(0, I)$  is the standard Gaussian noise.

1751 **Discretization of Backward OLLA** Similarly, we note that following the backward SDE

$$1753 \quad d\overleftarrow{X}_t = \frac{1}{2}\sigma(T-t)^2\Pi(\overleftarrow{X}_t) \left[ \nabla f(\overleftarrow{X}_t) + 2\nabla \ln q_{T-t}(\overleftarrow{X}_t) \right] \Delta t + \frac{1}{2}\sigma(T-t)^2\mathcal{H}(\overleftarrow{X}_t)dt$$

$$1754 \quad - \alpha\sigma(T-t)^2\nabla J(\overleftarrow{X}_t)^T G^\dagger(\overleftarrow{X}_t)J(\overleftarrow{X}_t)dt + \sigma(T-t)\Pi(\overleftarrow{X}_t) \circ d\overleftarrow{W}_t$$

1757 can be discretized as follows:

$$1758 \quad \overleftarrow{X}_{t_{k+1}} = \overleftarrow{X}_{t_k} + \frac{1}{2}\sigma(T-t_k)^2\Pi(\overleftarrow{X}_{t_k}) \left[ \nabla f(\overleftarrow{X}_{t_k}) + 2\nabla \ln q_{T-t_k}(\overleftarrow{X}_{t_k}) \right] \Delta t + \frac{1}{2}\sigma(T-t_k)^2\mathcal{H}(\overleftarrow{X}_{t_k})\Delta t$$

$$1759 \quad - \alpha\sigma(T-t_k)^2\nabla J(\overleftarrow{X}_{t_k})^T G^\dagger(\overleftarrow{X}_{t_k})J(\overleftarrow{X}_{t_k})\Delta t + \sigma(t_k)\sqrt{\Delta t}\Pi(\overleftarrow{X}_{t_k})\bar{\zeta}_k.$$

1762 In our notation, this is equivalent to

$$1764 \quad X'_k = X'_{k+1} + \frac{1}{2}\sigma_{k+1}^2\Pi(X'_{k+1}) \left[ \nabla f(X'_{k+1}) + 2\nabla \ln q_{k+1}(X'_{k+1}) \right] \Delta t + \frac{1}{2}\sigma_{k+1}^2\mathcal{H}(X'_{k+1})\Delta t$$

$$1765 \quad - \alpha\sigma_{k+1}^2\nabla J(X'_{k+1})^T G^\dagger(X'_{k+1})J(X'_{k+1})\Delta t + \sigma_{k+1}\sqrt{\Delta t}\Pi(X'_{k+1})\zeta'_{k+1}$$

1766 (Backward-OLLA)

1768 by changing  $k \leftarrow N - (k + 1)$ .

1770 **Discretized algorithm of constrained overdamped via Lagrangian multiplier** When we are  
 1771 available to use Newton's method, we replace the explicit normal drifts  $\mathcal{H}, -\alpha\nabla J^T G^\dagger J$  terms by a  
 1772 position projection via Lagrangian multipliers  $\lambda$  at each step so that  $J(x) = 0$  is satisfied. Under this  
 1773 way, we recover the following discretization of constrained overdamped Langevin dynamics using  
 1774 Lagrangian multiplier:

$$1775 \quad X_{k+1} = X_k - \frac{\sigma_k^2}{2}\Pi(X_k)\nabla f(X_k)\Delta t + \sigma_k\sqrt{\Delta t}\Pi(X_k)\zeta_k + \nabla J(X_k)^T \lambda_k \quad (\text{Forward-OLLA-P})$$

1777 with  $\lambda_k$  such that  $J(X_{k+1}) = 0$ . Similarly, the backward process can be obtained in a similar way  
 1778 as follows:

$$1779 \quad X'_k = X'_{k+1} + \frac{1}{2}\sigma_{k+1}^2\Pi(X'_{k+1}) \left[ \nabla f(X'_{k+1}) + 2\nabla \ln q_{k+1}(X'_{k+1}) \right] \Delta t + \sigma_{k+1}\sqrt{\Delta t}\Pi(X'_{k+1})\zeta'_{k+1}$$

$$1780 \quad + \nabla J(X'_{k+1})^T \lambda_{k+1} \quad (\text{Backward-OLLA-P})$$

1781

with  $\lambda_{k+1}$  such that  $J(X'_k) = 0$ .

**Discretization of ULLA** For the discretization of ULLA, we use a 1st order  $O\tilde{B}A$  splitting scheme which uses an approximated  $B$  process to compute the curvature correction term at  $P_k$  with  $\mathcal{O}(\Delta t)$  error. When discretizing, we use a collapsing technique to remove the momentum update rule. Because this collapsing technique requires saving only the position  $X_k$ 's. it is beneficial for saving memory, especially when saving forward trajectories is necessary.

**Discretization of Forward ULLA** From the previous subsection, we recall the following Itô version of the forward process SDE:

$$\begin{cases} dX_t = \sigma(t)^2 P_t dt - \alpha \sigma(t) \nabla J(X_t) G^\dagger(X_t) J(X_t) dt \\ dP_t = \Pi(X_t) \left[ \sigma(t)^2 [-\nabla f(X_t) - \gamma P_t] dt + \sigma(t) \sqrt{2\gamma} dW_t \right] \\ \quad - \sigma(t)^2 \nabla J(X_t)^T G^\dagger(X_t) [\mathcal{H}_1(X_t, P_t) - \alpha \mathcal{H}_2(X_t, P_t)] dt. \end{cases}$$

Under  $O\tilde{B}A$  scheme, this can be split into three parts  $O, \tilde{B}, A$  as follows:

$$\begin{aligned} O: & \begin{cases} dX_t = 0 \\ dP_t = -\sigma(t)^2 \Pi(X_t) \gamma P_t dt + \sigma(t) \Pi(X_t) \sqrt{2\gamma} dW_t \end{cases} + \\ \tilde{B}: & \begin{cases} dX_t = 0 \\ dP_t = -\sigma(t)^2 \left( \Pi(X_t) \nabla f(X_t) dt + \nabla J(X_t)^T G^\dagger(X_t) [\mathcal{H}_1(X_t, P_t) - \alpha \mathcal{H}_2(X_t, P_t)] dt \right) \end{cases} + \\ A: & \begin{cases} dX_t = \sigma(t)^2 P_t dt - \alpha \sigma(t)^2 \nabla J(X_t) G^\dagger(X_t) J(X_t) dt \\ dP_t = 0. \end{cases} \end{aligned}$$

First, we integrate  $O$  step from  $t = t_k$  to  $t = t_{k+1}$  with  $X_t$  and  $\sigma(t)$  frozen at  $t = t_k$ . Then, it gives

$$\begin{cases} X_{k+1}^O = X_k \\ P_{k+1}^O = (1 - \Pi(X_k)) P_k + a_k \Pi(X_k) P_k + \sqrt{1 - a_k^2} \Pi(X_k) \zeta_k = a_k \Pi(X_k) P_k + \sqrt{1 - a_k^2} \Pi(X_k) \zeta_k \end{cases}$$

where  $a_k := e^{-\gamma \sigma_k^2 \Delta t}$  and we used the assumption that  $P_k \in T_{X_k} \Sigma$ , while will be guaranteed on the collapsing technique. Next, we integrate  $\tilde{B}$  step with same integration domain with  $X_t, \sigma(t)$  frozen at  $t = t_k$ . Then, we have:

$$\begin{cases} X_{k+1}^{O\tilde{B}} = X_{k+1}^O + 0 = X_k \\ P_{k+1}^{O\tilde{B}} = P_{k+1}^O - \sigma_k^2 \Pi(X_k) \nabla f(X_k) \Delta t - \sigma_k^2 \nabla J(X_k)^T G^\dagger(X_k) [\mathcal{H}_1(X_k, P_k) - \alpha \mathcal{H}_2(X_k, P_k)] \Delta t. \end{cases}$$

Similarly, integrating  $A$  step with frozen  $X_t, \sigma(t)$  gives :

$$\begin{cases} X_{k+1} = X_{k+1}^{O\tilde{B}} + \sigma_k^2 P_{k+1}^{O\tilde{B}} \Delta t - \alpha \sigma_k^2 \nabla J(X_k)^T G(X_k)^\dagger J(X_k) \Delta t \\ P_{k+1} = P_{k+1}^{O\tilde{B}} + 0 = P_{k+1}^{O\tilde{B}}, \end{cases}$$

which is equivalent to

$$\begin{cases} P_{k+1} = \Pi(X_k) \left[ a_k P_k - \sigma_k^2 \nabla f(X_k) \Delta t + \sqrt{1 - a_k^2} \zeta_k \right] \\ \quad - \sigma_k^2 \nabla J(X_k)^T G^\dagger(X_k) [\mathcal{H}_1(X_k, P_k) - \alpha \mathcal{H}_2(X_k, P_k)] \Delta t \\ X_{k+1} = X_k + \sigma_k^2 P_{k+1} \Delta t - \alpha \sigma_k^2 \nabla J(X_k)^T G(X_k)^\dagger J(X_k) \Delta t. \end{cases}$$

We note that this can be collapsed into one update rule with respect to  $X$  state as follows:

$$\begin{aligned} X_{k+1} &= X_k + \sigma_k^2 \Delta t \Pi(X_k) \left[ a_k P_k - \sigma_k^2 \nabla f(X_k) \Delta t + \sqrt{1 - a_k^2} \zeta_k \right] \\ &\quad - \alpha \sigma_k^2 \nabla J(X_k)^T G(X_k)^\dagger J(X_k) \Delta t - \sigma_k^4 \Delta t^2 \nabla J(X_k)^T G(X_k)^\dagger [\mathcal{H}_1(X_k, P_k) - \alpha \mathcal{H}_2(X_k, P_k)]. \end{aligned}$$

Also, we observe that  $P_k$  can be recovered as  $P_k = \Pi(X_{k-1}) \left( \frac{X_k - X_{k-1}}{\sigma_{k-1}^2 \Delta t} \right)$  from the previous recursion which again can be approximated by  $P_k \approx \tilde{P}_k := \Pi(X_k) \left( \frac{X_k - X_{k-1}}{\sigma_{k-1}^2 \Delta t} \right)$  with error  $\mathcal{O}(\Delta t)$ ,

1836 guaranteeing the 1st order numerical error and  $P_k \in T_{X_k} \Sigma$  assumption on  $O$  step. Therefore, the  
 1837 final update rule for the forward process of ULLA becomes:  
 1838

$$1839 \quad X_{k+1} = X_k + \sigma_k^2 \Delta t \Pi(X_k) \left[ a_k \tilde{P}_k - \sigma_k^2 \nabla f(X_k) \Delta t + \sqrt{1 - a_k^2} \zeta_k \right] \quad (\text{Forward-ULLA})$$

$$1840 \quad - \alpha \sigma_k^2 \nabla J(X_k)^T G(X_k)^\dagger J(X_k) \Delta t - \sigma_k^4 \Delta t^2 \nabla J(X_k)^T G^\dagger(X_k) \left[ \mathcal{H}_1(X_k, \tilde{P}_k) - \alpha \mathcal{H}_2(X_k, \tilde{P}_k) \right]$$

1841 with  $\tilde{P}_k := \Pi(X_k) \left( \frac{X_k - X_{k-1}}{\sigma_{k-1}^2 \Delta t} \right)$  for  $k \in \{1, \dots, N-1\}$  and  $\tilde{P}_0 = \Pi(X_0) \zeta \in T_{X_0}^\Sigma$  where  $\zeta \sim$   
 1842  $\mathcal{N}(0, I)$ .  
 1843

1844 **Discretization of Backward ULLA** For the discretization of backward process SDE, we recall  
 1845 the following backward process SDE of ULLA:  
 1846

$$1847 \quad \begin{cases} d\overleftarrow{X}_t = -\sigma(T-t)^2 \overleftarrow{P}_t dt - \alpha \sigma(T-t)^2 \nabla J(\overleftarrow{X}_t) G^\dagger(\overleftarrow{X}_t) J(\overleftarrow{X}_t) dt \\ d\overleftarrow{P}_t = \Pi(\overleftarrow{X}_t) \left[ \sigma(T-t)^2 \left[ \nabla f(\overleftarrow{X}_t) dt + \gamma \overleftarrow{P}_t dt + 2\gamma \nabla_p \ln q_{T-t}(\overleftarrow{X}_t, \overleftarrow{P}_t) \right] dt + \sigma(T-t) \sqrt{2\gamma} d\overleftarrow{W}_t \right] \\ \quad + \sigma(T-t)^2 \nabla J(\overleftarrow{X}_t)^T G^\dagger(\overleftarrow{X}_t) \left[ \mathcal{H}_1(\overleftarrow{X}_t, \overleftarrow{P}_t) + \alpha \mathcal{H}_2(\overleftarrow{X}_t, \overleftarrow{P}_t) \right] dt. \end{cases}$$

1848 For the backward process of the  $O\tilde{B}A$  scheme, the  $2\sigma(T-t)^2 \Pi(\overleftarrow{X}_t) \gamma \overleftarrow{P}_t$  is added to  $\tilde{B}$  step to  
 1849 guarantee the stable non-exploding OU process at  $O$  step. We remark that the similar trick to handle  
 1850  $O$  step was previously used in Dockhorn et al. [51]. Under this technique, each  $O, \tilde{B}, A$  steps can  
 1851 be given as follows:  
 1852

$$1853 \quad O: \begin{cases} d\overleftarrow{X}_t = 0 \\ d\overleftarrow{P}_t = -\sigma(T-t)^2 \Pi(\overleftarrow{X}_t) \gamma \overleftarrow{P}_t dt + \sigma(T-t) \Pi(\overleftarrow{X}_t) \sqrt{2\gamma} d\overleftarrow{W}_t \end{cases} +$$

$$1854 \quad \tilde{B}: \begin{cases} d\overleftarrow{X}_t = 0 \\ d\overleftarrow{P}_t = \sigma(T-t)^2 \Pi(\overleftarrow{X}_t) \left( \nabla f(\overleftarrow{X}_t) + 2\gamma \nabla_p \ln q_{T-t}(\overleftarrow{X}_t, \overleftarrow{P}_t) + 2\gamma \overleftarrow{P}_t \right) dt + \\ \quad + \nabla J(\overleftarrow{X}_t)^T G^\dagger(\overleftarrow{X}_t) \left[ \mathcal{H}_1(\overleftarrow{X}_t, \overleftarrow{P}_t) + \alpha \mathcal{H}_2(\overleftarrow{X}_t, \overleftarrow{P}_t) \right] dt \end{cases} +$$

$$1855 \quad A: \begin{cases} d\overleftarrow{X}_t = \sigma(T-t)^2 \left( -\overleftarrow{P}_t dt - \alpha \nabla J(\overleftarrow{X}_t) G^\dagger(\overleftarrow{X}_t) J(\overleftarrow{X}_t) dt \right) \\ d\overleftarrow{P}_t = 0. \end{cases}$$

1856 Integrating  $O$  step from  $t = t_k$  to  $t = t_{k+1}$  with  $\overleftarrow{X}_t$  and  $\sigma(T-t)$  frozen at  $t = t_k$ . Then, it gives  
 1857

$$1858 \quad \begin{cases} \overleftarrow{X}_{t_{k+1}}^O = \overleftarrow{X}_{t_k} \\ \overleftarrow{P}_{t_{k+1}}^O = a_{N-k} \Pi(\overleftarrow{X}_{t_k}) \overleftarrow{P}_{t_k} + \sqrt{1 - a_{N-k}^2} \Pi(\overleftarrow{X}_{t_k}) \bar{\zeta}_k, \end{cases}$$

1859 where  $a_{N-k} = e^{-\gamma \sigma_{N-k}^2 \Delta t}$  and we again used the assumption that  $\overleftarrow{P}_{t_k} \in T_{\overleftarrow{X}_{t_k}}^\Sigma$ , which will be  
 1860 guaranteed via the collapsing technique later on. Next, integrating  $\tilde{B}$  step with the same integration  
 1861 domain with  $\overleftarrow{X}_t, \sigma(t)$  frozen at  $t = t_k$  gives:  
 1862

$$1863 \quad \begin{cases} \overleftarrow{X}_{t_{k+1}}^{O\tilde{B}} = \overleftarrow{X}_{t_{k+1}}^O + 0 = \overleftarrow{X}_{t_k} \\ \overleftarrow{P}_{t_{k+1}}^{O\tilde{B}} = \overleftarrow{P}_{t_{k+1}}^O + \sigma_{N-k}^2 \Pi(\overleftarrow{X}_{t_k}) \left( \nabla f(\overleftarrow{X}_{t_k}) + 2\gamma \nabla_p \ln q_{T-t_k}(\overleftarrow{X}_{t_k}, \overleftarrow{P}_{t_k}) + 2\gamma \overleftarrow{P}_{t_k} \right) \Delta t \\ \quad + \sigma_{N-k}^2 \nabla J(\overleftarrow{X}_{t_k})^T G^\dagger(\overleftarrow{X}_{t_k}) \left[ \mathcal{H}_1(\overleftarrow{X}_{t_k}, \overleftarrow{P}_{t_k}) + \alpha \mathcal{H}_2(\overleftarrow{X}_{t_k}, \overleftarrow{P}_{t_k}) \right] \Delta t \end{cases}$$

Similarly, integrating  $A$  step with frozen  $\overleftarrow{X}_t, \sigma(t)$  gives :

$$\begin{cases} \overleftarrow{X}_{t_{k+1}} = \overleftarrow{X}_{t_{k+1}}^{O\bar{B}} - \sigma_{N-k}^2 P_k^{O\bar{B}} \Delta t - \alpha \sigma_{N-k}^2 \nabla J(\overleftarrow{X}_{t_k})^T G(\overleftarrow{X}_{t_k})^\dagger J(\overleftarrow{X}_{t_k}) \Delta t \\ \overleftarrow{P}_{t_{k+1}} = \overleftarrow{P}_{t_{k+1}}^{O\bar{B}} + 0 = \overleftarrow{P}_{t_{k+1}}^{O\bar{B}} \end{cases}$$

which is equivalent to

$$\begin{cases} \overleftarrow{P}_{t_{k+1}} = \Pi(\overleftarrow{X}_{t_k}) \left[ a_{N-k} \overleftarrow{P}_{t_k} + \sigma_{N-k}^2 \left( \nabla f(\overleftarrow{X}_{t_k}) + 2\gamma \nabla_p \ln q_{T-t_k}(\overleftarrow{X}_{t_k}, \overleftarrow{P}_{t_k}) + 2\gamma \overleftarrow{P}_{t_k} \right) \Delta t \right] \\ + \sqrt{1 - a_{N-k}^2} \Pi(\overleftarrow{X}_{t_k}) \bar{\zeta}_k + \sigma_{N-k}^2 \nabla J(\overleftarrow{X}_{t_k})^T G^\dagger(\overleftarrow{X}_{t_k}) \left[ \mathcal{H}_1(\overleftarrow{X}_{t_k}, \overleftarrow{P}_{t_k}) + \alpha \mathcal{H}_2(\overleftarrow{X}_{t_k}, \overleftarrow{P}_{t_k}) \right] \Delta t \\ \overleftarrow{X}_{t_{k+1}} = \overleftarrow{X}_{t_k} - \sigma_{N-k}^2 \overleftarrow{P}_{t_{k+1}} \Delta t - \alpha \sigma_{N-k}^2 \nabla J(\overleftarrow{X}_{t_k})^T G(\overleftarrow{X}_{t_k})^\dagger J(\overleftarrow{X}_{t_k}) \Delta t \end{cases}$$

Similarly as before, we note that  $\overleftarrow{P}_{t_{k+1}}$  can be recovered as  $\overleftarrow{P}_{t_{k+1}} = \Pi(\overleftarrow{X}_{t_{k-1}}) \left( \frac{\overleftarrow{X}_{t_{k-1}} - \overleftarrow{X}_{t_k}}{\sigma_{N-k+1}^2 \Delta t} \right)$

from the previous recursion, and it can be approximated by  $\tilde{\overleftarrow{P}}_{t_{k+1}} \approx \Pi(\overleftarrow{X}_{t_k}) \left( \frac{\overleftarrow{X}_{t_{k-1}} - \overleftarrow{X}_{t_k}}{\sigma_{N-k+1}^2 \Delta t} \right)$  with  $\mathcal{O}(\Delta t)$  error. Therefore, we can collapse these two position-momentum updates into one single position update with approximated  $\tilde{\overleftarrow{P}}_{t_{k+1}}$  as follows:

$$\begin{aligned} \overleftarrow{X}_{t_{k+1}} &= \overleftarrow{X}_{t_k} + \sigma_{N-k}^2 \Delta t \sqrt{1 - a_{N-k}^2} \Pi(\overleftarrow{X}_{t_k}) \bar{\zeta}_k - \alpha \sigma_{N-k}^2 \nabla J(\overleftarrow{X}_{t_k})^T G(\overleftarrow{X}_{t_k})^\dagger J(\overleftarrow{X}_{t_k}) \Delta t \\ &- \sigma_{N-k}^2 \Delta t \Pi(\overleftarrow{X}_{t_k}) \left[ a_{N-k} \tilde{\overleftarrow{P}}_{t_k} + \sigma_{N-k}^2 \left( \nabla f(\overleftarrow{X}_{t_k}) + 2\gamma \left( \nabla_p \ln q_{T-t_k}(\overleftarrow{X}_{t_k}, \tilde{\overleftarrow{P}}_{t_k}) + \tilde{\overleftarrow{P}}_{t_k} \right) \right) \Delta t \right] \\ &- \sigma_{N-k}^4 \Delta t^2 \nabla J(\overleftarrow{X}_{t_k})^T G^\dagger(\overleftarrow{X}_{t_k}) \left[ \mathcal{H}_1(\overleftarrow{X}_{t_k}, \tilde{\overleftarrow{P}}_{t_k}) + \alpha \mathcal{H}_2(\overleftarrow{X}_{t_k}, \tilde{\overleftarrow{P}}_{t_k}) \right]. \end{aligned}$$

Finally, we recover the following discretization of the backward SDE of ULLA by changing index  $k \leftarrow N - (k + 1)$ :

$$\begin{aligned} X'_k &= X'_{k+1} + \sigma_{k+1}^2 \Delta t \sqrt{1 - a_{k+1}^2} \Delta t \Pi(X'_{k+1}) \zeta'_{k+1} - \alpha \sigma_{k+1}^2 \nabla J(X'_{k+1})^T G(X'_{k+1})^\dagger J(X'_{k+1}) \Delta t - \\ &\sigma_{k+1}^2 \Delta t \Pi(X'_{k+1}) \left[ a_{k+1} \tilde{P}'_{k+1} + \sigma_{k+1}^2 \left( \nabla f(X'_{k+1}) + 2\gamma \left( \nabla_p \ln q_{k+1}(X'_{k+1}, \tilde{P}'_{k+1}) + \tilde{P}'_{k+1} \right) \right) \Delta t \right] \\ &- \sigma_{k+1}^4 \Delta t^2 \nabla J(X'_{k+1})^T G^\dagger(X'_{k+1}) \left[ \mathcal{H}_1(X'_{k+1}, \tilde{P}'_{k+1}) + \alpha \mathcal{H}_2(X'_{k+1}, \tilde{P}'_{k+1}) \right] \end{aligned} \quad (\text{Backward-ULLA})$$

with  $\tilde{P}'_{k+1} := \Pi(X'_{k+1}) \left( \frac{X'_{k+2} - X'_{k+1}}{\sigma_{k+2}^2 \Delta t} \right)$  for  $k \in \{0, \dots, N-2\}$  and  $\tilde{P}'_N = \Pi(X'_N) \zeta \in T_{X'_N}^\Sigma$  where  $\zeta \sim \mathcal{N}(0, I)$ .

**Discretized algorithm of constrained underdamped via Lagrangian multiplier** Similar to constrained overdamped with Lagrangian multiplier, we replace the explicit normal drifts  $-\alpha \nabla J^T G^\dagger J, \nabla J^T G^\dagger \mathcal{H}_1, \nabla J^T G^\dagger \mathcal{H}_2$  terms by a position projection via Lagrangian multipliers  $\lambda$  at each step so that  $J(x) = 0$  is satisfied. In the collapsed underdamped case, we also used the approximation to re-tangent the momentum  $P$  by  $\tilde{P}$ . So, no separate momentum projection is required. Under this idea, we recover the following discretization of constrained underdamped Langevin dynamics via Lagrangian multiplier:

$$X_{k+1} = X_k + \sigma_k^2 \Delta t \Pi(X_k) \left[ a_k \tilde{P}_k - \sigma_k^2 \nabla f(X_k) \Delta t + \sqrt{1 - a_k^2} \zeta_k \right] + \nabla J(X_k)^T \lambda_k \quad (\text{Forward-ULLA-P})$$

with  $\lambda_k$  such that  $J(X_{k+1}) = 0$ . Similarly, we obtain the following backward discretization:

$$X'_k = X'_{k+1} + \sigma_{k+1}^2 \Delta t \sqrt{1 - a_{k+1}^2} \Pi(X'_{k+1}) \zeta'_{k+1} + \nabla J(X'_{k+1}) \lambda_{k+1} - \sigma_{k+1}^2 \Delta t \Pi(X'_{k+1}) \left[ a_{k+1} \tilde{P}'_{k+1} + \sigma_{k+1}^2 \left( \nabla f(X'_{k+1}) + 2\gamma \left( \nabla_p \ln q_{k+1}(X'_{k+1}, \tilde{P}'_{k+1}) + \tilde{P}'_{k+1} \right) \right) \right] \Delta t$$

(Backward-ULLA-P)

with  $\lambda_{k+1}$  such that  $J(X'_k) = 0$ .

## C DT-ELBO ON RIEMANNIAN MANIFOLD

In this section, we derive the DT-ELBO variational bounds of KL-divergence between the initial data distribution and the generated data distribution on the Riemannian manifold  $\Sigma := \{x \in \mathbb{R}^d \mid h(x) = 0, g(x) \leq 0\}$ .

Let  $\{x_0, \dots, x_N\} \subset \Sigma$  be the forward position trajectory produced by  $N$  steps of a discretized sampler, and let  $q(x_{k+1} \mid c_k)$  and  $p_\theta(x_k \mid d_k)$  denote the forward and backward transition densities, respectively. The context  $c_k, d_k$  depends on the history  $\{x_0, \dots, x_k\}$ : for the overdamped case we take  $c_k = \{x_k\}$ ,  $d_k = \{x_{k+1}\}$ , while for the (collapsed) underdamped case we use  $c_k = \{x_k, x_{k-1}\}$  (with  $c_0 = \{x_0, p_0\}$ ) and  $d_k = \{x_{k+1}, x_{k+2}\}$  (with  $d_{N-1} = \{x_N, p_N\}$ ). We set  $q_0(x) = q_{\text{data}}(x)$  as the data distribution and  $p(x) = p_N(x)$  as the prior. In what follows, all densities are understood with respect to the surface measure  $d\sigma_\Sigma$  on  $\Sigma$ , and, for clarity of exposition, we will assume that Lagrangian multiplier methods (OLLA-P, ULLA-P) are used each iterate to satisfy  $x_k \in \Sigma$ ; see the remark below for when this may fail under discretization.

**Remark 4** (On constraint enforcement and well-definedness of the DT-ELBO). The proposed discretized constrained Langevin dynamics with landing does *not* automatically guarantee  $x_k \in \Sigma$  at every step. By contrast, the RDDPM formulation [34] enforces feasibility at each time by solving the Lagrange multiplier system via Newton’s method (see Proposition B.1, Proposition B.2) so that

$$J(X_t) = 0 \quad \text{with} \quad \nabla J(X_t)^T P_t = 0 \quad (\text{if underdamped})$$

ensuring  $X_t \in \Sigma$  and  $P_t \in T_{X_t} \Sigma$  exactly. Throughout our DT-ELBO derivation we adopt the same *feasible-trajectory assumption*: we assume the multiplier solve succeeds by Newton’s method so that the sampled forward path lies on  $\Sigma$ , i.e.,  $x_k \in \Sigma$  for all  $k \in \{1, \dots, N\}$ .

This assumption is not merely cosmetic. If some  $x_k \notin \Sigma$ , the conditional densities that appear in the ELBO may be undefined or degenerate (e.g., in the overdamped case  $p_\theta(x_k \mid x_{k+1})/q(x_{k+1} \mid x_k)$ , and in the underdamped case  $p_\theta(x_{k-1} \mid x_k, x_{k+1})/q(x_{k+1} \mid x_k, x_{k-1})$ ). Such off-manifold iterates can induce singularities in the ELBO and render the induced NLL numerically unstable even under small constraint violations. For this reason, in our experiments we do not report test NLL; instead, we evaluate with task-specific metrics which reflect downstream performance.

To solidify our framework, we later introduce the Conditional Wasserstein Path Matching (CWPM) framework in Appendix D, which drops the feasibility-trajectory assumption. Its training loss has the same DT-ELBO form up to the choice of training loss weights, justifying that the CWPM framework shares the same principle with DT-ELBO framework in the constrained Langevin dynamics with landing as the step size  $\Delta t \rightarrow 0$ .

**DT-ELBO for overdamped Langevin** From the Markov property, the densities  $q(x_0, x_{1:N})$ ,  $p_\theta(x_{1:N} \mid x_0)$  are given by

$$q(x_1, \dots, x_N \mid x_0) = \prod_{k=0}^{N-1} q(x_{k+1} \mid x_k), \quad p_\theta(x_0, x_{1:N}) = p(x_N) \prod_{k=0}^{N-1} p_\theta(x_k \mid x_{k+1})$$

The common goal suggested in DDPM [1] (Euclidean space) or RDDPM [34] (Riemannian manifold) is to minimize  $\text{KL}^\Sigma(q_0 \parallel p_\theta)$ . For this, we observe that

$$\text{KL}^\Sigma(q_0(x_0) \parallel p_\theta(x_0)) = \int q_0(x_0) \ln q_0(x_0) d\sigma_\Sigma(x_0) - \underbrace{\int q_0(x_0) \ln \left( \int p_\theta(x_{0:N}) d\sigma_\Sigma(x_{1:N}) \right) d\sigma_\Sigma(x_0)}_{\text{Term (1)}}$$

1998 and

$$\begin{aligned}
\text{Term (1)} &= \int q_0(x_0) \ln \left( \int \frac{p_\theta(x_{0:N})}{q(x_{1:N}|x_0)} q(x_{1:N}|x_0) d\sigma_\Sigma(x_{1:N}) \right) d\sigma_\Sigma(x_0) \\
&\stackrel{\text{Jensen}}{\geq} \int q_0(x_0) q(x_{1:N}|x_0) \ln \left( \frac{p_\theta(x_{0:N})}{q(x_{1:N}|x_0)} \right) d\sigma_\Sigma(x_{1:N}) \\
&= \int q(x_{0:N}) \left( \sum_{k=0}^{N-1} \ln \frac{p_\theta(x_k|x_{k+1})}{q(x_{k+1}|x_k)} + \ln p(x_N) \right) d\sigma_\Sigma(x_{1:N})
\end{aligned}$$

2007 Because we want to minimize  $\text{KL}(q_0(x_0)||p_\theta(x_0))$ , the training loss  $L^{\text{over}}(\theta)$  can be set to

$$L^{\text{over}}(\theta) = - \int q(x_{0:N}) \sum_{k=0}^{N-1} \ln p_\theta(x_k|x_{k+1}) d\sigma_\Sigma(x_{1:N}) = -\mathbb{E}_{q(x_{0:N})} \left[ \sum_{k=0}^{N-1} \ln p_\theta(x_k|x_{k+1}) \right]$$

2012 where the product surface measure is defined to be  $d\sigma_\Sigma(x_1, \dots, x_N) := \prod_{k=1}^N d\sigma_\Sigma(x_k)$ .

2014 **Lemma C.1** (Backward transition density – overdamped [34]). *Suppose  $x_{k+1} \in \text{int}(\Sigma)$  and assume the followings hold:*

- 2016 1. *There exists a measurable set  $\mathcal{F}_{x_{k+1}} \subset T_{x_{k+1}}\Sigma$  such that, for every  $\eta \in \mathcal{F}_{x_{k+1}}$ , the Newton's method returns a unique pair  $(x, \lambda)$ ,  $x \in \text{int}(\Sigma)$  solving*

$$x = x_{k+1} + \mu_{k+1}^o(x_{k+1}) + \sigma_{k+1} \sqrt{\Delta t} \eta + \nabla J(x_{k+1}) \lambda, \quad \text{with } \lambda \text{ s.t. } J(x) = 0$$

2020 *with the minimal-displacement normal correction and it fails for  $\eta \notin \mathcal{F}_{x_{k+1}}$*

- 2022 2. *The solver success probability is  $1 - \epsilon_{x_{k+1}} := \mathbb{P}(\eta \in \mathcal{F}_{x_{k+1}}) \in (0, 1]$ .*

- 2024 3. *The map  $\Phi_{k+1} : \mathcal{F}_{x_{k+1}} \rightarrow \Sigma_{x_{k+1}} := \Phi_{k+1}(\mathcal{F}_{x_{k+1}}) \subset \text{int}(\Sigma)$  with  $\Phi_{k+1}(\eta) = x$  is a  $C^1$  bijection.*

2026 *Then, the backward transition density of OLLA-P with respect to surface measure  $d\sigma_\Sigma$  is given as*  
2027  $p_\theta(x_k|x_{k+1}) :$

$$p_\theta(x_k|x_{k+1}) = \frac{|\det(U(x_{k+1})^T U(x_k))|}{(2\pi\sigma_{k+1}^2 \Delta t)^{\frac{d-m}{2}} (1 - \epsilon_{x_{k+1}})} \exp \left( - \frac{\|\Pi(x_{k+1})(x_k - \mu_{k+1}^o(x_{k+1}))\|^2}{2\sigma_{k+1}^2 \Delta t} \right)$$

2032 *for  $x_k \in \Sigma_{x_{k+1}}$ , and  $p_\theta(x_k|x_{k+1}) = 0$  outside of  $\Sigma_{x_{k+1}}$ , where*

$$\mu_{k+1}^o(x_{k+1}) := x_{k+1} + \frac{1}{2} \sigma_{k+1}^2 \Delta t \Pi(x_{k+1}) \left[ \nabla f(x_{k+1}) + s_\theta^{(k+1)}(x_{k+1}) \right]$$

2036 *is the backward mean vector, and  $U(x)$  is the orthonormal matrix whose column vectors form an*  
2037 *orthonormal basis of  $T_x \Sigma$  so that  $U(x)U(x)^T = \Pi(x)$*

2039 *Proof.* Recall that the backward discretization of OLLA-P is given as below:

$$\begin{aligned}
x_k &= x_{k+1} + \frac{1}{2} \sigma_{k+1}^2 \Pi(x_{k+1}) \left[ \nabla f(x_{k+1}) + 2s_\theta^{(k+1)}(x_{k+1}) \right] \Delta t + \sigma_{k+1} \sqrt{\Delta t} \Pi(x_{k+1}) \zeta_{k+1} \\
&\quad + \nabla J(x_{k+1}) \lambda_{k+1}
\end{aligned}$$

2044 with  $\lambda_{k+1}$  such that  $J(x_k) = 0$ .

2045 Let  $\eta \sim \mathcal{N}(0, I_{d-m})$  on  $T_{x_{k+1}}\Sigma$ . Conditioning on the success event  $\{\eta \in \mathcal{F}_{x_{k+1}}\}$ , the conditional  
2046 density of  $\eta$  is given by

$$\phi(\zeta) = \frac{1}{(2\pi)^{\frac{d-m}{2}} (1 - \epsilon_{x_{k+1}})} \exp \left( -\frac{1}{2} \|\eta\|^2 \right) \mathbb{1}_{\eta \in \mathcal{F}_{x_{k+1}}}$$

2050 From the assumption, for each  $\eta \in \mathcal{F}_{x_{k+1}}$ , there is a unique  $x = \Phi_{k+1}(\eta) \in \Sigma$  solving

$$x = \mu_{k+1}^o(x_{k+1}) + \sigma_{k+1} \sqrt{\Delta t} \eta + \nabla J(x_{k+1}) \lambda$$

with  $\lambda$  such that  $J(x) = 0$ . Because  $\Phi_{k+1}$  is bijection from  $\mathcal{F}_{x_{k+1}} \subset T_{x_{k+1}}\Sigma$  onto  $\Sigma_{x_{k+1}} := \Phi_{k+1}(\mathcal{F}_{x_{k+1}}) \subset \Sigma$ , we can define  $G_{x_{k+1}} := \Phi_{k+1}^{-1}$  and its Jacobian is given by  $DG_{x_{k+1}}(x) = (\sigma_{k+1}\sqrt{\Delta t})^{-1}U(x_{k+1})^T U(x)$  for  $x \in \Sigma_{x_{k+1}}$ . Hence, we have

$$|\det(DG_{x_{k+1}}(x))| = (\sigma_{k+1}\sqrt{\Delta t})^{-(d-m)} |\det(U(x_{k+1})^T U(x))|$$

Lastly, we observe that, for  $x \in \Sigma_{x_{k+1}}$ , the pushforward of the conditional density  $\phi$  by  $\Phi_{k+1}$  yields the following density with respect to  $d\sigma_\Sigma$ :

$$p_\theta(x_k|x_{k+1}) = \frac{|\det(U(x_{k+1})^T U(x_k))|}{(2\pi\sigma_{k+1}^2\Delta t)^{\frac{d-m}{2}}(1-\epsilon_{x_{k+1}})} \exp\left(-\frac{\|\Pi(x_{k+1})^T(x_k - \mu_{k+1}^o(x_{k+1}))\|^2}{2\sigma_{k+1}^2\Delta t}\right)$$

which becomes zero outside  $\Sigma_{x_{k+1}}$  (equivalently, when projection failure happens).  $\square$

Therefore, assuming the forward path trajectory  $\{x_0, \dots, x_N\} \subset \text{int}(\Sigma)$ , the training loss has the following upper bound:

$$\begin{aligned} L^{\text{over}}(\theta) &= -\mathbb{E}_{q(x_{0:N})} \left[ \sum_{k=0}^{N-1} \ln p_\theta(x_k|x_{k+1}) \right] && \text{(Training loss-OLLA-P)} \\ &\leq \mathbb{E}_{q(x_{0:N})} \left[ \sum_{k=0}^{N-1} \left( \frac{\|\Pi(x_{k+1})^T(x_k - \mu_{k+1}^o(x_{k+1}))\|^2}{2\sigma_{k+1}^2\Delta t} \right) \right] + C \end{aligned}$$

where  $C := \sum_{k=0}^{N-1} \left[ -\ln \left( \frac{|\det(U(x_{k+1})^T U(x_k))|}{(2\pi\sigma_{k+1}^2\Delta t)^{\frac{d-m}{2}}} \right) \right]$  is constant with respect to  $\theta$  and the last inequality is obtained using  $\epsilon_{x_{k+1}} \leq 1$ .

**DT-ELBO for underdamped Langevin** In the collapsed underdamped setting, we keep only positions, which makes the forward chain a second-order Markov chain in  $\{x_0, \dots, x_N\}$ . Similarly, applying the same Jensen inequality argument used for the overdamped case gives the ELBO:

$$\begin{aligned} \text{Term (1)} &= \int q_0(x_0) \ln \left( \int \frac{p_\theta(x_{0:N})}{q(x_{1:N}|x_0, p_0)} q(p_0|x_0) q(x_{1:N}|x_0, p_0) d\sigma_{T_{x_0}\Sigma}(p_0) d\sigma_\Sigma(x_{1:N}) \right) d\sigma_\Sigma(x_0) \\ &\stackrel{\text{Jensen}}{\geq} \int q(x_0, p_0) q(x_{1:N}|x_0, p_0) \ln \left( \frac{p_\theta(x_{0:N})}{q(x_{1:N}|x_0, p_0)} \right) d\sigma_{T_{x_0}\Sigma}(p_0) d\sigma_\Sigma(x_{0:N}) \\ &\stackrel{\text{Jensen}}{\geq} \mathbb{E}_{q(x_{0:N}, p_0)} \mathbb{E}_{p(p_N|x_N)} \left[ \sum_{k=1}^{N-1} \ln \left( \frac{p_\theta(x_{k-1}|x_k, x_{k+1})}{q(x_{k+1}|x_k, x_{k-1})} \right) + \ln \left( \frac{p_\theta(x_{N-1}|x_N, p_N)}{q(x_1|x_0, p_0)} \right) + \ln p(x_N) \right] \end{aligned}$$

where the last inequality holds because  $p_\theta(x_{0:N})$  has the following form:

$$p_\theta(x_{0:N}) = p(x_N) \int p(p_N|x_N) p_\theta(x_{N-1}|x_N, p_N) \prod_{k=1}^{N-1} p_\theta(x_{k-1}|x_k, x_{k+1}) d\sigma_{T_{x_N}\Sigma}(p_N)$$

Therefore, the training loss  $L^{\text{under}}(\theta)$  is naturally given as follows to minimize  $\text{KL}^\Sigma(q_0(x_0)||p_\theta(x_0))$ :

$$\begin{aligned} L^{\text{under}}(\theta) &= -\mathbb{E}_{q(x_{0:N}, p_0)} \mathbb{E}_{\rho_N(p_N|x_N)} \left[ \sum_{k=0}^{N-2} \ln p_\theta(x_k|x_{k+1}, x_{k+2}) + \ln p_\theta(x_{N-1}|x_N, p_N) \right] \\ &= -\mathbb{E}_{q(x_{0:N}, p_0)} \mathbb{E}_{\rho_N(p_N|x_N)} \left[ \sum_{k=0}^{N-1} \ln p_\theta(x_k|x_{k+1}, x_{k+2}) \right] \end{aligned}$$

where  $\rho_N(p_N|x_N) \propto \exp\left(-\frac{1}{2}\|\Pi(x_N)p_N\|_2^2\right)$  is the density of the momentum prior density, and we used the notation abusing  $x_{N+1} := p_N$  for notational convenience.

**Lemma C.2** (Backward transition density – underdamped). *Suppose  $x_{k+1} \in \text{int}(\Sigma)$  and assume the followings hold:*

1. *There exists a measurable set  $\mathcal{F}_{x_{k+1}} \subset T_{x_{k+1}}\Sigma$  such that, for every  $\eta \in \mathcal{F}_{x_{k+1}}$ , the Newton’s method returns a unique pair  $(x, \lambda)$ ,  $x \in \text{int}(\Sigma)$  solving*

$$x = \mu_{k+1}^u(x_{k+1}, x_{k+2}) + \sigma_{k+1}^2 \Delta t \sqrt{1 - a_{k+1}^2} \eta + \nabla J(x_{k+1}) \lambda, \text{ with } \lambda \text{ s.t. } J(x) = 0$$

*with the minimal-displacement normal correction and it fails for  $\eta \notin \mathcal{F}_{x_{k+1}}$*

2. *The solver success probability is  $1 - \epsilon_{x_{k+1}} := \mathbb{P}(\eta \in \mathcal{F}_{x_{k+1}}) \in (0, 1]$ .*
3. *The map  $\Phi_{k+1} : \mathcal{F}_{x_{k+1}} \rightarrow \Sigma_{x_{k+1}} := \Phi_{k+1}(\mathcal{F}_{x_{k+1}}) \subset \text{int}(\Sigma)$  with  $\Phi_{k+1}(\eta) = x$  is a  $C^1$  bijection.*

*Then, the backward transition density of ULLA-P with respect to surface measure  $d\sigma_\Sigma$  is given as  $p_\theta(x_k|x_{k+1})$ :*

$$p_\theta(x_k|x_{k+1}) = \frac{|\det(U(x_{k+1})^T U(x_k))|}{(2\pi\sigma_{k+1}^2 \Delta t)^{\frac{d-m}{2}} (1 - \epsilon_{x_{k+1}})} \exp\left(-\frac{\|\Pi(x_{k+1})^T (x_k - \mu_{k+1}^u(x_{k+1}, x_{k+2}))\|^2}{2\sigma_{k+1}^4 \Delta t^2 (1 - a_{k+1})^2}\right)$$

*for  $x_k \in \Sigma_{x_{k+1}}$ , and  $p_\theta(x_k|x_{k+1}) = 0$  outside of  $\Sigma_{x_{k+1}}$ , where*

$$\mu_{k+1}^u = x_{k+1} + \sigma_{k+1}^2 \Delta t \Pi(x_{k+1}) [a_{k+1} \tilde{p}_{k+1} + \sigma_{k+1}^2 \Delta t (\nabla f(x_{k+1}) + s_{k+1}^\theta(x_{k+1}, \tilde{p}_{k+1}))]$$

*is the backward mean vector.*

*Proof.* Recall that the backward discretization of ULLA-P is given as below:

$$x_k = x_{k+1} + \sigma_{k+1}^2 \Delta t \sqrt{1 - a_{k+1}^2} \Pi(x_{k+1}) \zeta_{k+1} + \nabla J(x_{k+1}) \lambda_{k+1} + \sigma_{k+1}^2 \Delta t \Pi(x_{k+1}) [a_{k+1} \tilde{p}_{k+1} + \sigma_{k+1}^2 (\nabla f(x_{k+1}) + s_{k+1}^\theta(x_{k+1}, \tilde{p}_{k+1})) \Delta t]$$

with  $\lambda_{k+1}$  such that  $J(x_k) = 0$ . Using the same proof as the overdamped case, we observe that  $\Phi_{k+1}$  is bijection from  $\mathcal{F}_{x_{k+1}} \subset T_{x_{k+1}}\Sigma$  onto  $\Sigma_{x_{k+1}} := \Phi_{k+1}(\mathcal{F}_{x_{k+1}}) \subset \Sigma$ , and we can define  $G_{x_{k+1}} := \Phi_{k+1}^{-1}$  whose Jacobian is given by  $DG_{x_{k+1}}(x) = (\sigma_{k+1} \sqrt{\Delta t})^{-1} U(x_{k+1})^T U(x)$  for  $x \in \Sigma_{x_{k+1}}$ . Therefore, the determinant of this is:

$$|\det(DG_{x_{k+1}}(x))| = (\sigma_{k+1} \sqrt{\Delta t})^{-(d-m)} |\det(U(x_{k+1})^T U(x))|$$

Lastly, we observe that, for  $x \in \Sigma_{x_{k+1}}$ , the pushforward of the conditional density of  $\eta$  (defined on the overdamped case proof) by  $\Phi_{k+1}$  yields the following density with respect to  $d\sigma_\Sigma$ :

$$p_\theta(x_k|x_{k+1}) = \frac{|\det(U(x_{k+1})^T U(x_k))|}{(2\pi\sigma_{k+1}^2 \Delta t)^{\frac{d-m}{2}} (1 - \epsilon_{x_{k+1}})} \exp\left(-\frac{\|\Pi(x_{k+1})^T (x_k - \mu_{k+1}^u(x_{k+1}, x_{k+2}))\|^2}{2\sigma_{k+1}^4 \Delta t^2 (1 - a_{k+1})^2}\right)$$

which becomes zero outside  $\Sigma_{x_{k+1}}$  (equivalently, when projection failure happens).  $\square$

Therefore, assuming the forward path trajectory  $\{x_0, \dots, x_N\} \subset \text{int}(\Sigma)$ , the training loss has the following upper bound (with notation  $x_{N+1} := p_N$ ):

$$\begin{aligned} L^{\text{under}}(\theta) &= -\mathbb{E}_{q(x_{0:N}, p_0)} \mathbb{E}_{\rho_N(p_N|x_N)} \left[ \sum_{k=0}^{N-1} \ln p_\theta(x_k|x_{k+1}, x_{k+2}) \right] \quad (\text{Training loss-ULLA-P}) \\ &\leq \mathbb{E}_{q(x_{0:N})} \mathbb{E}_{\rho_N(p_N|x_N)} \left[ \sum_{k=0}^{N-1} \left( \frac{\|\Pi(x_{k+1})^T (x_k - \mu_{k+1}^u(x_{k+1}, x_{k+2}))\|^2}{2\sigma_{k+1}^4 \Delta t^2 (1 - a_{k+1})^2} \right) \right] + C \end{aligned}$$

where  $C := \sum_{k=0}^{N-1} \left[ -\ln \left( \frac{|\det(U(x_{k+1})^T U(x_k))|}{(2\pi\sigma_{k+1}^2 \Delta t)^{\frac{d-m}{2}}} \right) \right]$  is constant with respect to  $\theta$  and the last inequality is again obtained using  $\epsilon_{x_{k+1}} \leq 1$ .

## D CONDITIONAL WASSERSTEIN PATH MATCHING (CWPM)

Let  $q_0, q_1, \dots, q_N$  be the marginal forward probability densities at each step, evolved by discrete forward transition kernels  $Q_k = q(x_{k+1}|c_k)$ . Similarly, define  $p_N, p_{N-1}^\theta, \dots, p_0^\theta$  to be the marginal backward probability densities driven by parameterized backward transition kernels  $T_{k+1}^\theta = p^\theta(x_k|d_k)$ . Similarly as in DT-ELBO, the context vector is fixed to be  $c_k = \{x_k\}$ ,  $d_k = \{x_{k+1}\}$  for the overdamped case, while for the (collapsed) underdamped case we use  $c_k = \{x_k, x_{k-1}\}$  (with  $c_0 = \{x_0, p_0\}$ ) and  $d_k = \{x_{k+1}, x_{k+2}\}$  (with  $d_{N-1} = \{x_N, p_N\}$ ).

Our goal is to minimize  $W_2(q_0, p_0^\theta)$  so that the Wasserstein-2 distance between data distribution and generated data distribution becomes close to each other.

**CWPM framework for the overdamped** We first define **circuitous** density at step  $k$  as

$$\sigma_k := q_k T_k^\theta T_{k-1}^\theta, \dots, T_1^\theta \quad \text{for } k \in \{1, \dots, N\} \quad \sigma_0 := q_0.$$

We assume that for any probability measure  $\mu, \nu$  in  $\mathbb{R}^d$ , there exists  $K_{k+1} > 0$  such that

$$W_2(\mu T_{k+1}^\theta, \nu T_{k+1}^\theta) \leq K_{k+1} W_2(\mu, \nu) + \mathcal{O}(\sqrt{\Delta t}) \quad (\text{stepwise-Lipschitz})$$

Under this assumption, we can choose  $\Lambda_{k+1} > 0$  such that

$$W_2(\sigma_k, \sigma_{k+1}) \leq \Lambda_{k+1} W_2(q_k, q_{k+1} T_{k+1}^\theta) + \mathcal{O}(\sqrt{\Delta t})$$

for  $k \in \{0, \dots, N-1\}$ . We note that [Lemma D.1](#) implies that such  $\Lambda_k$  exists without stepwise-Lipschitz assumption when the function class of score function and the constraint functions are sufficiently regular.

Under this setup, from the triangular inequality of  $W_2$ , it holds that

$$\begin{aligned} W_2(q_0, p_0^\theta) &\leq W_2(q_0, \sigma_N) + W_2(\sigma_N, p_0^\theta) \leq \sum_{k=0}^{N-1} W_2(\sigma_k, \sigma_{k+1}) + W_2(\sigma_N, p_0^\theta) \\ &\leq \sum_{k=0}^{N-1} \Lambda_{k+1} W_2(q_k, q_{k+1} T_{k+1}^\theta) + \Lambda_{N+1} W_2(q_N, p_N) + \mathcal{O}(\sqrt{\Delta t}) \\ &\leq \sum_{k=0}^{N-1} \Lambda_{k+1} \mathbb{E}_{x_{k+1} \sim q_{k+1}} [W_2(q_{k|k+1}(\cdot|x_{k+1}), T_{k+1}^\theta(\cdot|x_{k+1}))] + \Lambda_{N+1} W_2(q_N, p_N) + \mathcal{O}(\sqrt{\Delta t}) \end{aligned}$$

Now, we know that  $T_{k+1}^\theta(\cdot|x_{k+1}) = \mathcal{N}(\mu_{k+1}^\theta(x_{k+1}), \sigma_{k+1}^2 \Pi(x_{k+1}))$  and let  $(\mu_{k|k+1}, \Sigma_{k|k+1})$  be the mean and covariance of  $q_{k-1|k}$ . Then, from the relation between Gelbrich distance and 2-Wasserstein distance [38, 37], we have:

$$\begin{aligned} &\mathbb{E}_{x_{k+1} \sim q_{k+1}} W_2(q_{k|k+1}, T_{k+1}^\theta(\cdot|x_{k+1}))^2 = \mathbb{E}_{x_{k+1} \sim q_{k+1}} \|\mu_{k|k+1}(x_{k+1}) - \mu_{k+1}^\theta(x_{k+1})\|^2 \\ &\quad + \mathbb{E}_{x_{k+1} \sim q_{k+1}} [\text{Tr}(\Sigma_{k|k+1}(x_{k+1})) + \text{Tr}(\sigma_{k+1}^2 \Pi(x_{k+1}))] + \mathbb{E}_{x_{k+1} \sim q_{k+1}} \Delta_{k+1}(x_{k+1}) \\ &\quad - 2 \mathbb{E}_{x_{k+1} \sim q_{k+1}} \left[ \text{Tr} \left( \left( \Sigma_{k|k+1}^{1/2}(x_{k+1}) \sigma_{k+1}^2 \Pi(x_{k+1}) \Sigma_{k|k+1}^{1/2}(x_{k+1}) \right)^{1/2} \right) \right] \\ &= \mathbb{E}_{x_k \sim q_k, x_{k+1} \sim q_{k+1}} \|x_k - \mu_{k+1}^\theta(x_{k+1})\|^2 + \text{Tr}(\sigma_{k+1}^2 \Pi(x_{k+1})) + \mathbb{E}_{x_{k+1} \sim q_{k+1}} \Delta_{k+1}(x_{k+1}) \\ &\quad - 2 \mathbb{E}_{x_{k+1} \sim q_{k+1}} \left[ \text{Tr} \left( \left( \Sigma_{k|k+1}^{1/2}(x_{k+1}) \sigma_{k+1}^2 \Pi(x_{k+1}) \Sigma_{k|k+1}^{1/2}(x_{k+1}) \right)^{1/2} \right) \right] \end{aligned}$$

where we used the law of total variance for the last equality, by observing that  $x_k = \mu_{k|k+1}(x_{k+1}) + \epsilon \sim q_k$  with  $\mathbb{E}[\epsilon|x_{k+1}] = 0$  and  $\text{Cov}(\epsilon|x_{k+1}) = \Sigma_{k|k+1}$ . Note that  $\Delta_{k+1}(x_{k+1}) \geq 0$  is the distance gap between Gelbrich distance and 2-Wasserstein distance (independent of  $\theta$ ) so that it becomes zero if the true conditional  $x_k|x_{k+1}$  follows the Gaussian distribution as

$$x_k|x_{k+1} \sim \mathcal{N}(\mu_{k|k+1}(x_{k+1}), \Sigma_{k|k+1}(x_{k+1}))$$

**Training loss (Overdamped) from CWPM** We note that from the above bound, minimizing  $\mathbb{E}_{x_k \sim q_k, x_{k+1} \sim q_{k+1}} \|x_k - \mu_{k+1}^\theta(x_{k+1})\|^2$  is to close the Wasserstein distance between  $q_0, p_0^\theta$ . Because  $\|x_k - \mu_{k+1}^\theta(x_{k+1})\|^2$  can be decomposed into

$$\|x_k - \mu_{k+1}^\theta(x_{k+1})\|^2 = \|\Pi(x_{k+1})(x_k - \mu_{k+1}^\theta(x_{k+1}))\|^2 + \underbrace{\|(I - \Pi(x_{k+1}))(x_k - \mu_{k+1}^\theta(x_{k+1}))\|^2}_{\text{constant w.r.t } \theta}$$

and  $\mu_k^o$  does not have  $\theta$  dependency on normal (second) term, the natural choice of loss (leveraging the saved forward trajectories) is

$$\begin{aligned} L_{\text{CWPM}}^{\text{over}}(\theta) &= \mathbb{E}_{q(x_{0:N})} \left[ \sum_{k=0}^{N-1} \lambda(k) \|\Pi(x_{k+1})(x_k - \mu_k^o(x_{k+1}))\|^2 \right] \\ &= \mathbb{E}_{q(x_{0:N})} \left[ \sum_{k=0}^{N-1} \frac{\|\Pi(x_{k+1})(x_k - \mu_k^o(x_{k+1}))\|^2}{2\sigma_{k+1}^2 \Delta t} \right] \end{aligned}$$

with some training loss weight  $\lambda(k)$ . We note that other seminar works [1, 39, 40] in diffusion model choose the weight proportional to the inverse of variance of corresponding term, which, in our case, becomes  $\lambda(k) = \frac{1}{2\sigma_{k+1}^2 \Delta t}$  with proportional constant 1/2. And, notably, this leads to the exactly the same training loss provided in DT-ELBO, (Lemma C.1) without the requirement  $x_k \in \Sigma$ .

**Lemma D.1** (Sufficient condition for  $\Lambda_k < \infty$  – Overdamped). *Let the one-step landing backward update of OLLA (Equation Backward-OLLA) be*

$$F_\theta^k(y, \zeta) = y + \Pi(y)b_\theta^k(y)\Delta t + \sigma_k \sqrt{\Delta t} \Pi(y)\zeta + \sigma_k^2 \phi(y)\Delta t, \quad \zeta \sim \mathcal{N}(0, I_d)$$

where  $b_\theta^k := \frac{\sigma_k^2}{2} [\nabla f(y) + 2s_\theta^k(y)]$  is the drift term and  $\phi(y)$  is the normal term. Assume:

- (Regularity of constraint functions) There exists constants  $c_\phi^0, c_\phi^1 < \infty$  such that for any square-integrable  $Y$ ,

$$\mathbb{E}\|\phi(Y)\|^2 \leq c_\phi^0 + c_\phi^1 \mathbb{E}\|Y\|^2$$

- (Regularity of function class) There exists  $L_s, B_s < \infty$  independent of  $\theta$  with

$$\|s_\theta^k(x) - s_\theta^k(y)\| \leq L_s \|x - y\|, \quad \|s_\theta^k(x)\| \leq B_s + L_s \|x\|$$

for all  $k \in \{1, \dots, N\}$  and assume  $\nabla f$  is Lipschitz with constant  $L_f$  so that

$$\|b_\theta^k(x) - b_\theta^k(y)\| \leq L_b \|x - y\|, \quad \|b_\theta^k(x)\| \leq C(1 + \|x\|)$$

for some constant  $L_b, C, k \in \{1, \dots, N\}$

Let  $T_k^\theta$  be the associated Markov kernel to  $F_\theta^k$ . That is,  $T_k^\theta(\cdot|y) = \text{Law}(F_\theta^k(y, \zeta))$ . Then, for any probability measures  $\mu, \nu$  in  $\mathbb{R}^d$ , we have

$$W_2(\mu T_k^\theta, \nu T_k^\theta) \leq K_k W_2(\mu, \nu) + \mathcal{O}\left(\sqrt{\Delta t} + \Delta t \left(1 + \sqrt{\mathbb{E}_{Y \sim \mu} \|Y\|^2} + \sqrt{\mathbb{E}_{Y' \sim \nu} \|Y'\|^2}\right)\right)$$

for some constant  $K_k$ . Also, if  $\mu$  is given as  $\mu = \rho T_i^\theta T_{i-1}^\theta \dots T_j^\theta$  for  $i \geq j$  and some density  $\rho$  independent of  $\theta$ , the supremum of second moment of  $\mu$  is finite under  $\theta \in \Theta$ :

$$\sup_{\theta \in \Theta} \mathbb{E}_{Y_\mu \sim \mu} \|Y_\mu\|^2 < \infty$$

Combining these two, we can guarantee the existence of  $\Lambda_k > 0$  such that

$$W_2(\sigma_{k-1}, \sigma_k) \leq \Lambda_k W_2(q_{k-1}, q_k T_k^\theta) + \mathcal{O}(\sqrt{\Delta t})$$

for  $k \in \{1, \dots, N\}$ .

*Proof.* Let  $(Y, Y')$  be the synchronous coupling for  $W_2(\mu, \nu)$  with shared noise  $\zeta \sim \mathcal{N}(0, I)$ . Set

$$\begin{aligned} \Delta F_\theta^k &:= F_\theta^k(Y, \zeta) - F_\theta^k(Y', \zeta) = (Y - Y') + \underbrace{\Delta t (\Pi(Y)b_\theta^k(Y) - \Pi(Y')B_\theta^k(Y'))}_{:=B} \\ &\quad + \underbrace{\sigma_k \sqrt{\Delta t} (\Pi(Y) - \Pi(Y'))\zeta}_{:=N} + \underbrace{\sigma_k^2 \Delta t (\phi(Y) - \phi(Y'))}_{:=L} \end{aligned}$$

2268 Then, using  $(a + b + c + d)^2 \leq 2(a^2 + b^2 + c^2 + d^2)$ , we have:

$$2269 \mathbb{E}\|\Delta F_\theta^k\|^2 \leq 2\mathbb{E}\|Y - Y'\|^2 + 2\mathbb{E}\|B\|^2 + 2\mathbb{E}\|N\|^2 + 2\mathbb{E}\|L\|^2$$

2271 Now, we note that

$$2272 B = \Delta t (\Pi(Y)(b_\theta(Y) - b_\theta(Y')) + (\Pi(Y) - \Pi(Y')) b_\theta(Y'))$$

2274 and it implies

$$2275 \|B\| \leq \Delta t (L_b \|Y - Y'\| + 2\|b_\theta(Y')\|)$$

2276 Therefore, we have

$$2277 \mathbb{E}\|B\|^2 \leq 2L_b^2 \Delta t \mathbb{E}\|Y - Y'\|^2 + 8C \Delta t^2 (1 + \mathbb{E}\|Y'\|^2)$$

$$2278 = 2L_b^2 \Delta t \mathbb{E}\|Y - Y'\|^2 + 4C \Delta t^2 (1 + \mathbb{E}\|Y'\|^2 + \mathbb{E}\|Y\|^2)$$

2280 where the last equality comes by swapping  $Y, Y'$  and taking average. Also, we note that

$$2282 \mathbb{E}\|N\|^2 = \sigma_k^2 \Delta t \mathbb{E}[\|(\Pi(Y) - \Pi(Y')) \zeta\|] \leq 4\sigma_k^2 \Delta t \mathbb{E}\|\zeta\|^2$$

2283 using  $\|\Pi\| \leq 1$ . Similarly, we observe that

$$2285 \mathbb{E}\|L\|^2 \leq 2\sigma_k^4 \Delta t^2 (\mathbb{E}\|\phi(Y)\|^2 + \mathbb{E}\|\phi(Y')\|^2) \leq 2\sigma_k^4 \Delta t^2 (2c_\phi^0 + c_\phi^1 (\mathbb{E}\|Y\|^2 + \mathbb{E}\|Y'\|^2))$$

2286 using  $(a - b)^2 \leq 2a^2 + 2b^2$ . By collecting all terms, we obtain

$$2288 \mathbb{E}\|\Delta F\|^2 \leq 2(1 + L_b^2 \Delta t) \mathbb{E}\|Y - Y'\|^2 + 8\sigma_k^2 \Delta t \mathbb{E}\|\zeta\|^2 + \Delta t^2 \tilde{C} (1 + \mathbb{E}\|Y\|^2 + \mathbb{E}\|Y'\|^2)$$

2289 for some  $\tilde{C} > 0$ . By taking square-root and using  $\sqrt{u + v + w} \leq \sqrt{u} + \sqrt{v} + \sqrt{w}$ , we get

$$2291 W_2(\mu T^\theta, \nu T^\theta) \leq K_k W_2(\mu, \nu) + \mathcal{O}(\sqrt{\Delta t} + \Delta t \left(1 + \sqrt{\mathbb{E}_{Y \sim \mu} \|Y\|^2} + \sqrt{\mathbb{E}_{Y' \sim \nu} \|Y'\|^2}\right))$$

2293 for some constant  $K_k$ . Also, following the similar algebraic techniques, one can show the following using the regularity assumptions:

$$2296 \mathbb{E}\|F_\theta^k(Y, \zeta)\|^2 \leq [1 + a_k \Delta t + \mathcal{O}(\Delta t^2)] \mathbb{E}\|Y\|^2 + b_k \Delta t + \mathcal{O}(\Delta t^2)$$

2298 for some  $a_k, b_k > 0$  independent of  $\theta$ . So, once  $\mu$  is given as  $\mu = \rho T_i^\theta T_{i-1}^\theta \dots T_j^\theta$  for  $i \geq j$  and some density  $\rho$  independent of  $\theta$ , then, by applying the recursive inequality above, we get:

$$2300 \mathbb{E}_{Y_\mu \sim \mu} \|Y_\mu\|^2 \leq \prod_{k=i}^j (1 + a_k \Delta t + \mathcal{O}(\Delta t^2)) \mathbb{E}_{Y_\rho \sim \rho} \|Y_\rho\|^2 + \mathcal{O}(\Delta t)$$

2303 and taking supremum over  $\theta$ , it implies that

$$2304 \sup_{\theta} \mathbb{E}_{Y_\mu \sim \mu} \|Y_\mu\|^2 < \infty$$

2307 because the constants and the density  $\rho$  is independent of  $\theta$ .  $\square$

2308 **CWPM framework for the underdamped** Let  $y_k := (x_k, x_{k+1}) \in \mathbb{R}^{2d}$  with  $x_k \sim q_k, x_{k+1} \sim$   
2309  $q_{k+1}$  and let  $\bar{q}_k$  be the law of  $y_k$ . The forward pair-kernel  $\bar{Q}_k$  and backward pair-kernel  $\bar{T}_{k+1}^\theta$  are

$$2311 \bar{Q}_k(x_{k+1}, x_{k+2} | x_k, x_{k+1}) = \delta_{x_{k+1}} \otimes Q_k(x_{k+2} | x_k, x_{k+1})$$

$$2312 \bar{T}_{k+1}^\theta(x_k, x_{k+1} | x_{k+1}, x_{k+2}) = T_{k+1}^\theta(x_k | x_k, x_{k+1}) \otimes \delta_{x_{k+1}}$$

2314 Now, we similarly define the **circuious** densities on pairs as

$$2315 \bar{\sigma}_0 := \bar{q}_0, \quad \bar{\sigma}_k := \bar{q}_k \bar{T}_k^\theta \dots \bar{T}_1^\theta, \quad \text{for } k \in \{1, \dots, N-1\}$$

2317 Assume the stepwise Lipschitz inequality on pairs holds such that there exists  $\bar{K}_{k+1} > 0$

$$2318 W_2(\mu \bar{T}_{k+1}^\theta, \nu \bar{T}_{k+1}^\theta) \leq \bar{K}_{k+1} W_2(\mu, \nu) + \mathcal{O}(\Delta t)$$

2319 for any probability measure  $\mu, \nu$  in  $\mathbb{R}^d$ . Then there exists finite  $\bar{\Lambda}_{k+1} > 0$  such that

$$2320 W_2(\bar{\sigma}_k, \bar{\sigma}_{k+1}) \leq \bar{\Lambda}_{k+1} W_2(\bar{q}_k, \bar{q}_{k+1} \bar{T}_{k+1}^\theta) + \mathcal{O}(\Delta t)$$

2321

for  $k \in \{0, \dots, N-2\}$ . (As in Lemma D.2) such  $\Lambda_k$  exists without stepwise-Lipschitz assumption under mild regularity of the score function class and constraints.

For the prior on pair chain setup, we let  $\bar{p}_{N-1}^\theta$  be a terminal pair prior on  $(X_{N-1}, X_N)$  induced by sampling  $X_N \sim p_N, P_N \sim \Pi(X_N)\zeta$  with  $\zeta \sim \mathcal{N}(0, I_d)$  so that  $X_{N-1} \sim \bar{T}_N^\theta(\cdot|X_N, P_N)$ . Then, we propagate backward by

$$\bar{p}_0^\theta := \bar{p}_{N-1} \bar{T}_{N-1}^\theta \cdots \bar{T}_1^\theta, \quad p_0^\theta := (\pi_1)_\# \bar{p}_0^\theta$$

where  $\pi_1(x_0, x_1) = x_0$  is the projection map onto first coordinate. Since  $\pi_1$  is 1-Lipschitz,  $W_2(q_0, p_0^\theta) \leq W_2(\bar{q}_0, \bar{p}_0^\theta)$  holds and, from the triangle inequality for  $W_2$ , we have

$$\begin{aligned} W_2(q_0, p_0^\theta) &\leq W_2(\bar{q}_0, \bar{p}_0^\theta) \leq W_2(\bar{q}_0, \bar{\sigma}_{N-1}) + W_2(\bar{\sigma}_{N-1}, \bar{p}_0^\theta) \\ &\leq \sum_{k=0}^{N-2} W_2(\bar{\sigma}_k, \bar{\sigma}_{k+1}) + W_2(\bar{\sigma}_{N-1}, \bar{p}_0^\theta) \\ &\leq \sum_{k=0}^{N-2} \bar{\Lambda}_k W_2(\bar{q}_k, \bar{q}_{k+1} \bar{T}_{k+1}^\theta) + \bar{\Lambda}_N W_2(\bar{q}_{N-1}, \bar{p}_{N-1}^\theta) + \mathcal{O}(\Delta t). \end{aligned}$$

Because in pair conditionals the second coordinate is a Dirac mass, the inner  $W_2$  reduces to a position-only conditional mismatch:

$$W_2(\bar{q}_k, \bar{q}_{k+1} \bar{T}_{k+1}^\theta) = \mathbb{E}_{(x_{k+1}, x_{k+2}) \sim \bar{q}_{k+1}} [W_2(q_{k|k+1}(\cdot|x_{k+1}, x_{k+2}), \bar{T}_{k+1}^\theta(\cdot|x_{k+1}, x_{k+2}))]$$

Also, we note that that the following decomposition holds by triangle inequality:

$$W_2(\bar{q}_{N-1}, \bar{p}_{N-1}^\theta) \leq W_2((q_N \otimes \rho_N)S_N, (p_N \otimes \rho_N)S_N) + W_2((p_N \otimes \rho_N)S_N, (p_N \otimes \rho_N)S_N^\theta)$$

where  $p_N$  is the prior of position,  $\rho_N(\cdot|X_N)$  is the prior of momentum defined by the law of  $\Pi(X_N)\zeta$  with  $\zeta \sim \mathcal{N}(0, I)$ ,  $x_N \sim p_N$ , and each  $S_N$  and  $S_N^\theta$  are defined by

$$S_N(x_N, p_N) := \bar{q}_{N-1|N}(\cdot|x_N, p_N) \otimes \delta_{x_N}, \quad S_N^\theta(x_N, p_N) := \bar{T}_N^\theta(\cdot|x_N, p_N) \otimes \delta_{x_N}$$

And, we observe that the first term is independent of  $\theta$ , and the second term is given by:

$$W_2((p_N \otimes \rho_N)S_N, (p_N \otimes \rho_N)S_N^\theta) = \mathbb{E}_{(x_N, p_N) \sim p_N \otimes \rho_N} [W_2^2(\bar{q}_{N-1|N}(\cdot|x_N, p_N), \bar{T}_N^\theta(\cdot|x_N, p_N))].$$

because the second coordinate is a Dirac delta. Therefore, we have the following bound:

$$\begin{aligned} W_2(q_0, p_0^\theta) &\leq \sum_{k=0}^{N-2} \bar{\Lambda}_k \mathbb{E}_{(x_k, x_{k+1}) \sim \bar{q}_k} [W_2(\bar{q}_{k|k+1}(\cdot|x_{k+1}, x_{k+2}), T_{k+1}^\theta(\cdot|x_{k+1}, x_{k+2}))] \\ &\quad + \bar{\Lambda}_N \mathbb{E}_{(X_N, p_N) \sim p_N \otimes \rho_N} [W_2^2(\bar{q}_{N-1|N}(\cdot|x_N, p_N), T_N^\theta(\cdot|x_N, p_N))] \\ &\quad + \bar{\Lambda}_N W_2((q_N \otimes \rho_N)S_N, (p_N \otimes \rho_N)S_N) + \mathcal{O}(\Delta t). \end{aligned}$$

Now, we recall that  $T_{k+1}^\theta(\cdot|x_{k+1}, x_{k+2}) = \mathcal{N}(\mu_{k+1}^u(x_{k+1}, x_{k+2}), \sigma_{k+1}^4 \Delta t^2 (1 - a_{k+1}^2) \Pi(x_{k+1}))$  and let  $(\mu_{k|k+1, k+2}, \Sigma_{k|k+1, k+2})$  be the mean and covariance of true one-step backward conditional  $\bar{q}_{k|k+1}$ . Then, from the relation between Gelbrich distance and 2-Wasserstein distance, we have:

$$\begin{aligned} &\mathbb{E}_{x_{k+1} \sim q_{k+1}, x_{k+2} \sim q_{k+2}} W_2(\bar{q}_{k|k+1}, T_{k+1}^\theta(\cdot|x_{k+1}, x_{k+2}))^2 \\ &= \mathbb{E}_{x_{k+1} \sim q_{k+1}, x_{k+2} \sim q_{k+2}} \|\mu_{k|k+1, k+2}(x_{k+1}, x_{k+2}) - \mu_{k+1}^u(x_{k+1}, x_{k+2})\|^2 \\ &\quad + \mathbb{E}_{x_{k+1} \sim q_{k+1}, x_{k+2} \sim q_{k+2}} [\text{Tr}(\Sigma_{k|k+1, k+2}(x_{k+1}, x_{k+2})) + \text{Tr}(\sigma_{k+1}^4 \Delta t^2 (1 - a_{k+1}^2) \Pi(x_{k+1}))] \\ &\quad + \mathbb{E}_{x_{k+1} \sim q_{k+1}, x_{k+2} \sim q_{k+2}} \Delta_{k+1}(x_{k+1}, x_{k+2}) - 2\sigma_{k+1}^4 \Delta t (1 - a_{k+1}^2) \\ &\quad \times \mathbb{E}_{x_{k+1} \sim q_{k+1}, x_{k+2} \sim q_{k+2}} \left[ \text{Tr} \left( \left( \Sigma_{k|k+1, k+2}^{1/2}(x_{k+1}, x_{k+2}) \Pi(x_{k+1}) \Sigma_{k|k+1, k+2}^{1/2}(x_{k+1}, x_{k+2}) \right)^{1/2} \right) \right] \\ &= \mathbb{E}_{x_k \sim q_k, x_{k+1} \sim q_{k+1}, x_{k+2} \sim q_{k+2}} \|X_k - \mu_k^u(x_{k+1}, x_{k+2})\|^2 + \sigma_{k+1}^4 \Delta t^2 (1 - a_{k+1}^2) \text{Tr}(\Pi(x_{k+1})) \\ &\quad + \mathbb{E}_{x_{k+1} \sim q_{k+1}, x_{k+2} \sim q_{k+2}} \Delta_{k+1}(x_{k+1}, x_{k+2}) \\ &\quad - 2\mathbb{E}_{x_{k+1} \sim q_{k+1}, x_{k+2} \sim q_{k+2}} \left[ \text{Tr} \left( \left( \Sigma_{k|k+1}^{1/2}(x_{k+1}) \sigma_{k+1}^2 \Pi(x_{k+1}) \Sigma_{k|k+1}^{1/2}(x_{k+1}) \right)^{1/2} \right) \right] \end{aligned}$$

where we used the law of total variance for the last equality, by observing that  $x_k = \mu_{k|k+1,k+2}(x_{k+1}, x_{k+2}) + \epsilon \sim q_k$  with  $\mathbb{E}[\epsilon|x_{k+1}, x_{k+2}] = 0$  and  $\text{Cov}(\epsilon|x_{k+1}, x_{k+2}) = \Sigma_{k|k+1,k+2}$ . Similar to overdamped case,  $\Delta_{k+1}(x_{k+1}, x_{k+2}) \geq 0$  is the distance gap between Gelbrich distance and 2-Wasserstein distance (independent of  $\theta$ ) so that it becomes zero if the true conditional  $x_k|x_{k+1}, x_{k+2}$  follows the Gaussian distribution as

$$x_k|x_{k+1}, x_{k+2} \sim \mathcal{N}(\mu_{k|k+1,k+2}(x_{k+1}, x_{k+2}), \Sigma_{k|k+1,k+2}(x_{k+1}, x_{k+2}))$$

**Training loss (Underdamped) from CWPM** Because  $\|x_k - \mu_k^u(x_{k+1}, x_{k+2})\|^2$  can be decomposed into

$$\begin{aligned} \|x_k - \mu_k^u(x_{k+1}, x_{k+2})\|^2 &= \|\Pi(x_{k+1})(x_k - \mu_k^u(x_{k+1}, x_{k+2}))\|^2 \\ &\quad + \underbrace{\|(I - \Pi(x_{k+1}))(x_k - \mu_k^u(x_{k+1}, x_{k+2}))\|^2}_{\text{constant w.r.t } \theta} \end{aligned}$$

where  $\mu_k^u$  does not have  $\theta$  dependency on normal (second) term. Therefore, by abusing notation to set  $p_N = x_{N+1}$ , the choice of training loss becomes

$$\begin{aligned} L_{\text{CWPM}}^{\text{under}}(\theta) &= \mathbb{E}_{q(x_{0:N})} \mathbb{E}_{\rho_N(p_N|x_N)} \left[ \sum_{k=0}^{N-1} \lambda(k) \|\Pi(x_{k+1})(x_k - \mu_k^u(x_{k+1}, x_{k+2}))\|^2 \right] \\ &= \mathbb{E}_{q(x_{0:N})} \mathbb{E}_{\rho_N(p_N|x_N)} \left[ \sum_{k=0}^{N-1} \frac{\|\Pi(x_{k+1})(x_k - \mu_k^u(x_{k+1}, x_{k+2}))\|^2}{2\sigma_{k+1}^4 \Delta t^2 (1 - a_{k+1}^2)} \right] \end{aligned}$$

with some training loss weight  $\lambda(k)$ . In our case, the training loss weight proportional to the inverse of variance can be chosen by  $\lambda(k) = \frac{1}{2\sigma_{k+1}^4 \Delta t^2 (1 - a_{k+1}^2)}$  with proportional constant  $1/2$ . And, notably, this leads to the exactly the same training loss provided in DT-ELBO, (Lemma C.2) without the requirement  $x_k \in \Sigma$ .

**Lemma D.2** (Sufficient condition for  $\Lambda_k < \infty$  – Underdamped). *Let the one-step landing backward update of ULLA (Equation Backward-ULLA) be*

$$\begin{aligned} \bar{F}_\theta^k(x_+, x_{++}, \zeta) &= x_+ - \sigma_k^2 \Delta t \Pi(x_+) [a_{k+1} \tilde{p} + b_\theta^k(x_+, \tilde{p})] \\ &\quad + \sigma_k^2 \Delta t \sqrt{1 - a_k^2} \Pi(x_+) \zeta + \sigma_k^2 \Delta t \phi(x_+, \tilde{p}) \end{aligned}$$

with pseudo-momentum

$$\tilde{p}(x_+, x_{++}) := \Pi(x_+) \left( \frac{x_{++} - x_+}{\sigma_{k+2}^2 \Delta t} \right) \in T_{x_+} \Sigma$$

, the normal term  $\phi(x_+, \tilde{p})$ , and the drift term  $b_\theta^k(x_+, \tilde{p})$ . Assume the following regularity:

- (Regularity of constraint functions) There exists constants  $c_\phi^0, c_\phi^1 < \infty$  such that for any square-integrable  $Y = (X, P)$ ,

$$\mathbb{E} \|\phi(Y)\|^2 \leq c_\phi^0 + c_\phi^1 (\|X\|^2 + \|P\|^2)$$

- (Regularity of function class) There exists constant  $L_g, C$  such that

$$\|b_\theta^k(x, p) - b_\theta^k(x', p')\| \leq L_g (\|x - x'\| + \|p - p'\|)$$

$$\|b_\theta^k(x, p)\| \leq C (1 + \|x\| + \|p\|)$$

Let  $\bar{T}_k^\theta$  be the associated Markov kernel to  $\bar{F}_\theta^k$ . That is,  $T_k^\theta(\cdot|x_+, x_{++}) = \text{Law}(\bar{F}_\theta^k(x_+, x_{++}, \zeta))$ . Then, for any probability measures  $\mu, \nu$  in  $\mathbb{R}^d$ , we have

$$W_2(\mu \bar{T}_k^\theta, \nu \bar{T}_k^\theta) \leq K_k W_2(\mu, \nu) + \mathcal{O} \left( \Delta t \left[ 1 + \sqrt{\mathbb{E}_{Y \sim \mu} \|Y\|^2} + \sqrt{\mathbb{E}_{Y' \sim \nu} \|Y'\|^2} \right] \right)$$

for some constant  $K_k$ . Also, if  $\mu$  is given as  $\mu = \rho \bar{T}_i^\theta \bar{T}_{i-1}^\theta \dots \bar{T}_j^\theta$  for  $i \geq j$  and some density  $\rho$  independent of  $\theta$ , the supremum of second moment of  $\mu$  is finite under  $\theta \in \Theta$ :

$$\sup_{\theta \in \Theta} \mathbb{E}_{Y_\mu \sim \mu} \|Y_\mu\|^2 < \infty$$

Combining these two, we can guarantee the existence of  $\Lambda_k > 0$  such that

$$W_2(\bar{\sigma}_{k-1}, \bar{\sigma}_k) \leq \Lambda_k W_2(\bar{q}_{k-1}, \bar{q}_k \bar{T}_k^\theta) + \mathcal{O}(\Delta t)$$

for  $k \in \{1, \dots, N-1\}$  and

$$W_2(\bar{\sigma}_{N-1}, \bar{p}_0^\theta) \leq \Lambda_N W_2(\bar{q}_{N-1}, \bar{p}_{N-1}^\theta \bar{T}_k^\theta) + \mathcal{O}(\Delta t) \quad (8)$$

*Proof.* Let  $(X_+, X_{++}), (X'_+, X'_{++})$  be the synchronous coupling for  $W_2(\mu, \nu)$  with shared noise  $\zeta \sim (0, I)$ . Write  $\Delta_+ := X_+ - X'_+$  and  $\Delta_{++} := X_{++} - X'_{++}$ , and

$$\Delta := \bar{F}_\theta^k(X_+, X_{++}, \zeta) - \bar{F}_\theta^k(X'_+, X'_{++}, \zeta)$$

Write  $\tilde{p} := \tilde{p}(X_+, X_{++}), \tilde{p}' := \tilde{p}(X'_+, X'_{++})$ , and

$$\Delta b := b_\theta^k(X_+, \tilde{p}) - b_\theta^k(X'_+, \tilde{p}'), \quad b' := b_\theta^k(X'_+, \tilde{p}')$$

Then,

$$\begin{aligned} \Delta &= \Delta_+ + \underbrace{\sigma_k^2 \Delta t \Pi(X_+) a_{k+1} (\tilde{p} - \tilde{p}')}_{(1)} + \underbrace{\sigma_k^2 \Delta t \Pi(X_+) \Delta b}_{(2)} + \underbrace{\sigma_k^2 \Delta t (\Pi(X_+) - \Pi(X'_+)) (a_{k+1} \tilde{p}' + b')}_{(3)} \\ &\quad + \underbrace{\sqrt{1 - a_k^2} \sigma_k^2 \Delta t (\Pi(X_+) - \Pi(X'_+)) \zeta}_{(4)} + \underbrace{\sigma_k^2 \Delta t (\phi(X_+, \tilde{p}) - \phi(X'_+, \tilde{p}'))}_{(5)} \end{aligned}$$

Now, note that (1) term is bounded by

$$\mathbb{E}\|(1)\|^2 \leq C_1 (\mathbb{E}\|X_{++} - X_+\|^2 + \mathbb{E}\|X'_{++} - X'_+\|^2) = \mathcal{O}(\Delta t^2)$$

for some constant  $C_1$  that depends on  $k$ . The term (2) is also can be bounded by

$$\mathbb{E}\|(2)\|^2 \leq C_2 \Delta t^2 \mathbb{E}\|\Delta_+\|^2 + C_2 (\mathbb{E}\|X_{++} - X_+\|^2 + \mathbb{E}\|X'_{++} - X'_+\|^2) = C \Delta t^2 \mathbb{E}\|\Delta_+\|^2 + \mathcal{O}(\Delta t^2)$$

for some  $C_2$ , because  $\|\Delta b\| \leq L_g (\|\Delta_+\| + \|\tilde{p} - \tilde{p}'\|)$ . For term (3), it is bounded by

$$\mathbb{E}\|(3)\|^2 \leq 4\sigma_k^4 \Delta t^2 (a_{k+1} \|\tilde{p}'\|^2 + \|b'\|^2) \leq C_3 \Delta t^2 (1 + \mathbb{E}\|X'_+\|^2)$$

for some constant  $C_3$ . Similarly, term (4) can be bounded by:

$$\mathbb{E}\|(4)\|^2 \leq 4\sigma_k^4 \Delta t^2 (1 - a_k^2) \mathbb{E}\|\zeta\|^2 = \mathcal{O}(\Delta t^2)$$

and term (5) is bounded by:

$$\mathbb{E}\|(5)\|^2 \leq 2\sigma_k^4 \Delta t^2 (\mathbb{E}\|\phi(X_+, \tilde{p})\|^2 + \mathbb{E}\|\phi(X'_+, \tilde{p}')\|^2) \leq C_5 \Delta t^2 (1 + \mathbb{E}\|X_+\|^2 + \mathbb{E}\|X'_+\|^2)$$

By combining these terms, we recover that

$$\mathbb{E}\|\Delta\|^2 \leq (1 + C_6 \Delta t) \mathbb{E}\|\Delta_+\|^2 + C_6 \Delta t^2 (1 + \mathbb{E}_{Y \sim \mu} \|Y\|^2 + \mathbb{E}_{Y' \sim \nu} \|Y'\|^2)$$

Since  $W_2(\mu, \nu)^2 \geq \mathbb{E}(\|\Delta_+\|^2 + \|\Delta_{++}\|^2) \geq \mathbb{E}\|\Delta_+\|^2$  holds, we have

$$W_2(\mu \bar{T}_k^\theta, \nu \bar{T}_k^\theta) \leq K_k W_2(\mu, \nu) + \mathcal{O}\left(\Delta t \left(1 + \sqrt{\mathbb{E}_{Y \sim \mu} \|Y\|^2} + \sqrt{\mathbb{E}_{Y' \sim \nu} \|Y'\|^2}\right)\right)$$

for some constant  $K_k > 0$ . Now, similarly as in overdamped case, one can show that similar algebraic techniques gives the following under regularity assumptions:

$$\mathbb{E}\|\bar{F}_\theta^k(X_+, X_{++}, \zeta)\|^2 \leq (1 + a_k \Delta t + \mathcal{O}(\Delta t^2)) \mathbb{E}(\|X_+\|^2 + \|X_{++}\|^2) + b_k \Delta t + \mathcal{O}(\Delta t^2)$$

for some constant  $a_k, b_k$ , using  $\mathbb{E}\|X_{++} - X_+\|^2 = \mathcal{O}(\Delta t)$  from our pair chain setup. Hence, the same logic as in the overdamped shows

$$\sup_{\theta \in \Theta} \mathbb{E}_{Y_\mu \sim \mu}^2 \|Y_\mu\|^2 < \infty$$

for  $\mu$  given as  $\mu = \rho \bar{T}_i^\theta \bar{T}_{i-1}^\theta \dots \bar{T}_j^\theta$  for  $i \geq j$  and some density  $\rho$  independent of  $\theta$ .  $\square$

## E EXPERIMENT SETTINGS AND SUPPLEMENTARY RESULTS

**Settings.** All experiments were implemented in Python using the PyTorch framework [54] and run in a Linux (Ubuntu) environment. The computational hardware was tailored to the specific experimental group. We utilized an NVIDIA L40S GPU with 45GB of VRAM for the Earth and climate science datasets, and an NVIDIA H100 GPU with 80GB of VRAM for the 3D mesh data experiments. All other tasks, including the  $SO(10)$  manifold, Alanine dipeptide, and the 7-DOF robot arm, were conducted on an NVIDIA H200 GPU with 141GB of VRAM.

### E.1 DESCRIPTION OF BASELINE ALGORITHMS

**Riemannian Flow Matching (RFM).** RFM [26] is a framework for training Continuous Normalizing Flows (CNF) [55] on Riemannian manifold by regressing a vector field  $v_t$  to a conditional target vector field  $u_t(x|x_1)$  for  $t \in [0, 1]$  defined via a user-specified premetric  $d(\cdot, \cdot)$  (e.g., geodesics, spectral distances). The model minimizes the Riemannian Conditional Flow Matching objective given as :

$$\mathcal{L}_{\text{RCFM}} = \mathbb{E}_{t \sim \mathcal{U}(0,1), x_1 \sim q_{\text{data}}, x_0 \sim p_{\text{prior}}} [\|v_t(x_t) - u_t(x_t | x_1)\|_g^2]$$

where  $x_t$  is as conditional flow sample interpolation between prior samples  $x_0$  and the data point  $x_1$ , and  $\|\cdot\|_g$  is the norm defined in the corresponding Riemannian manifold.

The computational requirements for  $x_t$  may depend on the manifold’s geometry. On simple manifold (e.g., spheres, tori), the geodesic distance can be used as the premetric, allowing  $x_t$  to be computed in closed form via the exponential map, thus making the algorithm simulation-free. In contrast, on general geometries (e.g., triangular meshes) where exact geodesics are intractable, spectral distances such as the biharmonic distance are employed as the premetric. In this case, computing  $x_t$  requires solving an ODE during the training process.

**Remark 5** (Implementation details on RFM). For RFM, we used the default configuration from the official code from authors. For Earth & Climate datasets, training iterations were reduced to 1/10, whereas Mesh data experiments were conducted using the unaltered default configuration.

**Riemannian Denoising Diffusion Probabilistic Models (RDDPM).** RDDPM [34] is a constrained diffusion model framework that adapts Denoising Diffusion Probabilistic Models (DDPMs) [1] to Riemannian manifold  $\Sigma := \{x \in \mathbb{R}^d \mid h(x) = 0\}$  setup by incorporating a Newton’s method projection step into the diffusion process.

The method constructs forward and backward Markov chains that alternate between diffusion steps along tangential direction of  $\Sigma$  and projecting the resulting sample back onto  $\Sigma$  via Newton’s method. While this guarantees feasibility at every step, the iterative nature of the projection leads to higher computational costs and potentially result in forward trajectory resampling due to projection failures. We remark that the projected version of OLLA (OLLA-P) corresponds to RDDPM under the equality-only scenario.

**Euclidean Forward with Backward Variants.** These baselines employ a standard unconstrained Euclidean diffusion process for the forward process, and distinguish themselves by the mechanism used to enforce constraints  $h(x) = 0, g(x) \leq 0$  during the backward process. In the forward process, it follows the update rule below:

$$x_{k+1} = x_k - \frac{\sigma_k^2 \Delta t}{2} \nabla f(x_k) + \sigma_k \sqrt{\Delta t} \zeta_k$$

with corresponding training loss

$$\mathcal{L}_{\text{Euclidean}}^{\text{over}}(\theta) = \mathbb{E}_{q(x_{0:N})} \left[ \sum_{k=0}^{N-1} \frac{\|x_k - \mu_{k+1}^o(x_{k+1})\|^2}{2\sigma_{k+1}^2 \Delta t} \right]$$

and  $\mu_{k+1}^o := x_{k+1} + \frac{\sigma_{k+1}^2 \Delta t}{2} [\nabla f(x_{k+1}) + s_{\theta}^{k+1}(x_{k+1})]$ .

1. **Euclidean:** This method performs sampling using the standard Euclidean backward without any constraint enforcement. The backward update rule is given as:

$$x_k = x_{k+1} + \frac{\sigma_{k+1}^2 \Delta t}{2} [\nabla f(x_{k+1}) + s_{\theta}^{k+1}(x_{k+1})] + \sigma_{k+1} \sqrt{\Delta t} \zeta_{k+1}$$

2538  
2539  
2540  
2541  
2542  
2543  
2544  
2545  
2546  
2547  
2548  
2549  
2550  
2551  
2552  
2553  
2554  
2555  
2556  
2557  
2558  
2559  
2560  
2561  
2562  
2563  
2564  
2565  
2566  
2567  
2568  
2569  
2570  
2571  
2572  
2573  
2574  
2575  
2576  
2577  
2578  
2579  
2580  
2581  
2582  
2583  
2584  
2585  
2586  
2587  
2588  
2589  
2590  
2591

This approach offers no guarantee that the generated samples lie on  $\Sigma$ .

2. **Projected:** This variant strictly enforces equality constraints by projecting the sample onto  $\Sigma$  immediately after each Euclidean backward step. Let  $\tilde{x}_k$  be the proposal from the Euclidean backward step. Then, the final state is obtained via  $x_k = \mathcal{P}_\Sigma(\tilde{x}_k)$ , where  $\mathcal{P}_\Sigma$  finds the root of  $h(y) = 0, g(y) \leq 0$  close to  $\tilde{x}_k$  using the interior point method [56, 57]. We remark that our implementation uses log-barrier for  $g(x) < 0$  and quadratic penalty for  $g(x) \geq 0$ .
3. **Lagrangian:** This method formulates the sampling step as a constrained optimization problem using the Augmented Lagrangian Method (ALM). At each timestep, the proposal  $\tilde{x}_k$  is refined by minimizing an augmented Lagrangian objective:

$$\mathcal{L}(x, \lambda, \mu) = \lambda^T h(x) + \frac{\rho}{2} \|h(x)\|^2 + \frac{1}{2\rho} (\|\text{ReLU}(\mu + \rho g(x))\|^2 - \|\mu\|^2)$$

The inequality term follows the Powell-Hestenes-Rockafellar (PHR) formulation. This specific form is derived by introducing a non-negative slack variable  $s \geq 0$  to convert the inequality constraint  $g(x) \leq 0$  into an equality  $g(x) + s = 0$ . By constructing the standard augmented Lagrangian for this equality and analytically minimizing it with respect to  $s$ , the slack variable is eliminated, resulting in the closed-form  $\text{ReLU}(\mu + \rho g(x))$  term. This ensures that penalties are applied correctly only when constraints are violated or multipliers are active. The multipliers  $\lambda$  and  $\mu$  are updated iteratively via dual ascent. We note that this approach is also introduced in Liang et al. [58].

4. **Guided:** This approach utilizes constraint guidance during sampling. The standard drift term of the backward process is modified by adding a guidance term derived from the gradient of a constraint violation energy potential. This potential is defined as  $V(x) = \frac{1}{2} \|h(x)\|^2 + \frac{1}{2} \|\text{ReLU}(g(x))\|^2$ , where the first term penalizes deviations from equality constraints and the second term penalizes violations of inequality constraints. Consequently, the backward update rule naturally incorporates a gradient descent step on this potential, which steers the generated trajectory towards the feasible set  $\Sigma$  by actively minimizing the constraint violation at each step.

## E.2 EXPERIMENT SETTINGS AND DESCRIPTIONS

**Earth and Climate Science Datasets  $S^2$ .** This benchmark [59, 60, 22, 61, 62] evaluates the model’s ability to learn geographical distributions on the Earth’s surface, which is modeled as the 2-sphere,  $S^2$ . The datasets represent the locations of phenomena such as volcanoes, earthquakes, floods, and fires.

*Mathematical formulation.* A sample  $x$  represents a point in 3D Euclidean space lying on the surface of a unit sphere. Thus,  $x \in \mathbb{R}^d$  with  $d = 3$ . The manifold is defined by a single, simple equality constraint  $h(x) = \|x\|_2 - 1 = 0$ .

*Prior distribution.* As this is a compact manifold, the prior distribution  $p_N$  is set to be the uniform distribution over the surface of the sphere  $S^2$ .

**3D Mesh Data on a Learned Manifolds.** The objective is to learn a probability distribution over the surface of a complex 3D shape, such as the Stanford Bunny [63] and Spot the Cow [64]. The manifold is implicitly defined as the zero-level set of a Signed Distance Function (SDF) that is itself represented by a pre-trained neural network  $h_{NN}(x)$  as performed in Rozen et al. [23], Gropp et al. [65].

*Mathematical Formulation.* A sample  $x$  represents a point in 3D Euclidean space, thus  $x \in \mathbb{R}^d$  with  $d = 3$ . The manifold is defined by a single equality constraint requiring any valid point to lie on the zero-level set of  $h_{NN}(x) = 0$ .

*Prior distribution.* The prior distribution  $p_N$  is chosen to be uniform distribution over the learned manifold surface  $\Sigma$  due to its compactness.

**High-Dimensional Special Orthogonal Group ( $SO(10)$ ).** This experiment tests the model’s ability to learn a multimodal distribution on the high-dimensional Lie group  $SO(10)$ . This is a challenging task due to the high dimensionality and non-trivial geometric structure of the manifold.

*Mathematical Formulation.* A sample is a  $10 \times 10$  matrix, which is vectorized into  $x \in \mathbb{R}^{100}$ . The constraints enforce the defining properties of a special orthogonal matrix. For the equality

constraints, we impose

$$h_{ij}(X) = (X^T X - I)_{ij} = 0 \quad \text{for } 1 \leq i \leq j \leq 10$$

and the determinant condition  $\det(X) = 1$  is handled by via rejection when it is violated.

*Prior distribution.* Similarly, the manifold is compact and we choose uniform distribution over  $SO(10)$  as our prior distribution.

**Alanine Dipeptide** This task involves generating valid 3D conformations of Alanine dipeptide, a model system in biophysics. The goal is to learn the distribution of structures subject to constraints on specific internal coordinates, including a mixed equality and inequality setup. Following the same approach in Liu et al. [34], we generated the dataset by running a 1ns constrained molecular dynamics simulation of alanine dipeptide in water using GROMACS [66] with a 1 fs timestep. A harmonic bias was applied through the COLVARS module [67], where the chosen collective variable was dihedral angle  $\phi$ . The harmonic restraint was centered at  $\phi = -70^\circ$  with a force constant 5.0. Other simulation settings follow closely those reported in Lelièvre et al. [68]. In total,  $10^4$  configurations were collected by saving a snapshot every 100 simulation steps. Hydrogen atoms were removed, leaving the coordinates of the 10 heavy atoms for further analysis.

*Mathematical Formulation.* The state  $x$  consists of the 3D coordinates of the 10 non-hydrogen atoms, so  $x \in \mathbb{R}^{30}$ . The constraints are placed on two of the molecule’s primary dihedral angles,  $\phi$  and  $\psi$ . For the equality constraints, the dihedral angle  $\phi$  is fixed to a specific value:

$$h(x) = \phi(x) - (-70^\circ)_{\text{rad}} = 0$$

and we impose an inequality constraint so that another adjacent dihedral angle  $\psi$  is constrained to lie within the range  $[130^\circ, 170^\circ]$ . This is formulated as a single inequality:

$$g(x) = \max \{ \psi(x) - 170^\circ_{\text{rad}}, 130^\circ_{\text{rad}} - \psi(x) \} \leq 0.$$

*Prior distribution.* Instead of introducing a potential-based drift term to induce a specific unimodal prior as in Liu et al. [34], we employ an empirical prior strategy. We first generate a large set of forward trajectories using the corresponding constrained dynamics (OLLA/ULLA) by running them to approximate the terminal prior distribution  $q_N$  on the feasible set. The terminal states  $x_N$  of these trajectories are collected, and the backward sampling process is initiated by drawing starting points uniformly from this pre-computed set, serving as a discrete approximation of the prior. Furthermore, to ensure the generated conformations respect physical symmetries, the score network for this task is designed to be  $SE(3)$ -invariant as proposed in Liu et al. [34].

**7-DOF Robot Arm Trajectory** This experiment focuses on learning a complex, bimodal distribution of trajectories for a 7-DOF Franka Emika Panda robot arm. The model is trained on a dataset of 400 valid paths (200 for S-shaped, 200 for reverse S-shaped paths) generated by the Rapidly-exploring Random Tree (RRT) algorithm. The primary task is to generate trajectories that trace both S-shaped and reverse S-shaped paths between fixed start and end points. Throughout the motion, the generated trajectories must satisfy several critical constraints: the robot arm must navigate around two spherical obstacles, and its end-effector must maintain a constant height of  $z = z_{\text{target}} = 0.1$ .

*Mathematical Formulation.* The fundamental state of the robot arm is its configuration in joint space, represented by a vector of 7 joint angles,  $\theta \in \mathbb{R}^7$ . A trajectory is a time-discretized sequence of these configurations,  $(\theta_l)_{l=1}^L$ . To avoid the periodicity issue of raw angles, which poses challenges for neural networks, we represent each joint angle  $\theta_{l,j}$  as a 2D vector on the unit circle  $(\cos(\theta_{l,j}), \sin(\theta_{l,j}))$ . Consequently, the state at a single time step  $l$  is a vector  $x_l \in \mathbb{R}^{14}$ . The full trajectory is flattened into a single vector  $x = [x_1, \dots, x_L] \in \mathbb{R}^d$ . For a trajectory with  $L \in \{10, 20, 30, 40\}$  as in our setup, the ambient space dimension is  $d \in \{140, 280, 420, 560\}$ .

The constraints on the robot’s behavior, such as end-effector position and obstacle avoidance, are defined in 3D Cartesian space. We bridge the joint space representation and the Cartesian space constraints using the forward kinematics function,  $\text{FK} : \mathbb{R}^7 \rightarrow \mathbb{R}^{3 \times K}$ , which maps a set of joint angles  $\theta_l$  to the 3D positions of the  $K = 7$  links of the robot arm. To handle the large number of resulting constraints efficiently, we employ a “summation trick” to combine multiple constraint violations into a single function for both equalities and inequalities. In particular, multiple geometric and kinematic conditions are aggregated into a single sum-of-squares function:

$$h(x) = \sum_{i=1}^m h_i(x)^2 = 0.$$

The individual components  $h_i(x)$  enforce: (1) the validity of the joint representation,  $h_{\text{rep}}(x_{l,j}) = \cos^2(\theta_{l,j}) + \sin^2(\theta_{l,j}) - 1 = 0$ , for each joint  $j$  at each time step  $l$ , (2) fixed start and end points for the trajectory

$$h_{\text{end}}(x_L) = \|\text{FK}(\theta_L)_{\text{end-effector}} - p_{\text{end}}\|^2 = 0, \quad h_{\text{start}}(x_1) = \|\text{FK}(\theta_0)_{\text{end-effector}} - p_{\text{start}}\|^2 = 0$$

with  $p_{\text{start}}$  and  $p_{\text{end}}$  being the target start and end positions, and (3) a fixed  $z$ -height for the end effector throughout the trajectory,  $h_z(x_l) = [\text{FK}(\theta_l)_{\text{end-effector}}]_z - z_{\text{target}} = 0$ .

For the inequality constraint, the robot arm must avoid two spherical obstacles. For each relevant robot link  $k \in [K]$  and obstacle  $o \in \{1, 2\} := N_{\text{obs}}$ , the distance between them must exceed a safety margin. These conditions are combined into a single function by summing the rectified violations:

$$g(x) = \sum_{l=1}^L \sum_{k=1}^K \sum_{o=1}^{N_{\text{obs}}} \text{ReLU}((r_{\text{obs},o} + r_{\text{safety}}) - \|\text{FK}(\theta_l)_{\text{link},k} - p_{\text{obs},o}\|) \leq 0.$$

with  $r_{\text{obs},o}, p_{\text{obs},o}$  being the radius and position of obstacles. This function is non-positive if and only if all links maintain the required minimum distance  $r_{\text{safety}}$  from all obstacles throughout the entire trajectory.

*Prior distribution.* Similar to the Alanine Dipeptide task, we employ an empirical prior strategy. We first generate a large set of forward trajectories using the corresponding constrained dynamics (OLLA/ULLA) by running them to approximate the target prior distribution  $q_N$  on the feasible set. The terminal states  $x_N$  of these trajectories are collected, and the backward sampling process is initiated by drawing starting points uniformly from this pre-computed set, serving as a discrete approximation of the prior.

Table 6: **Summary of constrained feasible set dimensions and constraint specifications.** Below table represent the ambient dimension  $d$ , the intrinsic manifold dimension, and the number of equality ( $m$ ) and inequality ( $l$ ) constraints. For the Robot Arm task,  $L$  denotes the number of time steps (e.g.,  $L \in \{10, \dots, 40\}$ ), and the constraint counts  $m$  and  $l$  are reported before applying the summation trick.

Dataset / Task	Ambient Dim. ( $d$ )	Intrinsic Dim.	Equality ( $m$ )	Inequality ( $l$ )
Earth & Climate ( $S^2$ )	3	2	1	0
3D Mesh (Bunny / Spot)	3	2	1	0
Lie Group $SO(10)$	100	45	55	0
Alanine Dipeptide	30	29	1	2
7-DOF Robot Arm	$14L$	$7L$	$8L + 2$	$14L$

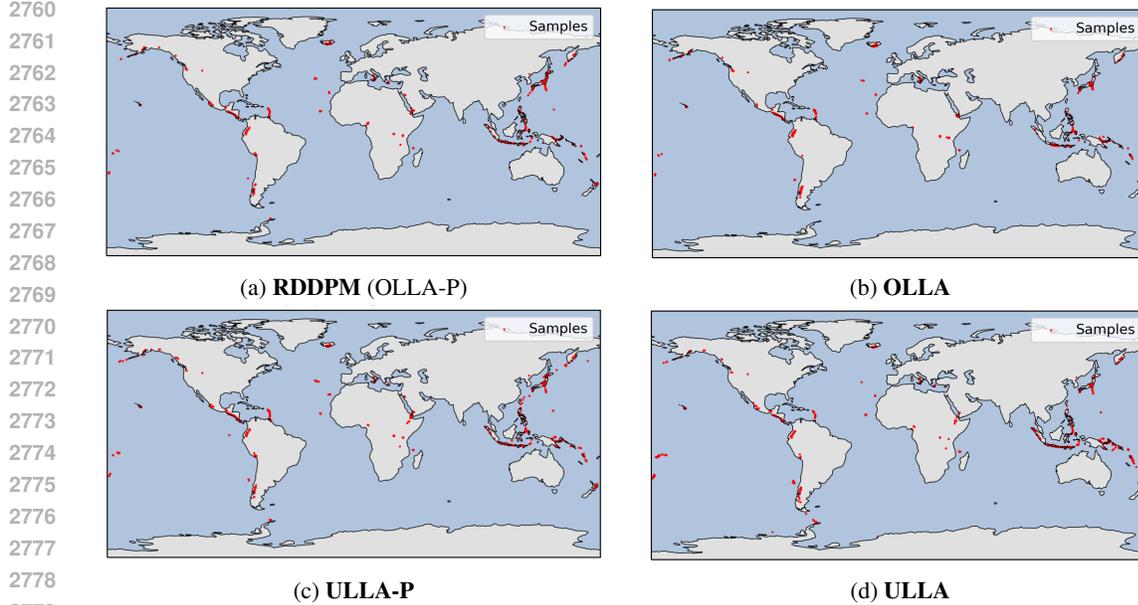
Table 7: Detailed hyperparameters for all datasets, specified per algorithm.

Dataset	Algorithm	$\gamma$	$\sigma_{\min}$	$\sigma_{\max}$	$N$	$T$	$l_f$	$N_{epoch}$	$B$	$N_{node}$	$N_{layer}$	$\alpha$	$\epsilon$
Volcano	OLLA	-	0.01	1.0	100	4.0	1	20000	128	512	5	50	-
	ULLA	3	0.1	1.3	50	2.0	1	20000	128	512	5	50	-
	ULLA-P	3	0.1	1.3	50	2.0	1	20000	128	512	5	50	-
Earthquake	OLLA	-	0.01	1.0	100	4.0	1	20000	512	512	5	50	-
	ULLA	3	0.1	1.3	50	2.0	1	20000	512	512	5	50	-
	ULLA-P	3	0.1	1.3	50	2.0	1	20000	512	512	5	50	-
Flood	OLLA	-	0.01	1.0	100	4.0	1	20000	512	512	5	50	-
	ULLA	3	0.1	1.3	50	2.0	1	20000	512	512	5	50	-
	ULLA-P	3	0.1	1.3	50	2.0	1	20000	512	512	5	50	-
Fire	OLLA	-	0.01	1.0	100	4.0	1	20000	512	512	5	50	-
	ULLA	3	0.1	1.3	50	2.0	1	20000	512	512	5	50	-
	ULLA-P	3	0.1	1.3	50	2.0	1	20000	512	512	5	50	-
Bunny ( $k = 50$ )	OLLA	-	0.07	0.07	100	8.0	100	2000	2048	256	5	25	-
	ULLA	20	0.2	0.6	30	3.0	100	2000	2048	256	5	25	-
	ULLA-P	20	0.2	0.6	50	3.0	100	2000	2048	256	5	25	-
Bunny ( $k = 100$ )	OLLA	-	0.07	0.07	100	5.0	100	2000	2048	256	5	25	-
	ULLA	20	0.2	0.6	30	3.0	100	2000	2048	256	5	25	-
	ULLA-P	20	0.2	0.6	50	3.0	100	2000	2048	256	5	25	-
Spot ( $k = 50$ )	OLLA	-	0.1	0.1	100	5.0	100	2000	2048	256	5	25	-
	ULLA	20	0.2	0.5	30	3.0	100	2000	2048	256	5	25	-
	ULLA-P	20	0.2	0.5	50	3.0	100	2000	2048	256	5	25	-
Spot ( $k = 100$ )	OLLA	-	0.1	0.1	100	3.0	100	2000	2048	256	5	25	-
	ULLA	20	0.2	0.5	30	3.0	100	2000	2048	256	5	25	-
	ULLA-P	20	0.2	0.5	50	3.0	100	2000	2048	256	5	25	-
SO(10) ( $m = 3$ )	OLLA	-	0.2	2.0	100	1.0	100	2000	512	512	3	50	-
	ULLA	50	0.3	2.2	50	1.0	50	2000	512	512	3	5	-
	ULLA-P	50	0.3	2.2	50	1.0	50	2000	512	512	3	5	-
SO(10) ( $m = 5$ )	OLLA	-	0.2	2.0	100	1.0	100	2000	512	512	3	50	-
	ULLA	50	0.3	2.2	50	1.0	50	2000	512	512	3	5	-
	ULLA-P	50	0.3	2.2	50	1.0	50	2000	512	512	3	5	-
Alanine dipeptide	ULLA-P	10	0.5	2.0	100	0.2	10	2000	512	1024	5	50	0.05
7-DOF robot arm	ULLA-P	1	0.1	2.0	100	0.2	50	10000	160	1024	5	200	0.001

2754 E.3 GENERATED SAMPLES FROM TASKS.  
 2755

2756 In this subsection, we demonstrate the effectiveness of our proposed methods by comparing the  
 2757 generated samples in various tasks to the baseline RDDPM [34].  
 2758

2759 **Earth and Climate Science Datasets  $S^2$  –Volcano.**



2780 Figure 5: Comparison of generated distributions across different algorithms-Volcano dataset  
 2781

2782 **3D Mesh data on learned manifold – Spot the Cow ( $k = 100$ ).**

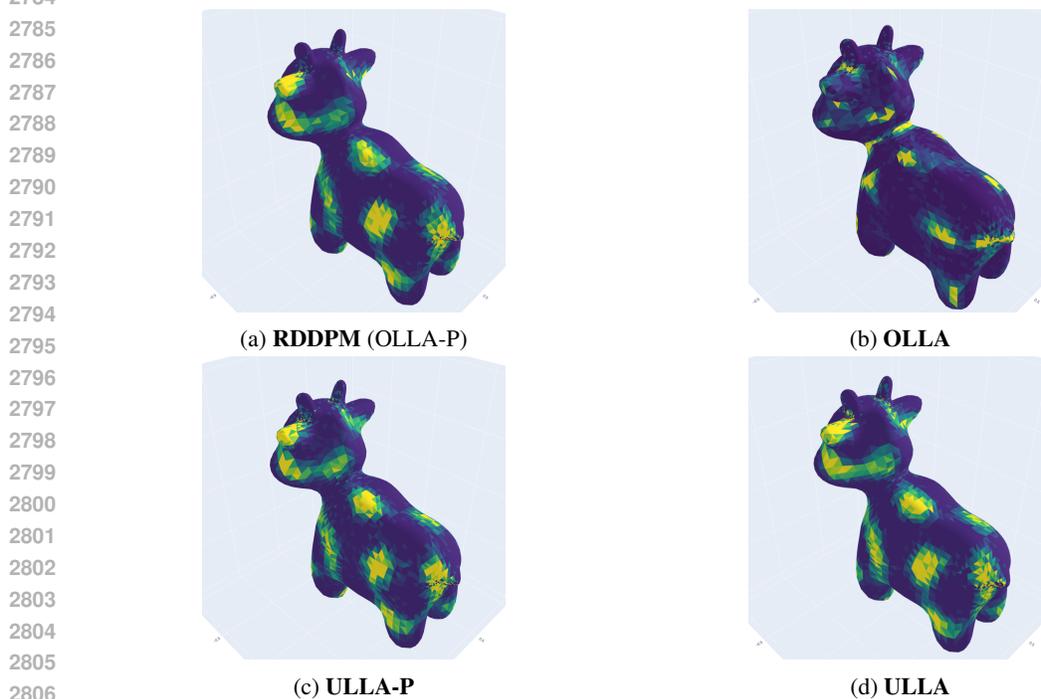


Figure 6: Comparison of generated distributions across different algorithms- Spot the Cow  $k = 100$

2808 **SO(10) manifold with  $m = 3$ .**

2809

2810

2811

2812

2813

2814

2815

2816

2817

2818

2819

2820

2821

2822

2823

2824

2825

2826

2827

2828

2829

2830

2831

2832

2833

2834

2835

2836

2837

2838

2839

2840

2841

2842

2843

2844

2845

2846

2847

2848

2849

2850

2851

2852

2853

2854

2855

2856

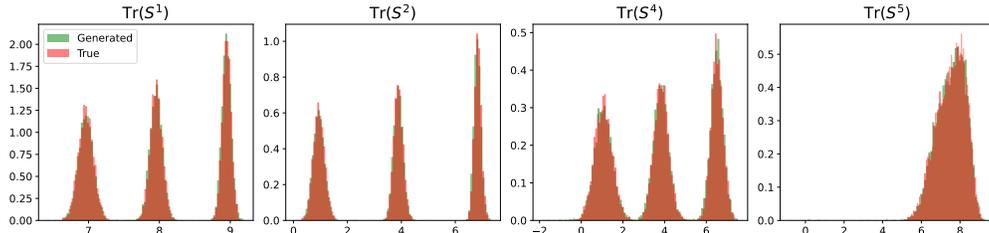
2857

2858

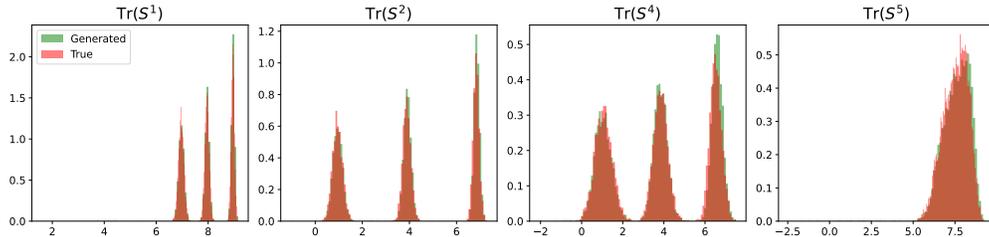
2859

2860

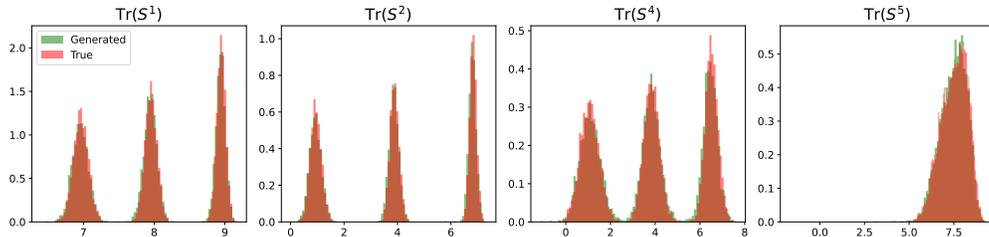
2861



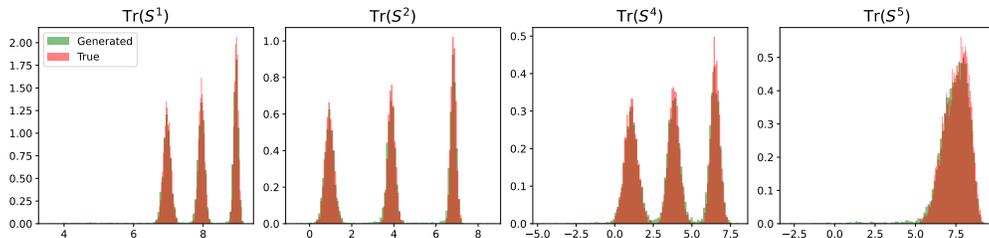
(a) RDDPM (OLLA-P)



(b) OLLA



(c) ULLA-P

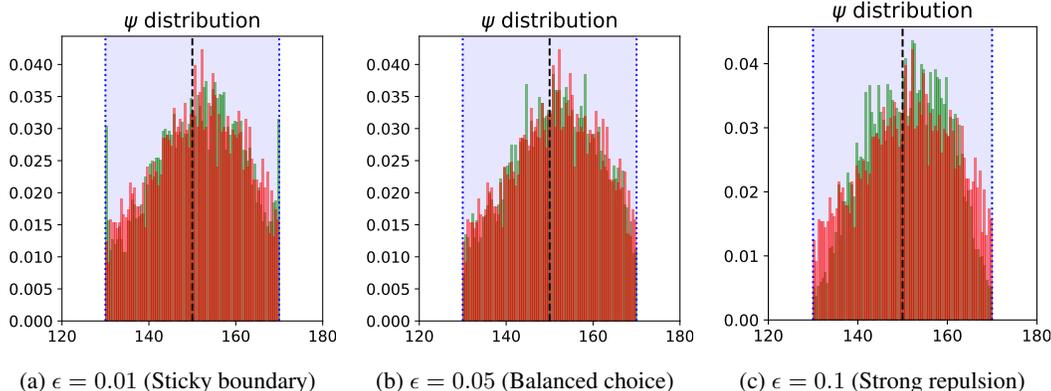


(d) ULLA

Figure 7: Comparison of generated distributions across different algorithms-  $SO(10)$  with  $m = 3$

E.4 SUPPLEMENTARY RESULTS - EFFECT OF HYPERPARAMETERS  $\alpha$  AND  $\epsilon$ .Table 8: Effect of hyperparameters  $\alpha$  (with fixed  $\epsilon = 0.05$ ) and  $\epsilon$  (with fixed  $\alpha = 50$ ) on JSD metrics and constraint violations for Alanine Di-peptide using ULLA without last step projection.

Parameter	JSD ( $\psi$ angle)	JSD (RMSD)	$\mathbb{E}[ h(x) ]$	$\mathbb{E}[g(x)^+]$
<i>Effect of <math>\alpha</math> (with <math>\epsilon = 0.05</math>)</i>				
$\alpha = 0.1$	0.419 $\pm$ .000	0.045 $\pm$ .002	$5.40 \times 10^{-3}$	$3.79 \times 10^{-3}$
$\alpha = 0.5$	0.247 $\pm$ .000	0.036 $\pm$ .001	$5.39 \times 10^{-3}$	$2.82 \times 10^{-3}$
$\alpha = 1.0$	0.134 $\pm$ .004	0.034 $\pm$ .003	$5.51 \times 10^{-3}$	$1.60 \times 10^{-3}$
$\alpha = 5.0$	0.060 $\pm$ .002	0.033 $\pm$ .001	$2.97 \times 10^{-3}$	$3.42 \times 10^{-4}$
$\alpha = 10.0$	0.053 $\pm$ .004	0.034 $\pm$ .002	$1.02 \times 10^{-3}$	$1.46 \times 10^{-4}$
$\alpha = 20.0$	0.043 $\pm$ .001	0.035 $\pm$ .002	$2.09 \times 10^{-4}$	$4.28 \times 10^{-7}$
$\alpha = 50.0$	0.033 $\pm$ .002	0.034 $\pm$ .002	$7.20 \times 10^{-5}$	$7.95 \times 10^{-8}$
$\alpha = 100.0$	0.051 $\pm$ .003	0.035 $\pm$ .002	$3.70 \times 10^{-5}$	$6.92 \times 10^{-7}$
$\alpha = 200.0$	0.059 $\pm$ .008	0.183 $\pm$ .066	$2.73 \times 10^{-3}$	$4.19 \times 10^{-3}$
$\alpha = 400.0$	0.235 $\pm$ .001	NaN	$6.74 \times 10^{-1}$	$9.95 \times 10^{-1}$
<i>Effect of <math>\epsilon</math> (with <math>\alpha = 50</math>)</i>				
$\epsilon = 0.001$	0.084 $\pm$ .001	0.036 $\pm$ .002	$7.00 \times 10^{-5}$	$3.49 \times 10^{-6}$
$\epsilon = 0.005$	0.054 $\pm$ .003	0.033 $\pm$ .002	$7.30 \times 10^{-5}$	$3.58 \times 10^{-5}$
$\epsilon = 0.01$	0.048 $\pm$ .002	0.032 $\pm$ .002	$7.00 \times 10^{-5}$	$5.49 \times 10^{-7}$
$\epsilon = 0.05$	0.033 $\pm$ .002	0.034 $\pm$ .002	$7.20 \times 10^{-5}$	$7.95 \times 10^{-8}$
$\epsilon = 0.1$	0.052 $\pm$ .007	0.035 $\pm$ .002	$2.57 \times 10^{-4}$	$5.26 \times 10^{-4}$
$\epsilon = 0.5$	0.081 $\pm$ .009	0.738 $\pm$ .012	$2.37 \times 10^{-1}$	$1.98 \times 10^{-1}$
$\epsilon = 1.0$	0.107 $\pm$ .024	0.788 $\pm$ .001	$3.01 \times 10^{-1}$	$1.95 \times 10^{-1}$
$\epsilon = 5.0$	0.134 $\pm$ .002	0.796 $\pm$ .002	$7.34 \times 10^{-1}$	$2.97 \times 10^{-1}$
$\epsilon = 10.0$	0.188 $\pm$ .003	0.796 $\pm$ .002	$1.05 \times 10^0$	$1.08 \times 10^0$

Figure 8: **Effect of boundary repulsion rate  $\epsilon$  on the generated distribution.** When  $\epsilon$  is too small (a), trajectories tend to stick to the boundary. Conversely, an excessively large  $\epsilon$  (c) aggressively pushes samples away from the boundary, distorting the distribution. A moderate choice (b) balances these effects, yielding the best sampling quality.